Fairness Maximization among Offline Agents in Online-Matching Markets

WILL MA, Columbia University, New York, NY
PAN XU, New Jersey Institute of Technology, Newark, NJ
YIFAN XU, Southeast University, Nanjing, China

16

Online matching markets (OMMs) are commonly used in today's world to pair agents from two parties (whom we will call *offline* and *online* agents) for mutual benefit. However, studies have shown that the algorithms making decisions in these OMMs often leave disparities in matching rates, especially for offline agents. In this article, we propose online matching algorithms that optimize for either individual or group-level fairness among offline agents in OMMs. We present two linear-programming (LP) based sampling algorithms, which achieve competitive ratios at least 0.725 for individual fairness maximization and 0.719 for group fairness maximization. We derive further bounds based on fairness parameters, demonstrating conditions under which the competitive ratio can increase to 100%. There are two key ideas helping us break the barrier of $1 - 1/e \sim 63.2\%$ for competitive ratio in online matching. One is *boosting*, which is to adaptively re-distribute all sampling probabilities among only the available neighbors for every arriving online agent. The other is *attenuation*, which aims to balance the matching probabilities among offline agents with different mass allocated by the benchmark LP. We conduct extensive numerical experiments and results show that our boosted version of sampling algorithms are not only conceptually easy to implement but also highly effective in practical instances of OMMs where fairness is a concern.

CCS Concepts: • **Theory of computation → Online algorithms**; • **Applied computing → Marketing**; • **Computing methodologies → Planning under uncertainty;**

Additional Key Words and Phrases: Fairness maximization, online matching markets

ACM Reference format:
Will Ma, Pan Xu, and Yifan Xu. 2023. Fairness Maximization among Offline Agents in Online-Matching Markets. *ACM Trans. Econ. Comput.* 10, 4, Article 16 (April 2023), 27 pages.
https://doi.org/10.1145/3569705

1 INTRODUCTION

Matching markets involve heterogeneous agents (typically from two parties) who are paired for mutual benefit. During the last decade, matching markets have emerged and grown rapidly through the medium of the Internet. They have evolved into a new format, called **Online**

A preliminary version of this work appeared as an extended abstract in the 17th Conference on Web and Internet Economics (WINE), 2021. Pan Xu was partially supported by NSF CRII Award IIS-1948157.
Authors' addresses: W. Ma, Columbia University, New York, NY 10027, USA; email: willma353@gmail.com; P. Xu, New Jersey Institute of Technology, 323 Dr Martin Luther King Jr Blvd, Newark, NJ 07102, USA; email: pxu@njit.edu; Y. Xu (corresponding author), Southeast University, No. 2 Sipailou, Nanjing, Jiangsu 210096, CN; email: xyf@seu.edu.cn.

2167-8375/2023/04-ART16 $15.00
https://doi.org/10.1145/3569705

Matching Markets (OMMs), with examples ranging from crowdsourcing markets to online recommendations to rideshare. There are two features distinguishing OMMs from traditional matching markets. The first is the dynamic arrivals of one side of the market, whose constituents are referred to as *online agents*, *e.g.,* keywords in Google Advertising, workers in **Amazon Mechanical Turk (AMT)**, and riders in Uber/Lyft. The constituents of the other side are known *a priori* and referred to as *offline agents*, such as sponsors in Google advertising, tasks in AMT, and drivers in rideshare platforms (when restricted to a short time window).[1] The second feature is the instant match-decision requirement. It is highly desirable to match each online agent with one (or multiple) offline agent(s) upon its arrival, due to the low "patience" of online agents. There are several articles that study matching-policy and/or pricing mechanism design in OMMs [6, 34]. The main focus of this article, instead, is fairness among offline agents in OMMs. Consider the following two motivating examples.

Fairness among Task Requesters in Mobile Crowdsoucing Markets (MCM). In MCMs like Gigwalk and TaskRabbit, each offline task has specific location information, and online workers have to travel to that location to complete it (*e.g.,* cleaning one's house, delivering some package). Survey results on TaskRabbit in Chicago [51] showed that tasks from certain areas with low socioeconomic status, like the South Side of Chicago, get assigned and completed at a much slower rate, and as a result "low socioeconomic-status areas are currently less able to take advantage of the benefit of mobile crowdsourcing markets."

Fairness among Drivers in Ride-Hailing Services. There are several reports showing the earning gap among drivers based on their demographic factors such as age, gender and race, see, *e.g.,* [11] and [45]. In particular, Hinchliffe [26] has reported that "Black Uber and Lyft drivers earned \$13.96 an hour compared to the \$16.08 average for all other drivers" and "Women drivers reported earning an average of \$14.26 per hour, compared to \$16.61 for men". The wage gap among drivers from different demographic groups is a consequence of the algorithms being employed which, for one reason or another, end up allocating jobs to different demographics in ways that result in statistically significant discrepancies in average wage rates.

Therefore, in this article, we explicitly propose fairness as an objective for the online matching algorithm. We study the prototypical online matching model with *known*, *independent*, and *identical* **(KIID)** arrivals, which is widely used to model the dynamic arrivals of online agents in several real-world OMMs including rideshare and crowdsourcing markets [14, 18, 54]. We choose this arrival model instead of the adversarial one, because under the latter it is impossible for an online algorithm to perform well.[2]

The online matching model under KIID is as follows. We have a bipartite graph (I, J, E), where I and J represent the types of offline and online agents, respectively, and an edge $e = (i, j)$ indicates the compatibility between the offline agent (of type) i and the online agent (of type) j. All offline agents are static, while online agents arrive dynamically in a certain random way. Especially, we have an online phase consisting of T rounds and during each round $t \in [T] \doteq \{1, 2, \ldots, T\}$, one single online agent $\hat{j}$ will be sampled (called $\hat{j}$ *arrives*) with replacements such that $\Pr[\hat{j} = j] = r_j/T$ for all $j \in J$ and $\sum_{j \in J} r_j/T = 1$. We assume that the arrival distribution $\{r_j/T\}$ is *known*, *independent*, and *identical* (KIID) throughout the online phase. Upon the arrival of an online agent

[1]More precisely, "offline agents" in this article represent a general class of agents that have less mobility in nature compared with the other party in the matching system; thus, they could join and depart the system dynamically in practice though at a far slower pace as the online counterpart.

[2]This can be seen as follows: Suppose there are a large number of tasks, defined by T. The first arriving worker can perform any of them. Workers $2, \ldots, T$ can each perform a different specialized task such that there is one task that can be served only by the first arriving worker. The online algorithm has a $1/T$ chance of correctly allocating the first worker to this task.

j, an immediate and irrevocable decision is required: either reject j or match j with one offline neighbor i with $i \sim j$, *i.e.,* $(i, j) \in E$. Throughout this article, we assume w.l.o.g. that each offline agent has a unit matching capacity.[3] Suppose we have a collection of groups $\mathcal{G} = \{G\}$, where each group G is a set of types of offline agents (possibly overlapping) that share some demographic characteristic such as gender, race, or religion. Consider a generic online algorithm ALG, and let $Z_i = 1$ indicate that offline agent i is matched (or served). We define the following two objectives:

Individual Fairness Maximization (IFM): $\max \min_{i \in I} \mathsf{E}[Z_i]$;
Group Fairness Maximization (GFM): $\max \min_{G \in \mathcal{G}} \frac{1}{|G|} \sum_{i \in G} \mathsf{E}[Z_i]$, where $|G|$ is the cardinality of G.

Here IFM denotes the minimum expected matching rate over all individual offline agents, while GFM denotes that over all pre-specified groups of offline agents. Our goal is to design an online-matching policy such that the above two objectives are maximized. Observe that IFM can be captured as a special case of GFM when each group consists of one single offline type. Also observe that the objective achieved will always be higher under GFM than IFM (because $\frac{1}{|G|} \sum_{i \in G} \mathsf{E}[Z_i] \geq \min_{i \in I} \mathsf{E}[Z_i]$ for any group G); however, the objective can often be much better under GFM than in the special case where it captures IFM.

*A related model: vertex-weighted online matching under KIID (*VOM*).* VOM under KIID [9, 28] shares almost the same setting as our model except in the objective, where each offline agent i is associated with a non-negative weight w_i and the objective is to maximize the expected total weight of all matched offline agents, *i.e.,* $\max \sum_{i \in I} w_i \cdot \mathsf{E}[Z_i]$.

Two assumptions on the arrival setting. (a) Integral arrival rates. Observe that for an offline agent j, it will arrive with probability r_j/T during each of the online T rounds. Thus, r_j is equal to the expected number of arrivals of j during the online phase and it is called *arrival rate* of j. In this article, we consider integral arrival rates for all offline agents, and by following a standard technique of creating r_j copies of j, we can further assume w.l.o.g. that all $r_j = 1$ [9]. (b) $T \gg 1$. *This is a standard assumption in the literature of online bipartite matching under KIID* [19, 24, 28, 39], where the objective is typically to maximize a linear function representing the total weight of all matched edges. We emphasize that both of these assumptions are mild in that: the interesting case is $T \gg 1$ because if T is fixed then the problem, with state space $|I|^T$, can be solved to optimality; and under the assumption $T \gg 1$, the arrival rates are arbitrarily close to integers.

2 PRELIMINARIES AND MAIN CONTRIBUTIONS

2.1 A Clairvoyant Optimal and Competitive Ratio

Competitive ratio (CR) is a common metric to evaluate the performance of online algorithms [41]. Consider an online maximization problem like ours. Let $\text{ALG}(\mathcal{I}) = \mathsf{E}_{\mathcal{A} \sim \mathcal{I}}[\text{ALG}(S)]$ denote the expected performance of ALG on an input $\mathcal{I}$, where the expectation is taken over both of the randomness of the arrival sequence $\mathcal{A}$ of online agents and that of ALG. Let $\text{OPT}(\mathcal{I}) = \mathsf{E}_{\mathcal{A} \sim \mathcal{I}}[\text{OPT}(S)]$ denote the expected performance of *a clairvoyant optimal* who has the privilege to optimize decisions *after* observing the full arrival sequence $\mathcal{A}$. We say ALG achieves a competitive ratio of $\rho \in [0, 1]$ if $\text{ALG}(\mathcal{I}) \geq \rho\, \text{OPT}(\mathcal{I})$ for all possible inputs $\mathcal{I}$. Generally, the competitive ratio captures the gap in the performance between an online algorithm subject to the real-time decision-making requirement and a clairvoyant optimal who is exempt from that. It is a common technique to use a **linear program (LP)** to upper bound the clairvoyant optimal (called

[3]For an offline agent i with a capacity of $b_i \in \mathbb{Z}_+$, the individual fairness on i after normalization (see definition below) will be reduced to the group-level fairness by treating it as a group consisting of b_i identical copies each having a unit capacity.

benchmark LP), and hence by comparing against the optimal value of the benchmark LP, we can get a valid lower bound on the target competitive ratio.

2.2 Benchmark Linear Programs

In this article, we use the benchmark LP as shown below. For each edge $(i,j) \in E$, let x_{ij} be the expected number of times that the edge (i,j) is matched by the clairvoyant optimal. For each i, let $\mathcal{N}_i = \{j : (ij) \in E\}$ denote the set of offline neighbors of i. Similarly, $\mathcal{N}_j$ denotes the set of online neighbors of j. For notation convenience, we use $i \sim j$ and $j \sim i$ to denote the relation that i is incident to j (*i.e.,* $i \in \mathcal{N}_j$) and j is incident to i (*i.e.,* $j \in \mathcal{N}_i$), respectively.

$$\text{IFM}: \ \max \ \min_{i \in I} x_i \tag{1}$$

$$\text{GFM}: \ \max \ \min_{G \in \mathcal{G}} \frac{1}{|G|} \sum_{i \in G} x_i \tag{2}$$

$$\text{VOM}: \ \max \ \sum_{i \in I} w_i x_i \tag{3}$$

$$x_j := \sum_{i \sim j} x_{ij} \leq 1, \qquad \forall j \in J \tag{4}$$

$$x_i := \sum_{j \sim i} x_{ij} \leq 1, \qquad \forall i \in I \tag{5}$$

$$0 \leq \sum_{j \in S} x_{ij} \leq 1 - \mathrm{e}^{-|S|}, \ \forall S \subseteq \mathcal{N}_i, |S| = O(1),^{4} \ \forall i \in I. \tag{6}$$

Throughout this article, we use LP (1) to denote the LP with Objective (1) and Constraints (4) to (6). Similarly for LP (2) and LP (3). Note that though Objective (1) is non-linear, we can reduce to a linear one by replacing it with max λ and adding one constraint $\sum_{j \sim i} x_{ij} \geq \lambda$ for all $i \in I$. Similarly for Objective (2), we can replace it with max λ and add one extra constraint $\sum_{i \in G} \sum_{j \sim i} x_{ij} \geq \lambda \cdot |G|$ for all $G \in \mathcal{G}$, where $|G|$ denotes the cardinality of group G. Our LPs are mainly inspired by [9]. In particular, Constraint (6) suggests that any offline agent i is matched by one of its neighbors in S with a probability no more than $1 - \mathrm{e}^{-|S|}$; see more details in the proof of Lemma 1. For the critical Constraint (6), we have omitted terms of size $O(1/T)$; see the proof of Lemma 1 for more details. Those errors can affect the competitive ratio by at most $O(1/T)$ [9].[5] Note that all the above three LPs can be solved polynomially even after removing the restriction on the size of $|S|$ in Constraint (6).[6]

Lemma 1. LP (1), LP (2), *and* LP (3) *are valid benchmarks for* IFM, GFM, *and* VOM, *respectively.*

Proof. Observe that the three objectives (1), (2), and (3) capture metrics of IFM, GFM, and VOM, respectively. Thus, it will suffice to justify the feasibility of all constraints for a clairvoyant optimal. Constraint (4) is valid since the total number of matches relevant to an online agent j should be no larger than that of expected arrivals, which is $r_j = 1$. Constraint (5) is due to that every offline agent i has a unit matching capacity. Constraint (6) can be justified as follows: Consider a given

[4]Here we assume the size of $|S|$ is upper bounded by a given constant, say $K = 100$, which is independent of $T \gg 1$.
[5]Further evidence can be seen from the competitive analysis of our algorithms. Consider SAMP-B, for example. The final competitive ratio is stated as a piecewise function with bounded first derivatives in the whole domain (Theorem 2), which suggests any error in Constraint (6) can get inflated by at most a constant in the final ratio.
[6]Though the LPs may have exponential number of constraints after removing the restriction that $|S|$ is a constant independent of T, they all can be solved polynomially since they admit a polynomial-time separation oracle [27]. For presentation convenience, we add that restriction and it will suffice for our analysis.

$i \in I$ and a given $S \subseteq \mathcal{N}_i$. Observe that $\sum_{j \in S} X_{ij} = 1$ denotes the random event that i is matched with one of its neighbors in S, whose probability should be no larger than that of at least one neighbor in S arrives at least once during the online T rounds. Thus,

$$\begin{aligned}\sum_{j \in S} x_{ij} &= \mathsf{E}\left[\sum_{j \in S} X_{ij}\right] = \Pr\left[\sum_{j \in S} X_{ij} = 1\right] \\ &\leq 1 - \Pr[\text{none of neighbors in } S \text{ arrives during the } T \text{ rounds}] \\ &= 1 - \left(1 - \frac{|S|}{T}\right)^T = 1 - \mathrm{e}^{-|S|} + O(1/T).\end{aligned}$$

The last equality is due to our assumption that $|S|$ is upper bounded by a given constant independent of $T \gg 1$. □

2.3 Overview of Our Techniques: LP-Based Sampling, Boosting, and Attenuation

Before describing our techniques, we first explain the standard LP-based sampling approach, which has been commonly used in algorithm design for various online-matching models under known distributions, either KIID [14] or known adversarial distributions [2] (where arrival distributions are still independent but not necessarily identical). A typical framework is as follows: One uses the benchmark LP to upper bound the performance of a clairvoyant algorithm, and then solves the LP to get statistics regarding how to match an online agent with its offline neighbors. After that, one uses these statistics to guide actions in the online policy. For example, suppose that, by solving the LP, we know that the clairvoyant optimal will match each pair of offline-online agents (i, j) with probability x_{ij}. Observe that, by Constraint (4) in the benchmark LP, we have $\sum_{i \in \mathcal{N}_j} x_{ij} \leq r_j = 1$. Thus, we can then transfer it to a simple non-adaptive matching policy as follows: Upon the arrival of online agent j, sample a neighbor $i \in \mathcal{N}_j$ with probability x_{ij} and match it if i is available.

A straightforward analysis yields that the aforementioned non-adaptive sampling policy achieves $1 - 1/\mathrm{e}$ of the LP optimum, see, *e.g.,* [9] and [25]. However, such a policy also achieves no better than $1 - 1/\mathrm{e}$, even when the LP has been tightened by Constraint (6).[7] To go beyond the competitive ratio of $1 - 1/\mathrm{e}$, we consider the following simple and natural idea:

Boosting. Suppose we are at (the beginning of) time t and an online agent j arrives. Assume that by solving the benchmark LP, we learn a sampling distribution $\mathbf{x}_j = \{x_{ij} | i \in \mathcal{N}_j\}$ for online agent j with $\sum_{i \in \mathcal{N}_j} x_{ij} \leq 1$ from the clairvoyant optimal. Let $\mathcal{N}_{j,t} \subseteq \mathcal{N}_j$ be the set of *available* neighbors of j at time t. Instead of non-adaptively following the same distribution $\mathbf{x}_j$ over $\mathcal{N}_j$ throughout the online phase [15, 54], we try to sample a neighbor j from $\mathcal{N}_{j,t}$ only following a boosted version of distribution $\mathbf{x}'_j = \{x_{ij} / \sum_{i \in \mathcal{N}_{j,t}} x_{ij}\}$. In this way, we promote the chance of each available neighbor of j at t getting matched with j. Also, we can guarantee that the offline neighbor we have sampled is available at the time and dismiss the case that we sample an unavailable neighbor and have to reject j ultimately, which is also a desirable feature to have in practice. This is the key idea in the algorithm we will present for IFM (see Theorem 2).

Note that our boosting idea has already been proposed and tested in several practical crowdsourcing applications [13, 14]. Though it proved to be helpful in some scenarios, the boosted version of LP-based sampling is more challenging to analyze since one has to consider the adaptive behavior of the algorithm. We are the first to show that it achieves a competitive ratio exceeding

[7] To see this, consider a complete bipartite graph with T nodes on each side. Setting $x_{ij} = 1/T$ for every edge (i, j) is feasible in the tightened LP that leaves every offline node fractionally matched. However, the corresponding sampling policy would only leave each offline node matched with probability $1 - 1/\mathrm{e}$. In fact, it can be seen on this example that any non-adaptive sampling policy must leave some offline node matched with probability at most $1 - 1/\mathrm{e}$.

$1 - 1/\mathrm{e}$, of 0.725, for IFM. One can contrast this to [39] who studied unweighted online matching under KIID and also introduced a boosting strategy, which is more complex—their main idea is to generate two *negatively* correlated sampling distributions from the original one, and sample two candidate neighbors for an online arriving agent.

Uniform vs. Non-Uniform Boosting. A critical element in our analysis of the simple boosting algorithm for IFM, however, is that an optimal LP solution places identical total mass x_i on each offline agent i. By contrast, for GFM, an optimal LP solution may place higher mass on specific offline agents, *e.g.,* those belonging in many groups. We show that when there is heterogeneity in the mass placed across offline agents, the boosting algorithm matches high-mass offline agents i with probability no better than $(1 - 1/\mathrm{e}) \cdot x_i$ (see Lemma 4), *i.e.,* its competitive ratio achieved is no longer better than $1 - 1/\mathrm{e}$.

To break this barrier of $1 - 1/\mathrm{e}$ for GFM, our idea is to use boosting in a *non-uniform* fashion. In the simple boosting algorithm above, for an incoming arrival, the probability of sampling unavailable neighbors was re-distributed proportionally across its available neighbors. However, our algorithm for GFM is more likely to suffer from offline agents i with high total mass x_i, since as explained above those agents are more difficult to match relative to its total mass x_i.

To accomplish this, we add an *attenuation factor* to offline agents with small total mass in the LP solution, in an attempt to *balance* the probability of matching every offline agent i relative to its total mass x_i. Most similar to ours are the attenuation techniques used in [9] and [28] to overcome the barrier of $1-1/\mathrm{e}$ for VOM under KIID. However, our attenuation technique is different and new in the following sense. Consider an online agent j that arrives at time t and let $\mathbf{x}_j = \{x_{ij} | i \in \mathcal{N}_j\}$ be the sampling distribution of j before attenuation. The idea in [9] and [28] is to carefully design factors α_{ij} that are added directly to the sampling distribution $\mathbf{x}_j$ such that we finally sample an offline neighbor $i \in \mathcal{N}_j$ with an updated probability $\alpha_{ij} \cdot x_{ij}$ upon the arrival of j. In this way, they can both promote the performance of an offline agent i with large mass by setting $\alpha_{ij} > 1$ and compress that of i with small mass by setting $\alpha_{ij} < 1$. In contrast, our idea is to adaptively and randomly update the set of neighbors of j subject to sampling. Recall that the idea of boosting is to sample a neighbor of j only from the set of available neighbors at that time following a boosted version of distribution $\mathbf{x}'_j = \{x_{ij} / \sum_{i \in \mathcal{N}_{j,t}} x_{ij}\}$. Our attenuation idea is to further enhance the power of boosting by randomly "muting" some available neighbors of j at time t with small mass (*i.e.,* forcefully labeling them as unavailable) and then apply the boosting idea to the set of all available neighbors that survives the muting procedure. As a result, our attenuation helps compress the performance of offline agents with small mass and promote that of offline agents with large mass.

Overview of Simulation-Based Attenuation. The term "simulation" throughout this article refers to the classical Monte Carlo simulation. Simulation-based attenuation is a powerful technique in algorithm design for stochastic optimization problems; see, *e.g.,* stochastic knapsack [35], stochastic matching [1, 8], and matching policy design in rideshare [16, 20]. The high-level idea is as follows: Suppose we have a random event EV that occurs with an unknown probability $\Pr[\mathrm{EV}] = p > c$, where c is a target we aim to attenuation EV to. A typical approach is as follows: We first apply Monte Carlo simulations to get an estimate $\bar{p}$ such that $\bar{p} \in [(1-\epsilon)p, (1+\epsilon)p]$ with probability at least $1 - \delta$. Then, by simply ignoring event EV with probability $1 - c/\bar{p}$ (regardless if EV happens), we can make that event EV will "occur" with probability $cp/\bar{p} \in [c/(1+\epsilon), c/(1-\epsilon)]$. The number of samples needed for an estimate as shown above is $\Theta(1/(c\epsilon^2) \cdot \log(1/\delta))$ by applying a standard Chernoff bound. In our context, a random event subject to attenuation is typically that an offline agent is matched by some time, and the target c takes a constant value between $[1/\mathrm{e}, 1]$ (see Section 5.2). We will take $\epsilon = 1/\mathrm{poly}(N)$, where N is the problem size such that the error ϵ will bring lower-order terms in the final competitive ratios. A detailed discussion on simulation-based attenuation can be seen Appendix B in [8].

2.4 Price of Fairness

*Price of Fairness (*POF*)* is a common question when fairness promotion is considered together with another potentially conflicting objective such as utility (a.k.a. system efficiency) maximization [4, 5, 12]. In our context, IFM (1) and GFM (2) each can be viewed as a fairness-related objective, while VOM can be treated as a utility-oriented objective. Let u_f be a fairness metric (such as IFM or GFM) and μ be a utility metric (such as VOM). Consider a given instance $\mathcal{I}$ of online matching under KIID. Let $\mathcal{X}(\mathcal{I})$ be the collection of all solutions feasible to Constraints (4), (5), and (6) related to instance $\mathcal{I}$. We define the POFon $\mathcal{I}$ and in general as follows:

$$\mathrm{POF}(\mu_f, \mu, \mathcal{I}) = 1 - \frac{\max_{\mathbf{x} \in \mathrm{argmax}_{\mathcal{X}(\mathcal{I})} \mu_f} \mu(\mathbf{x})}{\max_{\mathbf{x} \in \mathcal{X}(\mathcal{I})} \mu(\mathbf{x})}, \tag{7}$$

$$\mathrm{POF}(\mu_f, \mu) = \sup_{\mathcal{I}} \mathrm{POF}(\mu_f, \mu, \mathcal{I}), \tag{8}$$

where the supremum in (8) is taken over all possible feasible instances $\mathcal{I}$s.

Remarks on the Two Definitions Above. (1) For any fairness and utility metrics μ_f and μ, we have $\mathrm{POF}(\mu_f, \mu) \in [0, 1]$. Also, in our context, $\mathrm{POF}(\mathrm{IFM}, \mu) \le \mathrm{POF}(\mathrm{GFM}, \mu)$, since any feasible instance of IFM can be cast as special case of GFM with each group consisting of one single offline agent. (2) POF in (7) captures the relative loss in utility under μ between an optimal solution maximizing μ and another one that maximizes the fairness μ_f and also makes the utility μ as large as possible meanwhile. The definition of POF in (7) can be viewed as an optimistic version: We compute the *least* possible loss in the utility by comparing against an optimal solution under the fairness metric that is most favorable to the utility. We choose the optimistic version instead of pessimistic (*i.e.,* replacing max with min in the numerator) or general (choosing an arbitrary optimal solution maximizing fairness) just to make the task of computing POF in (8) slightly more challenging. In fact, we can otherwise easily come up with an instance $\mathcal{I}$ with $\mathrm{POF}(\mu_f, \mu, \mathcal{I}) = 1$ for μ_f being either IFM or GFM and μ being VOM; see the example below.

Example 1. Consider an unweighted bipartite graph made up of two disjoint parts: The first is a star graph with n offline agents connected to one single online agent; the second part is a perfect matching consisting of n disjoint edges. Thus, we have in total $2n$ offline agents and $n+1$ online agents. For this instance, we can verify that (1) the optimal value under VOM is $(1-1/\mathrm{e}) \cdot n + 1$; (2) among all possible optimal solutions under IFM, the largest and lowest possible utility values that could be ever achieved on VOM are $(1-1/\mathrm{e}) \cdot n + 1$ and 2, respectively. Thus, with μ_f and μ being IFM and VOM, respectively, we have that under the "pessimistic" version of POF,

$$1 - \frac{\min_{\mathbf{x} \in \mathrm{argmax}_{\mathcal{X}(\mathcal{I})} \mu_f} \mu(\mathbf{x})}{\max_{\mathbf{x} \in \mathcal{X}(\mathcal{I})} \mu(\mathbf{x})} = 1 - \frac{2}{(1-1/\mathrm{e}) \cdot n + 1},$$

which leads to $\mathrm{POF}(\mathrm{IFM}, \mathrm{VOM}) = 1 \le \mathrm{POF}(\mathrm{GFM}, \mathrm{VOM}) = 1$.

2.5 Main Contributions

In this article, we propose two generic online-matching based models to study individual and group fairness maximization among offline agents in OMMs. Our main results are summarized as follows:

Online Algorithm Design and Competitive Analysis for IFM and GFM. For IFM and GFM, we present an LP-based sampling with boosting (SAMP-B) and another sampling algorithm with boosting and attenuation (SAMP-AB), respectively.

Theorem 1. *For* IFM *and* GFM*, both* Greedy *and* Ranking *achieve a competitive ratio of 0.*

Table 1. Competitive Ratio of $\eta(\tau)$ Achieved by SAMP-B for IFM, Displayed for Values of τ where $\tau = 1 - \mathrm{e}^{-k}$ for Some $k = 0, 1, 2, \ldots$

k	0	1	2	3	4	5	6	7	8	∞
$\eta(1 - \mathrm{e}^{-k})$	1	0.7970	0.7493	0.7339	0.7285	0.7266	0.7258	0.7256	0.7255	0.7254

We note that $\eta(\tau)$ is a decreasing function of τ, so if τ takes a value, *e.g.*, 0.5, which lies between $1 - \mathrm{e}^{-0} = 1$ and $1 - \mathrm{e}^{-1} \sim 0.632$, then the competitive ratio will be between 1 and 0.7970.

Remarks on Theorem 1. We emphasize that Theorem 1 requires a non-trivial analysis and makes a significant statement. Indeed, it is *a priori* unclear why Greedy (with randomized tiebreaking) or Ranking should be so poor for fairness maximization, when randomization is built into both of them. This contrasts with facts that Greedy achieves a ratio *equal to* $1 - 1/\mathrm{e}$ for vertex-weighted online matching under KIID [23] and Ranking achieves $1 - 1/\mathrm{e}$ for unweighted online matching even under adversarial [31]. *This shows that* IFM *and* GFM *are new, and distinct online matching problems in which the baseline algorithms of* Greedy *and* Ranking *do not work.*

Theorem 2. *A simple* LP*-based sampling algorithm with boosting (*SAMP-B*) achieves a competitive ratio at least* $\eta(\tau) \geq \eta(1) \sim 0.7254$ *for* IFM*, where* τ *is the optimal value to* LP (1)*, and* $\eta(\tau)$ *is an efficiently computable value defined later in Lemma 2.*

Theorem 3. *An LP-based sampling algorithm with attenuation and boosting (*SAMP-AB*) achieves a competitive ratio at least* 0.719 *for* GFM *and* VOM.

Remarks on Theorems 2 and 3. (1) We numerically compute a few values of $\eta(\tau)$ via Mathematica;[8] see Table 1, where $\eta(0)$ is obtained by taking $\tau \to 0_{+}$. Note that neither of the algorithm SAMP-B nor SAMP-AB takes the optimal value τ as part of the input. Results suggest that SAMP-B can perform far beyond $1 - 1/\mathrm{e}$ when the benchmark LP value τ is small, which is common in practice when offline agents are outnumbered by online ones (*e.g.*, ride-hailing during off-peak hours with drivers being more than riders). The dependence of SAMP-B's performance on the benchmark LP value offers another evidence that IFM essentially differs from VOM, where we are unaware of any algorithm whose performance can adapt to the benchmark LP value. (2) As for the hardness side of IFM or GFM: There is an upper bound of 0.865 due to [39] for unweighted KIID with integral arrival rates, which can be modified to hold for IFM and GFM. We acknowledge that for VOM under KIID with integral arrival rates, the guarantee of 0.719 implied by our SAMP-AB algorithm does not improve the two state-of-the-art algorithms, which achieve ratios of 0.725 [28] and 0.729 [9], respectively. However, we believe that the algorithms analyzed in this article are much more natural than those from the aforementioned articles, in particular our simple boosting algorithm SAMP-B, which has been previously suggested but for which no non-trivial competitive ratio guarantees were previously known. Moreover, as displayed in Table 1, for most values of parameter τ (more precisely, any $\tau \leq 1 - \mathrm{e}^{-3} \sim 0.95$), our guarantees are actually higher than any known guarantees for VOM.

Comparison of Optimal Competitive Ratios on IFM, GFM, and VOM. We show the fairness maximization problems to be no harder than VOM in terms of the optimal competitive ratio. Indeed, for a given model, let $\Phi(\cdot)$ denote the optimal competitive ratio that an online algorithm can achieve in the worst case. We establish the following:

Theorem 4. $\Phi(\text{IFM}) \geq \Phi(\text{GFM}) \geq \Phi(\text{VOM})$.

[8]Throughout this article, all numerical computations are done via Mathematica 12.3.1.0 on PC Mac OS X x86 (64-bits) with 16-GM memory. We mainly use numerical functions offered by Mathematica.

Of course, this result is on the theoretically optimal online algorithms, which cannot be computed. The fairness maximization problems are computationally distinctive from VOM.

Price of Fairness with Respect to IFM and VOM. In this article, we study Price of Fairness with respect to the individual fairness IFM and two utility metrics, a weighted version VOM and an unweighted version VOM′ (in this case, all offline agents have a uniform weight).

Theorem 5. $\mathrm{POF}(\mathrm{IFM}, \mathrm{VOM}) = 1$ *and* $\mathrm{POF}(\mathrm{IFM}, \mathrm{VOM}') = 0$.

The result above suggests that there is substantial difference in the relative loss imposed by IFM between weighted and unweighed VOMs.

Empirical Evaluations on Real Datasets. We show that the algorithms analyzed in this article also perform better in simulations. We compare our algorithms against Greedy, Ranking, and these state-of-the-art algorithms in the literature [9, 28, 39] in simulations based on real data. Our datasets include a public ride-hailing dataset collected from the city of Chicago, from which we construct instances for IFM and GFM, as well as four datasets from the Network Data Repository [46], from which we test the classical problem VOM. For IFM and GFM, simulation results show that our algorithm SAMP-B, as well as the algorithm from [39] (after being adapted to our problem), significantly outperform others. For VOM, simulation results show that SAMP-B, along with Greedy, significantly outperform others. This demonstrates that among algorithms which appear to perform well in practice (boosting, Greedy, [39]), our simple boosting algorithm achieves the best guarantee, and importantly, simultaneously performs well both for fairness maximization and for weight maximization (whereas [39] only performs well for IFM/GFM while Greedy only performs well for VOM). Our simple boosting idea is also much simpler to implement than the cleverly correlated sampling of [39].

Roadmap. In Section 4, we present the algorithm SAMP-B for IFM and prove Theorem 2; in Section 5, we present the algorithm SAMP-AB for IFM and VOM and prove Theorem 3; in Section 6, we prove Theorems 1, 4, and 5; and in Section 7, we present details regarding our real datasets and relevant experimental results.

3 OTHER RELATED WORKS

In recent years, online-matching-based models have seen wide applications ranging from blood donation [40] to volunteer crowdsourcing [38] and from kidney exchange [33] to rideshare [16]. Here we briefly discuss a few studies that investigate the fairness issue. Both works of [50] and [32] have studied the income inequality among rideshare drivers. However, they mainly considered a complete offline setting where the information of all agents in the system including drivers and riders is known in advance. They justified that by focusing a short window and thus, all agents can be assumed offline. There are several other works that considered fairness in matching in an offline setting where all agents' information is given as part of the input, see, *e.g.*, [21] and [48]. Nanda et al. [43] proposed a bi-objective online-matching-based model to study the tradeoff between the system efficiency (profit) and the fairness among rideshare riders during high-demand hours. In contrast, Xu and Xu [53] presented a similar model to examine the tradeoff between the system efficiency and the income equality among rideshare drivers. Unlike focusing on one single objective of fairness maximization like here, both studies in [43] and [53] seek to balance the objective of fairness maximization with that of profit maximization. Recently, Ma et al. [36] considered a similar problem to ours but focus on the fairness among online agents. Manshadi et al. [37] studied fair online rationing such that each arriving agent can receive a fair share of resources proportional to its demand. The fairness issue has been studied in other domains/applications as well, see, *e.g.*, online selection of candidates [47], influence maximization [52], bandit-based online learning [22, 30, 44], online resource allocation [3, 49], and classification [17].

4 INDIVIDUAL FAIRNESS MAXIMIZATION (IFM)

Let $\mathcal{N}_{j,t}$ denote the set of available neighbors of j at t. For the ease of notation, we use $\{x_{ij}\}$ to denote an optimal solution to LP (1) when the context is clear. Let $x_i \doteq \sum_{j \sim i} x_{ij}$ for each $i \in I$. Assume w.l.o.g. that $x_i = \tau$ for all $i \in I$, where $\tau \in [0, 1]$ is the optimal value of LP (1).[9] Our LP-based sampling with boosting is formally stated as follows:

ALGORITHM 1: Sampling with Boosting (SAMP-B).

```
Offline Phase:
Solve LP (1) and let {x_ij} be an optimal solution.
Online Phase:
for t = 1, ..., T do
    Let an online agent of type j arrive at (the beginning of) time t.
    Let N_{j,t} = {i ∈ N_j, i is available at t}, i.e., the set of available neighbors of j at t.
    if N_{j,t} = ∅ then
        Reject j.
    else
        Sample a neighbor i ∈ N_{j,t} with probability x_ij / Σ_{i' ∈ N_{j,t}} x_{i',j}.
```

An Auxiliary Balls-and-Bins Model for Analysis Purposes. We present a virtual model that re-interprets the matching process in SAMP-B as follows: Consider a given time $t \in [T]$, and let $I_t \subseteq I$ be the set of available (or unmatched) offline agents at (the beginning of) time t with $I_1 = I$. We can view each offline agent $i \in I_t$ as a bin and each edge $e = (i, j)$ with $i \in I_t$ as a ball; at time t, a ball $e = (i, j)$ will be sampled from the pool $\{e = (i, j) : i \in I_t\}$ with probability $(1/T) \cdot (x_{ij} / \sum_{i' : i' \sim j, i' \in I_t} x_{i',j})$ (where $1/T$ is the probability of drawing type j) and it will land in bin i and as a result, i becomes occupied (we also call it *unavailable* or *matched*). This model will prove useful later in analyzing SAMP-B.

4.1 Proof of the Main Theorem 2

For an offline agent $i \in I$, let $Z_i = 1$ indicate that i is matched in the end in SAMP-B. The key idea is to show the lemma below:******

LEMMA 2. *For any value of $\tau \in [0, 1)$, let ℓ_τ denote the largest integer satisfying $1 - \mathrm{e}^{-\ell_\tau} \leq \tau$. Define*

$$\eta(\tau) := \frac{1}{\tau}\left(1 - \exp\left(-\frac{g_\tau(1 - \mathrm{e}^{-1}) + g_\tau(\mathrm{e}^{-1} - \mathrm{e}^{-2}) + \cdots + g_\tau(\tau - (1 - \mathrm{e}^{-\ell_\tau}))}{\tau}\right)\right)$$

for all $\tau \in [0, 1]$, with $\eta(1) := \lim_{\tau \to 1^-} \eta(\tau)$. Then, $\mathsf{E}[Z_i] \geq \tau \cdot \eta(\tau)$ for all $i \in I$.

It is easy to numerically verify that $\eta(\tau)$ is a decreasing function over [0,1], with a limiting value of $\eta(1) \geq 0.725$. Therefore, Lemma 2 suggests that each offline agent gets matched in SAMP-B with probability at least $0.725 \cdot \tau$. Since $\min_i \mathsf{E}[Z_i] \geq 0.725 \cdot \tau \geq 0.725 \cdot \mathrm{OPT}$ by Lemma 1, where OPT denotes the performance of a clairvoyant optimal, Lemma 2 would establish the main Theorem 2.

We now proceed to prove Lemma 2. For each offline agent $i \in I$ and $t \in [T]$, let $\chi_{i,t} = 1$ indicate that i is available at (the beginning of) t in SAMP-B, and $q_{i,t}$ be the probability that i is matched during round t conditioning on i is available at (the beginning of) t, *i.e.*, $q_{i,t} = \Pr[\chi_{i,t+1} = 0 | \chi_{i,t} = 1]$. Recall that $Z_i = 1$ indicate that i is matched in the end in SAMP-B. Thus,

[9]We can always make it by decreasing all $\{x_{ij} | j \sim i\}$ for those i with $x_i > \tau$ without affecting the optimal LP value.

$$\mathsf{E}[Z_i] = 1 - \prod_{t=1}^{T}(1 - q_{i,t}). \tag{9}$$

Recall that in the optimal solution, we have $x_i = \tau$ for all $i \in I$.

LEMMA 3. *For any i and i' with $i \neq i'$ and any time t, we have* $\Pr[\chi_{i',t} = 1|\chi_{i,t} = 1] \leq (1-\tau/T)^{t-1}$.

Lemma 3 suggests pairwise negative correlations among offline agents getting matched at any time t. This plays a vital role in helping SAMP-B overcome the barrier of $1 - 1/\mathrm{e}$, which can be seen as follows: The counterpart Lemma 5 serves a similar role in SAMP-AB.

Consider an instance such that $x_i = x_j = 1$ for all i and j.[10] Consider a given $i \in I$. For an offline agent $i' \neq i$, we call i' an *offline* neighbor of i if the two share at least an (online) neighbor j with $j \sim i$ and $j \sim i'$. According to SAMP-B, i will be matched at $t = 1$ with probability equal to $q_1 := \sum_{j \in i}(1/T) \cdot x_{ij}/\sum_{i' \in \mathcal{N}_{j,t}} x_{i',j} = 1/T$ since $\mathcal{N}_{j,t=1} = \mathcal{N}_j$ (all neighbors of j is available at $t = 1$) and thus, $\sum_{i' \in \mathcal{N}_{j,t}} x_{i',j} = x_j = 1$ and $q_1 = \sum_{j \sim i}(1/T) \cdot x_{ij} = x_i/T = 1/T$. Note that $\cup_{j \sim i}\mathcal{N}_{j,t}$ captures the exact set of all available *offline* neighbors of i at t (except that it can possibly include i itself).

Assume i survives during $t = 1$, *i.e.,* it is not matched during $t = 1$ with $\chi_{i,2} = 1$. Under the absence of pairwise negative correlations among offline agents, it is possible that all offline neighbors of i remain available at $t = 2$ conditioning on i's availability, which suggests that $\mathcal{N}_{j,t=2} = \mathcal{N}_{j,t=1} = \mathcal{N}_j$ for each $j \sim i$. Applying the same analysis of i during $t = 1$ to $t = 2$, we get that i will be matched during $t = 2$ with probability equal to $1/T$. Continuing this reasoning for all $t \in [T]$, we claim that i will be matched with probability equal to $1-(1-1/T)^T \sim 1-1/\mathrm{e}$ in the end. The pairwise negative correlation among offline agents shown in Lemma 3 can greatly improve the analyses above: i' availability at $t = 2$ suggests that every offline neighbor i' of i sharing some online neighbor j^* will survive at $t = 2$ with probability no more than the unconditional case, which is equal to $\mathsf{E}[\chi_{i',2}] = 1-1/T$. This helps strictly decrease the value $\mathsf{E}[\sum_{i' \in \mathcal{N}_{j^*,t=2}} x_{i',j^*}|\chi_{i,2} = 1]$ and boost the matching probability for i when $j^* \sim i$ arrives at $t = 2$.

We now prove Lemma 2 assuming Lemma 3 is true, and prove Lemma 3 afterward.

PROOF OF LEMMA 2. Focus on a given offline agent i^*. For the ease of notation, we drop the subscription of i^*, and use q_t, χ_t, and Z to denote the corresponding values with respect to i^*.

Now, we try to lower bound the value of q_t. Consider a given t. For each $i \neq i^*$, recall that $\chi_{i,t} = 1$ indicate that i is available at t. By the nature of SAMP-B, we see that conditioning on i^* is available at t ($\chi_t = 1$), i^* will be matched during t iff one of its neighbors $j \sim i^*$ arrives and (i^*, j) gets sampled. Recall that for each $j \sim i^*$, $\mathcal{N}_{j,t}$ denotes the set of available neighbors incident to j at t, and $X_{j,t} = \sum_{i \in \mathcal{N}_{j,t}} x_{ij}$. Observe that

$$q_t = \mathsf{E}\left[\frac{1}{T}\sum_{j \sim i^*}\frac{x_{i^*,j}}{X_{j,t}}\middle|\chi_t = 1\right] \geq \frac{1}{T}\sum_{j \sim i^*}\frac{x_{i^*,j}}{\mathsf{E}[X_{j,t}|\chi_t = 1]}. \tag{10}$$

The last inequality above is due to Jensen's inequality and the convexity of function $1/x$. Note that

$$\begin{aligned}
\mathsf{E}[X_{j,t}|\chi_t = 1] = \mathsf{E}\left[\sum_{i \in \mathcal{N}_{j,t}} x_{ij}|\chi_t = 1\right] &= x_{i^*,j} + \sum_{i \neq i^*, i \sim j} x_{i,j} \cdot \mathsf{E}[\chi_{i,t}|\chi_t = 1] \\
&\leq x_{i^*,j} + \sum_{i \neq i^*, i \sim j} x_{i,j} \cdot (1 - \tau/T)^{t-1}. \text{ (By Lemma 3)} \\
&\leq x_{i^*,j} + (1 - x_{i^*,j}) \cdot (1 - \tau/T)^{t-1}. \text{ (Due to Constraint (4) of LP (1))}
\end{aligned}$$

[10] An example can be seen as a complete bipartite graph with $|I| = |J| = n = T \gg 1$, and we can verify that an optimal solution to LP (1) is that all edges take a value of $1/n$ such that $x_i = x_j = 1$ for all $i \in I$ and $j \in J$.

Substituting the above inequality to Inequality (10), we have

$$q_t \geq \frac{1}{T} \sum_{j \sim i^*} \frac{x_{i^*,j}}{x_{i^*,j} + (1 - x_{i^*,j}) \cdot (1 - \tau/T)^{t-1}}.$$

Plugging the above results into Equation (9), we have

$$\begin{aligned}
\mathsf{E}[Z] &= 1 - \prod_{t=1}^{T} (1 - q_t) \\
&\geq 1 - \prod_{t=1}^{T} \left(1 - \frac{1}{T} \sum_{j \sim i^*} \frac{x_{i^*,j}}{x_{i^*,j} + (1 - x_{i^*,j}) \cdot (1 - \tau/T)^{t-1}} \right) \\
&\geq 1 - \exp\left(- \sum_{t=1}^{T} \frac{1}{T} \sum_{j \sim i^*} \frac{x_{i^*,j}}{x_{i^*,j} + (1 - x_{i^*,j}) \cdot (1 - \tau/T)^{t-1}} \right) \\
&= 1 - \exp\left(- \sum_{j \sim i^*} \sum_{t=1}^{T} \frac{1}{T} \frac{x_{i^*,j}}{x_{i^*,j} + (1 - x_{i^*,j}) \cdot (1 - \tau/T)^{t-1}} \right) \\
&= 1 - \exp\left(- \sum_{j \sim i^*} \int_0^1 d\zeta \frac{x_{i^*,j}}{x_{i^*,j} + (1 - x_{i^*,j}) \cdot \mathrm{e}^{-\tau \cdot \zeta}} \right) \text{(Taking } T \to \infty) \\
&= 1 - \exp\left(- \frac{1}{\tau} \sum_{j \sim i^*} \ln(1 + x_{i^*,j} \cdot (\mathrm{e}^{\tau} - 1)) \right).
\end{aligned}$$

The second inequality above is due to the fact that $1 - x \leq e^{-x}$.

Let $g_\tau(x) = \ln(1 + x(\mathrm{e}^\tau - 1))$, where $\tau \in [0, 1]$ is a parameter. To get a lower bound for $\mathsf{E}[Z]$, we need to solve the below minimization program. For the ease of notation, we omit the subscription of i^* and use $x_j \doteq x_{i^*,j}$.

$$\left\{ \min \sum_{j \sim i^*} g_\tau(x_j) : \sum_{j \sim i^*} x_j = \tau; \sum_{j \in S} x_j \leq 1 - \mathrm{e}^{-|S|}, \forall S \subseteq \mathcal{N}_{i^*}, |S| = O(1). \right\} \tag{11}$$

Note that the first constraint is due to our assumption $x_{i^*} = \tau$; the rest is due to Constraint (6) in the benchmark LP. Function g_τ is concave over (0,1), and hence the minimization problem (11) must have an extreme point optimal solution. We claim that for any τ the only extreme points to the feasible region in (11) are described by $x_{j_1} = 1 - \mathrm{e}^{-1}, x_{j_2} = \mathrm{e}^{-1} - \mathrm{e}^{-2}, \ldots, x_{j_{\ell_\tau}} = \tau - (1 - \mathrm{e}^{-\ell_\tau})$ and $x_j = 0$ for all other indices j, where ℓ_τ is the largest integer satisfying $1 - \mathrm{e}^{-\ell_\tau} \leq \tau$, and $j_1, \ldots, j_{\ell_\tau}$ are some indices in $\mathcal{N}_{i^*}$. Since the objective $\sum_{j \sim i^*} g_\tau(x_j)$ is symmetric over $\{x_j : j \in \mathcal{N}_{i^*}\}$, assuming this claim is true, all extreme points must have the same objective value of

$$g_\tau(1 - \mathrm{e}^{-1}) + g_\tau(\mathrm{e}^{-1} - \mathrm{e}^{-2}) + \cdots + g_\tau(\tau - (1 - \mathrm{e}^{-\ell_\tau})) \tag{12}$$

(because $g_\tau(0) = 0$). This would show that the optimal value of optimization problem (11) equals (12).

We now prove this claim, by characterizing the extreme points of the feasible region to optimization problem (11). Take any feasible solution $(x_j)_{j \in \mathcal{N}_{i^*}}$, and let $m := |\mathcal{N}_{i^*}|$ and re-index the coordinates so that $x_1 \geq x_2 \geq \cdots \geq x_m$. We show that if $(x_j)_{j \in [m]}$ does not satisfy

$$x_1 = 1 - \mathrm{e}^{-1}, x_2 = \mathrm{e}^{-1} - \mathrm{e}^{-2}, \ldots, x_{\ell_\tau} = \tau - (1 - \mathrm{e}^{-\ell_\tau}), x_{\ell_\tau + 1} = 0, \ldots, x_m = 0, \tag{13}$$

then it is not an extreme point. Indeed, for $(x_j)_{j\in[m]}$ to not satisfy (13), there must be a smallest coordinate j such that x_j is strictly less than the value $e^{-(j-1)} - e^{-j}$ prescribed in (13). Due to the constraint $\sum_{j=1}^{m} x_j = \tau$, we must still have that x_j is non-negative; otherwise $x_j = x_{j+1} = \cdots = x_m = 0$ and $\sum_{j=1}^{m} x_j = \tau$ can never be satisfied. Moreover, it must be the case that $0 < x_{j+1} \le x_j < e^{-(j-1)} - e^{-j}$. Since we have that both x_j, x_{j+1} lie in $(0, e^{-(j-1)} - e^{-j})$, it is clear that we can replace x_j, x_{j+1} with either $x_j + \epsilon, x_{j+1} - \epsilon$ or $x_j - \epsilon, x_{j+1} + \epsilon$ for some small $\epsilon > 0$ while still satisfying $\sum_{j=1}^{m} x_j = \tau$. This perturbation also preserves overall feasibility in (11) (note that $1 - e^{-1} \ge e^{-1} - e^{-2} \ge \cdots \ge \tau - (1 - e^{-\ell_\tau})$), demonstrating that $(x_j)_{j\in[m]}$ is not an extreme point. Therefore, an optimal solution to (11) takes the form described in (13).

Consequently, we have for the arbitrary offline agent i^* that

$$\frac{\mathsf{E}[Z_{i^*}]}{\tau} \ge \frac{1}{\tau}\left(1 - \exp\left(-\frac{g_\tau(1-e^{-1}) + g_\tau(e^{-1} - e^{-2}) + \cdots + g_\tau(\tau - (1 - e^{-\ell_\tau}))}{\tau}\right)\right) \tag{14}$$

which completes the proof of Lemma 2. □

Proof of Lemma 3. Consider a given time t and let I_t be the set of unmatched offline agents at t. According to the aforementioned balls-and-bins model: each $i \in I_t$ corresponds to a bin and each edge (ij) with $i \in I_t$ corresponds to a ball; at time t, the ball (ij) will arrive with probability $(x_{ij}/X_{j,t}) \cdot (1/T)$ and land in bin i, where $X_{j,t} \doteq \sum_{i' \sim j, i' \in I_t} x_{i',j}$. Observe that for any j and t, we have $X_{j,t} = \sum_{i' \sim j, i' \in I_t} x_{i',j} \le \sum_{i' \sim j} x_{i',j} \le 1$ due to Constraint 4 in LP (1).

The fact $\chi_{i,t} = 1$ suggests that for any round $t' < t$, none of the balls $e = (ij)$ with $j \sim i$ arrives at t'. Consider a given $t' < t$ and a given $i' \ne i$. Assume $\chi_{i,t} = 1$ and i' is not occupied at t'. Then, we see that each ball $e = (i', j')$ with $j' \sim i'$ will arrive and shoot i' with probability at least $(x_{i',j'}/X_{j',t'}) \cdot (1/T) > x_{i',j'}/T$ since $X_{j',t'} \le 1$ for all $j' \sim i'$ and $t' < t$. This implies that the probability that none relevant balls will shoot i' during t' should be at most $1 - \sum_{j' \sim i'} x_{i',j'}/T = 1 - \tau/T$. Here we invoke our assumption that every offline agent i' has $x_{i'} = \sum_{j' \sim i'} x_{i',j'} = \tau$ in the optimal solution. Therefore, we claim that i' will remain unoccupied after $t-1$ rounds with probability at most $(1 - \tau/T)^{t-1}$. □

5 GROUP FAIRNESS MAXIMIZATION AND AGENT-WEIGHTED MATCHING

5.1 Motivation for Attenuation

We first give an example showing that SAMP-B can never beat $1-e^{-1}$ for VOM without attenuation.

Example 2. Consider such a bipartite graph (I, J, E) as follows: Recall that, by KIID assumption with all unit arrival rates, we have $T = n = |J|$. The set of neighbors of j, denoted by $\mathcal{N}_j$, satisfies the property that (1) $|\mathcal{N}_j| = n, \forall j \in J$; (2) $\cap_{j\in J} \mathcal{N}_j = \{i^*\}$. In other words, each j has a set of n neighbors and they are almost disjoint except sharing one single offline agent i^*. Thus, under this setting, we have (1) $|I| = m = n(n-1) + 1$; (2) i^* has J as the set of neighbors and every $i \ne i^*$ has one single neighbor in J. Let $w_{i^*} = 1$ and $w_i = \epsilon^3$ with $\epsilon = 1/n$ for all $i \ne i^*$. We can verify that any clairvoyant optimal will have a performance at least 1 by simply matching any arriving online agent with i^*.

Lemma 4. SAMP-B *can never beat the ratio of* $1-e^{-1}+o(1)$ *on Example 2, where* $o(1)$ *is a vanishing term when* $T \to \infty$.

Proof. Consider i^* and let $Z_{i^*} = 1$ indicate that i^* is matched in SAMP-B in the end. Observe that during every round t, one $j \sim i^*$ will be sampled uniformly with probability $1/n$ and land in one available neighbor $i \in \mathcal{N}_{j,t}$. Let $N_j = |\mathcal{N}_{j,T+1}|$ be the number of available neighbors incident to j surviving in the end, and let $M_j = n - N_j$, which refers to the number of neighbors of j got occupied. Observe that we have $T = n$ online arrivals and every arrival will land in one item in $\mathcal{N}_j$ uniformly over all $j \in J$. This process can be interpreted as a balls-and-bins model where we have n balls and n bins, and thus, $M = \max_j M_j$ can be viewed as the largest bin load. From [42], we see that with probability $1 - 1/n$, the largest bin load is $M = \Theta(\ln n / \ln \ln n) \leq \ln n$ when n is sufficiently large.

Let SF be the event that $M \leq \ln n$. Assume SF occurs. We see that for all $j \sim i^*$ and $t \in [T]$, $X_{j,t} = \sum_{i \in \mathcal{N}_{j,t}} x_{ij} \geq \epsilon(n - M) \geq \epsilon(n - \ln n) = 1 - \ln n/n$. Let $Z_{i^*} = 1$ indicate that i^* is matched in the end. We have

$$\begin{aligned}
\mathsf{E}[Z_{i^*}|\mathsf{SF}] &= 1 - \prod_{t=1}^{T}\left(1 - \sum_{j \sim i^*} \frac{1}{T} \cdot \frac{x_{i^*,j}}{X_{j,t}}\right) \leq 1 - \prod_{t=1}^{T}\left(1 - \sum_{j \sim i^*} \frac{1}{T} \cdot \frac{\epsilon}{1 - \ln n/n}\right) \\
&= 1 - \left(1 - \frac{1}{T} \cdot \frac{1}{1 - \ln n/n}\right)^T \leq 1 - \exp\left(-\frac{1}{1 - \ln n/n} - \frac{1}{T}\frac{1}{(1 - \ln n/n)^2}\right) \\
&\leq 1 - \mathrm{e}^{-1} + O(\ln n/n).
\end{aligned}$$

Therefore, we have that

$$\mathsf{E}[Z_{i^*}] = \mathsf{E}[Z_{i^*}|\mathsf{SF}] \cdot \Pr[\mathsf{SF}] + \mathsf{E}[Z_{i^*}|\neg\mathsf{SF}] \cdot \Pr[\neg\mathsf{SF}] \leq 1 - \mathrm{e}^{-1} + O(\ln n/n).$$

Recall that any clairvoyant optimal has a performance at least 1. For each $i \neq i^*$, let $Z_i = 1$ indicate that i is matched in SAMP-B in the end. Observe that the expected total values obtained by SAMP-B should be at most

$$\sum_{i \neq i^*} w_i \cdot \mathsf{E}[Z_i] + w_{i^*} \cdot \mathsf{E}[Z_{i^*}] \leq \epsilon^3 \cdot n^2 + \mathsf{E}[Z_{i^*}] \leq 1 - \mathrm{e}^{-1} + O(\ln n/n).$$

Thus, the final competitive ratio of SAMP-B on Example 2 should be at most

$$\frac{1 - \mathrm{e}^{-1} + O(\ln n/n)}{1} \leq 1 - \mathrm{e}^{-1} + O(\ln n/n). \qquad \square$$

5.2 An LP-Based Sampling Algorithm with Attenuation and Boosting (SAMP-AB)

For a given offline agent i, we say $i' \in I$ is an *offline neighbor* of i iff there exists one online agent of j such that $j \sim i$ and $j \sim i'$. Let $\mathcal{S}_i$ be the set of offline neighbors of i. Example 2 suggests that when all offline neighbors of i have very small values in the optimal solution, the boosting strategy shown in SAMP-B will have little effect on improving the overall matching probability of i. Observe that for each offline vertices $i \neq i^*$ on Example 2, it will be matched with a probability at least $\mathsf{E}[Z_i] \geq 1 - \mathrm{e}^{-\epsilon} \sim \epsilon = x_i$. In other words, the chance of getting matched for every $i \neq i^*$ in SAMP-B almost matches its contribution in the LP solution. In contrast, the chance that i^* is matched is only a fraction of $1 - \mathrm{e}^{-1}$ of its contribution in the LP solution. These insights motivate us to add appropriate attenuations to those unsaturated offline vertices such that the boosting strategy can work properly for those saturated ones.

Offline-Phase Simulation-Based Attenuation. Let us first introduce two auxiliary states for offline vertices, called *active* and *inactive*, which are slightly different from available (not matched) and unavailable (matched) as shown before. In our attenuation framework, we assume all offline vertices are active at the beginning ($t = 1$). When an active offline agent i is matched, we will label it as inactive. Meanwhile, we need to forcefully mute some active offline agent, label it as inactive, and view it as being virtually matched. Consider the instance on Example 2: In order to make the boosting strategy work for the dominant agent i^*, we have to intentionally label those active non-dominant vertices i as inactive such that the sampling probability of i^* can be effectively promoted when some $j \in \mathcal{N}_i \cap \mathcal{N}_{i*}$ arrives. Note that the transition from being active to inactive is *irreversible*: Once an active offline agent i is labeled as inactive, it will stay on that state permanently.

Here are the details of our simulation-based attenuation. By simulating all online steps of SAMP-AB up to time t, we can get a very sharp estimate of the probability that each i is active at t, say $\alpha_{i,t}$. If $\alpha_{i,t} \leq (1-1/T)^{t-1}$, then no attenuation is needed at t. Otherwise, add an attenuation factor of $(1-1/T)^{t-1}/\alpha_{i,t}$ to agent i at t as follows: If i is active at t, then label i as *inactive* with probability $1-(1-1/T)^{t-1}/\alpha_{i,t}$ and keep it active with probability $(1-1/T)^{t-1}/\alpha_{i,t}$. In this way, we decrease the probability of i being active at t to the target $(1-1/T)^{t-1}$. The above attenuation can be summarized as follows: If i is available at t, then label i as active and inactive with respective probabilities $\beta_{i,t}$ and $1-\beta_{i,t}$, where $\beta_{i,t} = \min(1, (1-1/T)^{t-1}/\alpha_{i,t})$.

Remarks on the Simulation-Based Attenuation Scheme Above. (1) When computing the attenuation factor $\beta_{i,t}$ for i at t, we should simulate all online steps of SAMP-AB up to t that include applying all the attenuation factors as proposed during all the rounds before t. (2) During every round, we apply the corresponding attenuation factor to each active offline agent in an independent way. (3) All attenuation factors can be computed in an *offline* manner, *i.e.,* before the online phase actually starts.

ALGORITHM 2: Sampling with Attenuation and Boosting (SAMP-AB).

1 **Offline Phase**:
/* The offline phase will take as input $\{(I,J,E),\{w_i\},\{r_j\},T\}$, and output $\{\beta_{i,t}\}$, where $\beta_{i,t}$ denotes the attenuation factor applied to an offline agent i during round t. */
2 Solve LP (3) and let $\{x_{ij}\}$ be an optimal solution.
3 *Initialization*: When $t=1$, set $\beta_{i,t}=1$ for all $i \in I$.
4 **for** $t = 2, 3, \ldots, T$ **do**
5   Applying Monte-Carlo method to simulate Step 10 to Step 15 for all the rounds $t' = 1, 2, \ldots, t-1$ of Online Phase, we get a sharp estimate of the probability that each offline agent i is active at (the beginning of) t, say $\alpha_{i,t}$.
6   Set $\beta_{i,t} = \min(1, (1-1/T)^{t-1}/\alpha_{i,t})$.
7 **Online Phase**:
8 *Initialization*: Label all offline vertices *active* at $t=1$.
9 **for** $t = 1, \ldots, T$ **do**
10   *Independently* relabel each *active* offline agent i as active and inactive with respective probabilities $\beta_{i,t}$ and $1-\beta_{i,t}$.
11   Let an online agent of type j arrive at time t. Let $\mathcal{N}_{j,t} = \{i \in \mathcal{N}_j, i \text{ is active at } t\}$, *i.e.,* the set of active neighbors of j at t.
12   **if** $\mathcal{N}_{j,t} = \emptyset$ **then**
13   Reject j.
14   **else**
15   Sample a neighbor $i \in \mathcal{N}_{j,t}$ with probability $x_{ij}/\sum_{i' \in \mathcal{N}_{j,t}} x_{i',j}$ and label i as inactive.

5.3 Proof of Theorem 3

Similar to the proof of Theorem 2, we aim to show that each offline agent i will be matched in SAMP-AB with a probability $\mathsf{E}[Z_i] \geq 0.719 \cdot x_i$, where $Z_i = 1$ indicates that i is matched in SAMP-AB, and $x_i = \sum_{j \sim i} x_{ij}$ is the total mass allocated to i in the optimal LP solution. This will suffice to prove Theorem 3. The argument is as follows: (1) For GFM, we have $\frac{1}{|G|}\sum_{i \in G} \mathsf{E}[Z_i] \geq \frac{0.719}{|G|}\sum_{i \in G} x_i$ for all $G \in \mathcal{G}$. This suggests that SAMP-AB $= \min_{G \in \mathcal{G}} \frac{1}{|G|}\sum_{i \in G} \mathsf{E}[Z_i] \geq 0.719 \cdot$

$\min_{G\in\mathcal{G}} \sum_{i\in G} x_i/|G| \geq 0.719 \cdot \text{OPT}$ due to Lemma 1, where SAMP-AB and OPT refer to the performance of SAMP-AB and a clairvoyant optimal, respectively. (2) For VOM, we have SAMP-AB $= \sum_{i\in I} w_i \cdot \mathsf{E}[Z_i] \geq 0.719 \cdot \sum_{i\in I} w_i \cdot x_i \geq 0.719 \cdot \text{OPT}$.

For each offline agent i, let $\chi'_{i,t} = 1$ and $\chi_{i,t} = 1$ indicate that i is active at t *before* and *after* the attenuation procedure shown in Step 10 prior to the sampling process. Let $\alpha_{i,t} = \mathsf{E}[\chi'_{i,t}]$ and $\gamma_{i,t} = \mathsf{E}[\chi_{i,t}]$. Let $q_{i,t}$ be the probability that i is matched during t conditioning on i is active at t after attenuation, *i.e.*, $q_{i,t} = \Pr[\chi'_{i,t+1} = 0|\chi_{i,t} = 1] = 1 - \mathsf{E}[\chi'_{i,t+1}|\chi_{i,t} = 1]$. According to our attenuation, for all i and t, we have

$$\gamma_{i,t} = \alpha_{i,t} \cdot \beta_{i,t},\ \beta_{i,t} = \min\left(1, (1-1/T)^{t-1}/\alpha_{i,t}\right),\ \alpha_{i,t+1} = \gamma_{i,t} \cdot (1 - q_{i,t}). \tag{15}$$

Observe that $\alpha_{i,1} = \beta_{i,1} = 1$ for all i, and $\gamma_{i,t} \leq (1-1/T)^{t-1}$ for all i and t. Though our definition of $\{\chi_{i,t}\}$ is slightly different than before, Lemma 3 of Section 4 still works here.

Lemma 5. *For any i and i' with $i \neq i'$ and any time t, we have* $\Pr[\chi_{i',t} = 1|\chi_{i,t} = 1] \leq (1-1/T)^{t-1}$.

Proof. The proof of Lemma 3 in Section 4 suggests that $\chi'_{i,t}$ and $\chi'_{i',t}$ are negatively correlated before attenuation. Thus, we have $\Pr[\chi'_{i',t} = 1|\chi'_{i,t} = 1] \leq \Pr[\chi'_{i',t} = 1]$. Observe that attenuation factors are applied independently to all offline vertices. Therefore,

$$\Pr[\chi_{i',t} = 1|\chi_{i,t} = 1] = \beta_{i',t} \cdot \Pr[\chi'_{i',t} = 1|\chi'_{i,t} = 1] \leq \beta_{i',t} \cdot \Pr[\chi'_{i',t} = 1] = \beta_{i',t} \cdot \alpha_{i',t} \leq (1-1/T)^{t-1}.$$

□

Consider a given offline agent i^* with a fixed value of $x_{i^*} \doteq \sum_{j\sim i^*} x_{i^*,j}$. For the ease of notation, we drop the subscription of i^* and use $\chi'_t, \chi_t, \alpha_t, \beta_t, \gamma_t$, and q_t to denote the corresponding values relevant to i^*. Here are a few properties of $\{q_t\}$.

Lemma 6. ***(P1)**: $q_t \leq q_{t+1}, \forall t \geq 1$; **(P2)** $q_t \geq \frac{1}{T} \sum_{j\sim i^*} \frac{x_{i^*,j}}{x_{i^*,j} + (1-x_{i^*,j})\cdot(1-1/T)^{t-1}}, \forall t \geq 1$.*

Proof. Recall that for each $j \sim i^*$, $\mathcal{N}_{j,t}$ denotes the set of *active* neighbors of j at t right after attenuation. Let $X_{j,t} = \sum_{i\in\mathcal{N}_{j,t}} x_{ij} = \sum_{i\sim j} x_{ij} \cdot \chi_{i,t}$. Observe that

$$q_t = \mathsf{E}\left[\frac{1}{T}\sum_{j\sim i^*} \frac{x_{i^*,j}}{X_{j,t}}\middle|\chi_t = 1\right] = \mathsf{E}\left[\frac{1}{T}\sum_{j\sim i^*} \frac{x_{i^*,j}}{\sum_{i\neq i^*, i\sim j} x_{i,j} \cdot \chi_{i,t} + x_{i^*,j}}\middle|\chi_t = 1\right]. \tag{16}$$

Observe that for each given $i \neq i^*$, $\{\chi_{i,t}|t = 1, 2, \ldots, T\}$ will be a non-increasing sequence due to the irreversibility of the transition from active to inactive of i. Therefore, we claim that $\{q_t|t = 1, \ldots, T\}$ is a non-decreasing series. Thus, we prove **(P1)**. From Equation (16), we have

$$q_t \geq \frac{1}{T}\sum_{j\sim i^*} \frac{x_{i^*,j}}{\sum_{i\neq i^*, i\sim j} x_{i,j} \cdot \mathsf{E}[\chi_{i,t}|\chi_t = 1] + x_{i^*,j}} \geq \frac{1}{T}\sum_{j\sim i^*} \frac{x_{i^*,j}}{(1-x_{i^*,j})\cdot(1-1/T)^{t-1} + x_{i^*,j}}.$$

The first inequality is due to Jensen's inequality and convexity of the function $1/x$. The second one follows from Lemma 5 and the fact of $\sum_{i\sim j} x_{ij} \leq 1$ due to Constraint (4). We get **(P2)**. □

(P1) in Lemma 6 suggests that $\{q_t\}$ is a non-decreasing sequence. Let $K \in [T]$ be such a turning point that $q_{K-1} < 1/T$ and $q_K \geq 1/T$.

Lemma 7. *For each $1 < t \leq K$, we have $\beta_t < 1$ and $\gamma_t = (1-1/T)^{t-1}$, and for each $t > K$, $\beta_t = 1$.*

Proof. By **(P1)**, we have $q_1 \leq q_2 \leq \cdots \leq q_{K-1} < 1/T$. Observe that $\alpha_1 = \beta_1 = \gamma_1 = 1$. Now we consider $t = 2$. From Equation 15, we see i^* will be active at $t = 2$ before attenuation with probability $\alpha_2 = \gamma_1 \cdot (1-q_1) > 1 - 1/T$. Thus, $\beta_2 = (1-1/T)/\alpha_2 < 1$ and $\gamma_2 = 1 - 1/T$. Continuing this analysis, we see for each $t = 2, 3, \ldots, K$, $\alpha_t > (1-1/T)^{t-1}$, $\beta_t < 1$, and $\gamma_t = (1-1/T)^{t-1}$. Now consider the case $t = K+1$. We see that

$$\alpha_{K+1} = \gamma_K \cdot (1-q_K) = (1-1/T)^{K-1} \cdot (1-q_K) \leq (1-1/T)^K.$$

Therefore, $\beta_{K+1} = 1$ and $\gamma_{K+1} = \alpha_{K+1}$. Following this analysis, we have $\alpha_t = \gamma_t \leq (1-1/T)^{t-1}$ and $\beta_t = 1$ for all $t \geq K+1$. □

The above lemma implies that we will keep adding a proper attenuation factor $\beta_t < 1$ to the agent i^* for all $1 < t \leq K$, and afterwards, we will essentially add no attenuation to i^*. Let $Z = 1$ indicate that i^* is matched in the end in SAMP-AB.

Lemma 8. $\mathsf{E}[Z] \geq 0.719 \cdot x_{i^*}$.

Proof. Let $Z = Z_a + Z_b$, where $Z_a = 1$ and $Z_b = 1$ indicate that i^* is matched during any round $t < K$ and $t \geq K$, respectively. Let $K/T = \kappa + o(1)$, where $\kappa \in [0,1]$ is a constant and $o(1)$ is a vanishing term when $T \to \infty$. Let $f(p,x) = \frac{x}{x+(1-x)\cdot p}$. We can verify that for any fixed $p \in (0,1]$, $f(p,x)$ is an increasing concave function over $x \in [0,1]$.

Lower Bounding the Value of $\mathsf{E}[Z_a]$. For each $t < K$, let $Z_t = 1$ indicate that i^* is matched during the round of t. Observe that $Z_t = 1$ iff i^* is active at t after attenuation (*i.e.,* $\chi_t = 1$) and i^* is inactive at $t+1$ before attenuation (*i.e.,* $\chi'_{t+1} = 0$). Thus, we have

$$\mathsf{E}[Z_t] = \Pr[(\chi_t = 1) \wedge (\chi'_{t+1} = 0)] = \Pr[\chi_t = 1] \cdot \Pr[\chi'_{t+1} = 0 | \chi_t = 1] = \gamma_t \cdot q_t.$$

Observe that from Lemma 7, we have $\gamma_t = (1-1/T)^{t-1}$ for all $2 < t \leq K$ and it is valid for $t = 1$ as well. Therefore, we have

$$\mathsf{E}[Z_a] = \sum_{1 \leq t < K} \mathsf{E}[Z_t] = \sum_{1 \leq t \leq K} \gamma_t \cdot q_t = \sum_{1 \leq t < K} \left(1 - \frac{1}{T}\right)^{t-1} \cdot q_t \tag{17}$$

$$\geq \sum_{1 \leq t < K} \left(1 - \frac{1}{T}\right)^{t-1} \cdot \frac{1}{T} \sum_{j \sim i^*} \frac{x_{i^*,j}}{x_{i^*,j} + (1 - x_{i^*,j}) \cdot (1-1/T)^{t-1}}. \tag{18}$$

Recall that $f(p,x) = \frac{x}{x+(1-x)\cdot p}$ is an increasing concave function over $x \in [0,1]$. Define $\mathcal{S}(x) = \{x\}$ if $0 \leq x \leq 1 - \mathrm{e}^{-1}$, and $\mathcal{S}(x) = \{1-\mathrm{e}^{-1}, x - (1-\mathrm{e}^{-1})\}$ if $1 - \mathrm{e}^{-1} < x \leq 1 - \mathrm{e}^{-2}$, and $\mathcal{S}(x) = \{1-\mathrm{e}^{-1}, \mathrm{e}^{-1} - \mathrm{e}^{-2}, x - (1-\mathrm{e}^{-2})\}$ if $1 - \mathrm{e}^{-2} < x \leq 1$. Following the same analysis as shown in Lemma 2, we see that

$$\sum_{j \sim i^*} \frac{x_{i^*,j}}{x_{i^*,j} + (1-x_{i^*,j}) \cdot (1-1/T)^{t-1}} \geq \sum_{x \in \mathcal{S}(x_{i^*})} f((1-1/T)^{t-1}, x). \tag{19}$$

Recall that $K/T = \kappa + o(1)$. Plugging Inequality (19) to Inequality (18), we have

$$\mathsf{E}[Z_a] \geq \sum_{1 \leq t < K} \left(1 - \frac{1}{T}\right)^{t-1} \cdot \frac{1}{T} \cdot \sum_{x \in \mathcal{S}(x_{i^*})} f((1-1/T)^{t-1}, x)$$

$$= \sum_{x \in \mathcal{S}(x_{i^*})} \int_0^\kappa d\zeta \cdot \mathrm{e}^{-\zeta} \cdot f(\mathrm{e}^{-\zeta}, x). \ \text{(Taking } T \to \infty).$$

Lower Bounding the Value of $\mathsf{E}[Z_b]$. By definition, $\mathsf{E}[Z_b] = \sum_{K \leq t \leq T} \mathsf{E}[Z_t]$. Observe that i^* will be active at (the beginning of) t after attenuation with probability $\mathsf{E}[\chi_K] = (1-1/T)^{K-1}$. What's more, there will be no attenuation in essence during all $t > K$. Thus, assume i^* is active after attenuation at $t = K$, we can apply almost the same analysis as in Section 4 to lower bound $\mathsf{E}[Z_b]$.

$$
\begin{aligned}
\mathsf{E}[Z_b] &= \sum_{t \geq K} \mathsf{E}[Z_t] = \left(1-\frac{1}{T}\right)^{K-1}\left(1-\prod_{t\geq K}(1-q_t)\right)\\
&\geq \left(1-\frac{1}{T}\right)^{K-1}\left[1-\exp\left(-\sum_{t\geq K}\sum_{j\sim i^*}\frac{1}{T}\cdot\frac{x_{i^*,j}}{x_{i^*,j}+(1-x_{i^*,j})(1-1/T)^{t-1}}\right)\right]\\
&\geq \left(1-\frac{1}{T}\right)^{K-1}\left[1-\exp\left(-\sum_{t\geq K}\frac{1}{T}\cdot\sum_{x\in\mathcal{S}(x_{i^*})} f((1-1/T)^{t-1},x)\right)\right]\\
&= \mathrm{e}^{-\kappa}\left[1-\exp\left(-\sum_{x\sim\mathcal{S}(x_{i^*})}\int_{\kappa}^{1} d\zeta\cdot f(\mathrm{e}^{-\zeta},x)\right)\right] \text{ (Taking } T\to\infty)\\
&= \mathrm{e}^{-\kappa}\left[1-\prod_{x\sim\mathcal{S}(x_{i^*})}\exp\left(-\int_{\kappa}^{1} d\zeta\cdot f(\mathrm{e}^{-\zeta},x)\right)\right].
\end{aligned}
$$

Putting together the lower bounds on $E[Z_a]$ and $E[Z_b]^*$, we have

$$
\mathsf{E}[Z] \geq F(x_{i^*},\kappa) \doteq \sum_{x\in\mathcal{S}(x_{i^*})}\int_0^{\kappa} d\zeta\cdot\mathrm{e}^{-\zeta}\cdot f(\mathrm{e}^{-\zeta},x)+\mathrm{e}^{-\kappa}\left[1-\prod_{x\sim\mathcal{S}(x_{i^*})}\exp\left(-\int_{\kappa}^{1} d\zeta\cdot f(\mathrm{e}^{-\zeta},x)\right)\right].
$$

We can verify via Mathematica that $\min_{0\leq x_{i^*}\leq 1, 0\leq\kappa\leq 1} F(x_{i^*},\kappa)/x_{i^*} \geq 0.719$ and the inequality becomes tight when $x_{i^*} = 1-\mathrm{e}^{-1}$ and $\kappa = 1$. ◻

6 PROOF OF MAIN THEOREMS 1, 4, AND 5

6.1 Proof of Theorem 1

Let us briefly describe Greedy and Ranking here for IFM and GFM. For Greedy, it will always assign an online arriving agent to an offline available neighbor such that the match can improve the current objective of IFM and GFM most; break the tie uniformly at random if any if possible. For Ranking, it will first choose a random permutation π over all offline neighbors and then it will always assign an online arriving agent to an offline available neighbor with the lowest rank in π. Observe that IFM is a special case of GFM when each group consists of one single offline type. Thus, it will suffice to show that Greedy and Ranking achieve a ratio of zero for IFM to prove Theorem 1.

Example 3. Consider such an instance $\mathcal{I}$ of IFM as follows: We have $|I| = |J| = T = n$ offline and online agents. For $j = 1$, it can serve all offline agents, *i.e.,* $\mathcal{N}_{j=1} = I$. For each online agent $j = 2,3,\ldots,n$, it can serve one single offline agent $i = j$. Consider such an offline algorithm ALG (not necessarily a clairvoyant optimal): Try to match each online agent $j \in J$ with $i = j$ if agent j arrives at least once. We can verify that in ALG, each offline agent will be matched with probability at least $1-\mathrm{e}^{-1}$. Thus, we claim that for any clairvoyant optimal, its performance should have OPT $\geq$ ALG $\geq 1-\mathrm{e}^{-1}$.

Lemma 9. Greedy *achieves a competitive ratio of zero for* IFM *Example 3 when $T = n$ both approach infinity.*

Proof. Let K_t be number of unmatched offline agents excluding $i = 1$ at time t. According to Greedy, when $j = 1$ arrives at t, it will match $i = 1$ with a probability $1/(K_t+1)$ if $i = 1$ is not matched then. Observe that $K_t \geq n-1-(t-1) = n-t$, which implies that $1/(K_t+1) \leq 1/(n-t+1)$. Thus, we see that $i = 1$ will be matched at time t with a probability at most $(1/n) \cdot 1/(n-t+1)$ given $i = 1$ is not matched during the first $t-1$ rounds. Thus, $i = 1$ will be matched in Greedy with probability at most $\sum_{t=1}^{n} 1/(n \cdot (n-t+1)) = \Theta(\ln n/n)$. This is in contrast with that OPT $\geq 1 - 1/\mathrm{e}$. Thus, we establish that Greedy achieves a ratio of zero when $n \to \infty$. □

Lemma 10. Ranking *has a competive ratio of zero for* IFM *on Example 3 when $T = n$ both approach infinity.*

Proof. Let $\mathcal{S}$ be the (random) set of indices of offline nodes that fall before $i = 1$ under π. Consider a given $\mathcal{S}$ with $|\mathcal{S}| = K$. For each $j \in J$, let $A_{j,t}$ be the number of arrivals of online agent $j \in J$ before the start of time $t \in [T]$. For each $i \in \mathcal{S}$, let $Z_{i,t} = 1$ indicate that $i \in \mathcal{S}$ is matched by t. Observe that

$$\Pr\left[\sum_{i \in \mathcal{S}} Z_{i,t} \geq K\right] \leq \Pr\left[\sum_{j \in \mathcal{S}} \min(A_{j,t}, 1) + A_{1,t} \geq K\right].$$

Observe that $A_{1,t}$ can be viewed as the sum of t i.i.d. Bernoulli random variables each with $1/T$. By Chernoff bound, $\Pr[A_{1,t} \geq K/\mathrm{e}^2] \leq \mathrm{e}^{-\Omega(K^2 \cdot T/t)}$. Thus, we have that

$$\Pr\left[\sum_{i \in \mathcal{S}} Z_{i,t} \geq K\right] \leq \Pr\left[\sum_{j \in \mathcal{S}} \min(A_{j,t}, 1) + A_{1,t} \geq K\right] \tag{20}$$

$$\leq \exp(-\Omega(K^2 \cdot T/t)) + \Pr\left[\sum_{j \in \mathcal{S}} \min(A_{j,t}, 1) + K/\mathrm{e}^2 \geq K\right]. \tag{21}$$

Observe that (1) $\{\min(A_{j,t}, 1)\}$ are negatively associated due to [29]; (2) $\mathsf{E}[\min(A_{j,t}, 1)] = 1 - (1 - 1/T)^t = 1 - \mathrm{e}^{-t/T - o(t/T)}$ for each $j \in \mathcal{S}$. By applying Chernoff-Hoeffding bound, we have

$$\Pr\left[\sum_{j \in \mathcal{S}} \min(A_{j,t}, 1) + K/\mathrm{e}^2 \geq K\right] \tag{22}$$

$$= \Pr\left[\sum_{j \in \mathcal{S}} \min(A_{j,t}, 1) - \mathsf{E}\left[\sum_{j \in \mathcal{S}} \min(A_{j,t}, 1)\right] \geq K - K/\mathrm{e}^2 - K \cdot (1 - \mathrm{e}^{-t/T - o(t/T)})\right] \tag{23}$$

$$= \Pr\left[\sum_{j \in \mathcal{S}} \min(A_{j,t}, 1) - \mathsf{E}\left[\sum_{j \in \mathcal{S}} \min(A_{j,t}, 1)\right] \geq K \cdot \mathrm{e}^{-t/T - o(t/T)} - K/\mathrm{e}^2\right] \tag{24}$$

$$\leq \Pr\left[\sum_{j \in \mathcal{S}} \min(A_{j,t}, 1) - \mathsf{E}\left[\sum_{j \in \mathcal{S}} \min(A_{j,t}, 1)\right] \geq K \cdot \mathrm{e}^{-t/T - o(t/T)}/2\right] \tag{25}$$

$$\leq \exp(-\mathrm{e}^{-2t/T - o(t/T)} \cdot K/2). \tag{26}$$

Thus, plugging into the above result to Inequality (21), we have

$$\Pr\left[\sum_{i \in \mathcal{S}} Z_{i,t} \geq K\right] \leq \exp(-\Omega(K^2 \cdot T/t)) + \exp(-\mathrm{e}^{-2t/T - o(t/T)} \cdot K/2).$$

Consider a given $\mathcal{S}$ with $|\mathcal{S}| = K$. We see that

$$\begin{aligned} \mathsf{E}[Z_1|K] &\leq \frac{1}{T}\sum_{t=1}^{T}\Pr\left[\sum_{i\in\mathcal{S}} Z_{i,t} \geq K\right] \\ &\leq \frac{1}{T}\sum_{t=1}^{T}[\exp(-\Omega(K^2\cdot T/t)) + \exp(-\mathrm{e}^{-2t/T-o(t/T)}\cdot K/2)] \\ &= \mathrm{e}^{-\Omega(K^2)} + \int_0^1 \mathrm{e}^{-(K/2)\cdot \mathrm{e}^{-2\zeta}}d\zeta \leq \mathrm{e}^{-\Omega(K^2)} + \mathrm{e}^{-K/(2\mathrm{e}^2)}. \end{aligned}$$

Observe that K takes values $0, 1, 2, \ldots, n-1$ with a uniform probability $1/n$. Thus,

$$\mathsf{E}[Z_1] \leq \frac{1}{n}\sum_{K=0}^{n-1}(\mathrm{e}^{-\Omega(K^2)} + \mathrm{e}^{-K/(2\mathrm{e}^2)}) = O(1/n).$$

This is in contrast with that OPT $\geq 1 - 1/\mathrm{e}$. Thus, we claim that Ranking achieves a ratio of zero. □

6.2 Proof of Theorem 4

Consider any bipartite graph (I, J, E) and let T denote the length of the time horizon. Let Ψ denote the *finite* set of all *deterministic* online matching policies given the graph and T. For any $\psi \in \Psi$ and offline node $i \in I$, let $q_{i,\psi}$ denote the probability (over the random arrival draws) that i gets matched under algorithm ψ by the end of the time horizon.

Let $n = |I|$ and fix a feasible solution $x \in [0, 1]^n$ to the LP (4)–(6) for the graph. Consider the following LP:

$$\begin{aligned} &\max\ \gamma \\ \text{s.t.}\ &\sum_{\psi\in\Psi} q_{i,\psi}z_\psi \geq x_i\gamma && \forall i = 1, \ldots, n \\ &\sum_{\psi\in\Psi} z_\psi = 1 \\ &z_\psi \geq 0 && \forall \psi \in \Psi, \end{aligned}$$

where variable z_ψ represents the probability that a randomized algorithm for IFM or GFM selects deterministic policy ψ. Objective γ is set to the maximum value for which the randomized online algorithm can uniformly guarantee a matching probability of $x_i\gamma$ for every offline agent i. Taking the dual of this LP, we get:

$$\begin{aligned} &\min\ \theta \\ \text{s.t.}\ &\sum_{i=1}^{n} q_{i,\psi}w_i \leq \theta && \forall \psi \in \Psi \qquad (27) \\ &\sum_{i=1}^{n} x_i w_i = 1 \\ &w_i \geq 0 && \forall i = 1, \ldots, n. \end{aligned}$$

Since Ψ is finite, by strong LP duality, whenever there exists a $x \in [0, 1]^n$ such that the optimal objective value of the primal LP is $\gamma = c$, there exist feasible weights $w_i \geq 0$ such that in the dual LP, (27) holds with $\theta = c$. That is, the LP for VOM has a feasible solution x with objective value $\sum_i x_i w_i = 1$, yet any deterministic online policy ψ cannot earn more than c (by (27)).

Since deterministic online policies are optimal in VOM with known weights, this shows that the competitive ratio for VOM cannot be better than c.

We complete the proof with the following argument. An upper bound of c on the competitive ratio for GFM must consist of an instance and an optimal offline solution that matches each agent i with probability x_i. The GFM objective of the offline solution is

$$\min_G \frac{1}{|G|} \sum_{i \in G} x_i. \tag{28}$$

It must be impossible to have a randomized online algorithm match each offline agent i with probability at least cx_i, since then the objective of the online algorithm would be at least c times the value in (28). By our LP duality argument, this implies that any online algorithm (knowing the weights w_i) cannot collect more than $c \sum_{i \in I} w_i x_i$, whereas the offline algorithm would be able to collect $\sum_{i \in I} w_i x_i$, yielding an upper bound of c on the competitive ratio for VOM as well. Taking an infimum over instances completes the proof that $\Phi(\text{GFM}) \geq \Phi(\text{VOM})$, while $\Phi(\text{IFM}) \geq \Phi(\text{GFM})$ trivially holds since IFM is a special case of GFM.

We emphasize that this result ignores any computational differences between the problems, and also does not suggest that there is no separation in their worst-case competitive ratios. Indeed, more structure is imposed on the optimal offline solution $(x_i)_{i \in I}$ in GFM than VOM: we have a lower bound on $\frac{1}{|G|} \sum_{i \in G} x_i$ for *every group* G, instead of just one lower bound on $\sum_{i \in I} w_i x_i$ for a set of weights $(w_i)_{i \in I}$. Even more structure is imposed on the optimal offline solution under IFM: we have a lower bound on x_i for *every agent* i. This is why we only conclude that $\Phi(\text{IFM}) \geq \Phi(\text{GFM}) \geq \Phi(\text{VOM})$.

6.3 Proof of Theorem 5

We split the whole proof into the following two parts.

Lemma 11. POF(IFM, VOM) = 1.

Proof. Consider a weighted star graph instance $\mathcal{I}$ as follows: There are $n + 1$ offline agents connected to one single onine agent with the weight on the first offline agent being $w_0 = n \gg 1$ and all the rest being $w_i = \epsilon \ll 1$ with $i \in [n] := \{1, 2, \ldots, n\}$. We can verify that: (1) The optimal value under VOM is at least $(1 - 1/\mathrm{e}) \cdot n$; (2) There is one single optimal solution under IFM in which all $x_e = 1/(n + 1)$ and thus, it achieves a utility of $n/(n + 1) + \epsilon \cdot n/(n + 1)$ on VOM. Thus,

$$\text{POF}(\text{IFM}, \text{VOM}, \mathcal{I}) \geq 1 - \frac{n/(n+1) + \epsilon \cdot n/(n+1)}{(1 - 1/\mathrm{e}) \cdot n},$$

which leads to POF(IFM, VOM) = 1. □

Lemma 12. POF(IFM, VOM′) = 0.

Proof. We prove the claim by contradiction. Suppose there exists an unweighted bipartite graph instance $\mathcal{I}$ such that $\text{POF}(\text{IFM}, \text{VOM}', \mathcal{I}) > 0$. Let $\mathcal{X}_f$ and $\mathcal{X}$ denote the collections of optimal solutions under IFM and VOM′, respectively. The fact that $\text{POF}(\text{IFM}, \text{VOM}', \mathcal{I}) > 0$ suggests $\mathcal{X}_f \cap \mathcal{X} = \emptyset$. Let $\bar{\mathbf{x}} \in \mathcal{X}$ be an optimal solution under VOM′ that achieves a largest possible value under IFM. Since $\bar{\mathbf{x}} \notin \mathcal{X}_f$, we claim that (1) there must exist one local perturbation applied to $\bar{\mathbf{x}}$ such that IFM can get improved *strictly* since $\bar{\mathbf{x}} \notin \mathcal{X}_f$; (2) any local perturbation that can strictly improve IFM on $\bar{\mathbf{x}}$ will result in a strict decrease in VOM′ by the choice of $\bar{\mathbf{x}}$. WLOG consider a typical local perturbation, that is to increase by $\epsilon > 0$ the value on some critical edge $e = (i, j)$ such that the fairness on i gets improved; meanwhile, it is to decrease by ϵ the value on some other edge $e' = (i', j)$. In this way, the sum on j remains invariant, and so does VOM′ since an unweighted

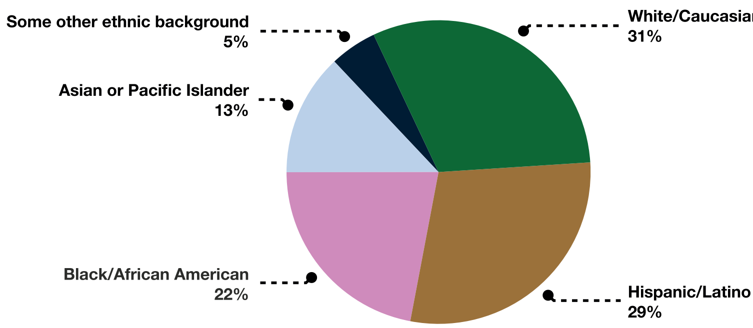


Fig. 1. The distribution of Lyft drivers by races, *i.e.,* White/Caucasian, Hispanic/Latino, Black/African American, Asian or Pacific Islander, and some other ethic background.

version is considered. Thus, we end up with such a local perturbation on $\bar{\mathbf{x}}$ that we can improve IFM strictly while maintaining VOM′ unchanged. This contradicts our previous claim. □

7 EXPERIMENTAL RESULTS

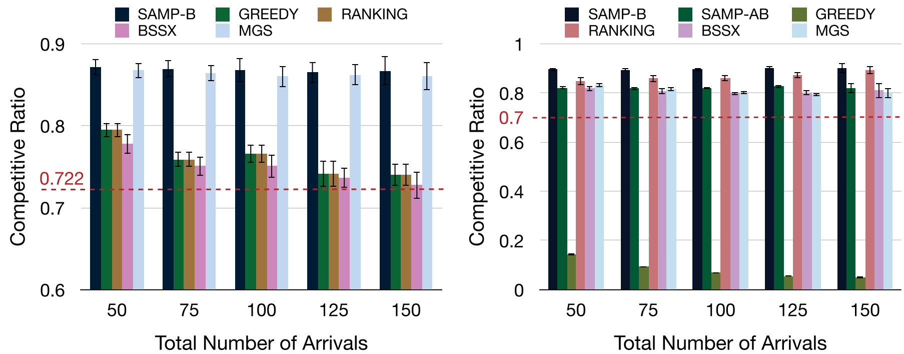
7.1 Experiments on IFM and GFM

Preprocessing of a Ride-Hailing Dataset. We test our algorithms of IFM and GFM on a public ride-hailing dataset, which is collected from the city of Chicago.[11] Following the setting in [43] and [53], we focus on a short time window and assume that drivers are offline agents while riders are online agents that arrive dynamically. Our goal is to maximize individual and group fairness among all drivers. The dataset has more than 169 million trips starting from November 2018. Each trip record includes the trip length, the starting and ending time, the pick-up and drop-off locations for the passenger, and some other information such as the fare and the tip. Note that Chicago is made up of 77 community areas that are well defined and do not overlap. Thus, we can categorize all trips according to the pre-defined community areas. According to the statistics of Lyft drivers[12] in 2021, we divide all drivers into 5 groups based on races, *i.e.,* White/Caucasian, Hispanic/Latino, Black/African American, Asian or Pacific Islander, and some other ethic background. The distribution of these 5 groups can be found in Figure 1. Recall that our metric of group fairness is defined as the minimum matching rates of offline agents over all groups, which reflects the minimum average-earning-rate among ride-hailing drivers across different races in Chicago. Thus, we believe our objective of maximizing group fairness among drivers across different races can help promote the racial and social equity.

We construct the input bipartite graph as follows: We focus on the time window from $18:00$ to $19:00$ on September 29, 2020, and subsample T trips from a total of 11, 228 trips. For each trip, we create an individual driver i and rider j, where i has an attribute of a starting community area while j has an attribute of a pair of starting and ending areas. In this way, we have $|I| = |J| = T$. For each driver in GFM, we set its race group following the distribution in Figure 1. For each driver-rider pair, we add an edge if they share the same starting area.

Algorithms. For the problems IFM and GFM, we compare our algorithm SAMP-B against the following: (a) GREEDY: Assign each arriving agent to an available neighbor (IFM) or an available neighbor whose group has the lowest matching rate at the time of arrival (GFM); break ties uniformly at random. (b) RANKING: Fix a uniform random permutation of I at the start; assign each

[11] https://data.cityofchicago.org/Transportation/Transportation-Network-Providers-Trips/m6dm-c72p.
[12] https://financesonline.com/lyft-statistics/.



(a) Experimental results on IFM. (b) Experimental results on GFM.

Fig. 2. Experimental results of IFM and GFM on a real ride-hailing dataset in Chicago: The total number of arrivals T takes values from $\{50, 75, 100, 125, 150\}$ with $|I| = |J| = T$. We add a 95% confidence interval to each bar.

arriving agent to the adjacent available offline agent who is earliest in this order. (c) BSSX: The algorithm from [8] but customized to our setting by replacing its benchmark-LP objectives with LP (1) (IFM) and LP (2) (GFM). (d) MGS: Similar to our boosting algorithm, but following [39], generating two random candidate neighbors upon the arrival of every online agent instead, and matching it with the first available one. Note that here we do not compare against the algorithm in [28], which relies on the special structure of the LP solution that all $x_e \in \{0, 1/3, 2/3\}$. This structure unfortunately no longer holds when the objective is either LP (1) (IFM) or LP (2) (GFM).

Computational complexity of SAMP-B. SAMP-B consists of two parts: **Offline Phase** and **Online Phase**. As for **Offline Phase**, SAMP-B needs to solve the benchmark LP (1) that has $N := |E|$ variables. Thus, theoretically the running time on the part of solving LP (1) can be as low as $O^*(N^{2+1/6} \log(N/\delta))$ [10], where δ is the relative accuracy and $N = |E|$. As for **Online Phase**, SAMP-B just needs to sample an assignment from a one-dimensional vector with a size no larger the size of neighboring offline agents, which is bounded by $|I|$ (a constant). Thus, the dominant part of the running time will be solving the benchmark LP (1) in **Offline Phase**. Fortunately, all computations in **Offline Phase** can be done well before the online process starts. Similar analyses can be applied to SAMP-AB.

Results and Discussions. For the real dataset, we vary the number of subsampled trips T in $\{50, 75, 100, 125, 150\}$. We first construct 100 subsampled instances for each given T, and then run 100 trials on each instance, reporting the average performance. Note that in the offline phase of SAMP-AB, when it comes to estimation of the attenuation factor $\beta_{i,t}$ for i at t, we apply the Monte-Carlo method by simulating 100 times and then taking the average.

Figure 2(a) shows that for IFM, SAMP-B performs as well as MGS, and both have a significant advantage over GREEDY, RANKING, and BSSX. The competitive ratios of SAMP-B always stay above 0.722, which is consistent with our theoretical bound in Theorem 2. Figure 2(b) shows that for GFM, SAMP-B performs as good as RANKING, BSSX and MGS, and only SAMP-AB and GREEDY fall behind. That being said, unlike GREEDY, SAMP-AB achieves a steady ratio well above 0.7 over different choices of T. This is consistent with results in Theorem 3. All results here suggest that SAMP-B and MGS are top two candidates in practice for both IFM and GFM.

We emphasize that although our SAMP-B algorithm does not significantly outperform the (fairness-adapted) MGS algorithm from the literature, it is both conceptually and implementation-wise much simpler. To our understanding, it is surprising that such a simple adaptive boosting algorithm has not been analyzed and extensively tested before.

Table 2. Network Data Statistics for Graphs from the Network Data Repository [46]

	Nodes	Edges	Max degree	Min degree	Ave. degree
socfb-Caltech36	769	16,700	248	1	43
socfb-Reed98	962	18,800	313	1	39
econ-beause	507	44,200	766	2	174
econ-mbeaflw	492	49,500	679	0	201

7.2 Experiments on VOM

Construction of Input Instances. We acknowledge that it is hard to identify real applications that can perfectly fit the model of VOM. Borodin et al. [7] have conducted comprehensive experimental studies, which compare the performance of different algorithms for unweighted online matching under KIID on a wide variety of synthetic and real datasets. They proposed an idea, called *random balanced partition method*, to generate a bipartite graph from a practical social network. The details are as follows: Suppose we have a real social network with V being the set of vertices and E being the set of edges. The method partitions V uniformly randomly into two blocks L and R, such that $|L| = \lfloor |V|/2 \rfloor$ and $|R| = \lceil |V|/2 \rceil$. It keeps only those edges that connect two vertices from the two different partitions. As indicated by [7], research on how to form a maximum matching on a bipartite graph constructed from a real social network can offer great insights regarding how to boost friendship ties among users active in online social platforms (*e.g.,* Meta Platforms).

We follow the idea in [7] and select four datasets from the Network Data Repository [46], namely, **socfb-Caltech36**, **socfb-Reed98**, **econ-because,** and **econ-mbeaflw**. The former two datasets are Meta Platforms social-network graphs, where vertices are users and edges are friendship ties. The latter two datasets are two economic networks collected from the U.S.A. in 1972, where vertices are commodities/industries and edges are economic transactions. We list detailed statistics of these four datasets in Table 2. For each network graph (V, E), we first downsample the network size $|V|$ to 200. Since the original graphs are non-bipartite, we first partition all nodes uniformly at random into two blocks to construct I and J, such that $|I| = \lfloor |V|/2 \rfloor$ and $|J| = \lceil |V|/2 \rceil$. We keep only the edges that connect two vertices from different partitions. We assign the weight for each offline vertex i to be a random value, uniformly selected from $[0, 1]$.

Algorithms. Similar to IFM and GFM, we compare SAMP-B and SAMP-AB against several baselines, including GREEDY, RANKING, BSSX [8], and MGS [39]. Additionally, we test the algorithm presented by [28], denoted by JL. For each of the four instances, we run the above 7 algorithms for 100 times and take the average as the final performance.

Results and Discussion. Figure 3 shows that SAMP-B is second only to GREEDY and is comparable to GREEDY in half of the total instances. The gap between SAMP-B and GREEDY declines as the average degree of all nodes increases; see **econ-because** and **econ-mbeaflw**. For all instances, SAMP-B outperforms the other three LP-based algorithms, BSSX, MGS and JL, all of which involve a much more complicated implementation. This establishes the superiority of SAMP-B in practical instances of VOM over the three LP-based baselines. We observe that the competitive ratios of SAMP-AB are always above 0.719, which is consistent with our theoretical bound in Theorem 3. Also, note that SAMP-AB can beat the rest three LP-based baselines in almost all scenarios (except for **socfb-Reed98**), which suggests that SAMP-AB is a top candidate among all LP-based algorithms.

8 CONCLUSIONS AND FUTURE WORK

In this article, we proposed two online-matching based models to study individual and group fairness maximization among offline agents in OMMs. For individual and group fairness maximization,

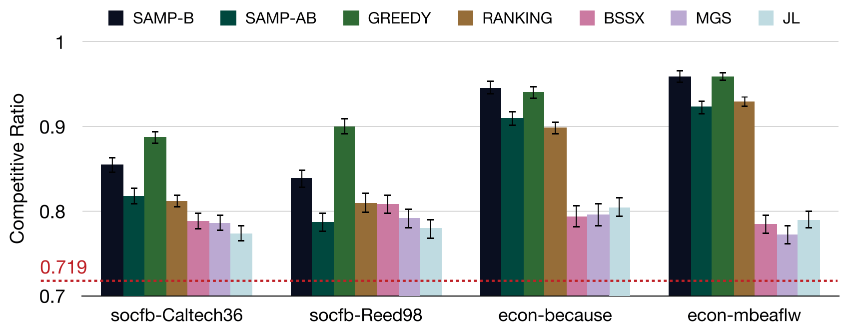


Fig. 3. Experimental results of VOM with on four real datasets from the Network Data Repository [46]. We add a 95% confidence interval to each bar.

we presented two LP-based sampling algorithms, namely SAMP-B and SAMP-AB, which achieve competitive ratios at least 0.725 and 0.719, respectively. We conducted extensive numerical experiments and results show that SAMP-B is not only conceptually easy to implement but also highly effective in practical instances of fairness-maximization related models.

One interesting future direction is to show some explicit upper bounds for IFM, GFM, and VOM. So far, all existing upper bound for the three models is 0.865, which is due to the case of unweighted online matching under KIID with integral arrival rates (UOM-KIID) [39]. Can we derive some upper bounds specifically for IFM, GFM, or VOM? We expect the upper bound of IFM should be higher than that of VOM as suggested by Theorem 4. To get an improved upper bound for IFM, for example, which is strictly lower than that from UOM-KIID, we need to first identify the structural properties exclusively existing in IFM, and then exploit them to design an instance that can weaken the power of an optimal online algorithm further compared with the clairvoyant optimal. Note that IFM differs from UOM-KIID only in the objective: the former takes the minimum while the latter the sum over all offline agents. This suggests it is impossible to break the barrier on any symmetric instances, on which both the clairvoyant optimal and online optimal achieve a uniform performance on every offline agent. Thus, our last resort is some asymmetric instances, where at least one of the clairvoyant optimal and online optimal should demonstrate a certain "preference" to some particular offline agent.

REFERENCES

[1] Marek Adamczyk, Fabrizio Grandoni, and Joydeep Mukherjee. 2015. Improved approximation algorithms for stochastic matching. In *Algorithms - ESA 2015 - Proceedings of the 23rd Annual European Symposium (ESA 2015).*

[2] Saeed Alaei, MohammadTaghi Hajiaghayi, and Vahid Liaghat. 2012. Online prophet-inequality matching with applications to ad allocation. In *Proceedings of the 13th ACM Conference on Electronic Commerce.* 18–35.

[3] Nikhil Bansal, Haotian Jiang, Raghu Meka, Sahil Singla, and Makrand Sinha. 2021. Online discrepancy minimization for stochastic arrivals. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA).* SIAM, 2842–2861.

[4] Xiaohui Bei, Xinhang Lu, Pasin Manurangsi, and Warut Suksompong. 2021. The price of fairness for indivisible goods. *Theory of Computing Systems* 65, 7 (2021), 1069–1093.

[5] Dimitris Bertsimas, Vivek F. Farias, and Nikolaos Trichakis. 2011. The price of fairness. *Operations Research* 59, 1 (2011), 17–31.

[6] Kostas Bimpikis, Ozan Candogan, and Daniela Saban. 2019. Spatial pricing in ride-sharing networks. *Operations Research* 67, 3 (2019), 744–769.

[7] Allan Borodin, Christodoulos Karavasilis, and Denis Pankratov. 2020. An experimental study of algorithms for online bipartite matching. *Journal of Experimental Algorithmics (JEA)* 25 (2020), 1–37.

[8] Brian Brubach, Karthik Abinav Sankararaman, Aravind Srinivasan, and Pan Xu. 2020. Attenuate locally, win globally: An attenuation-based framework for online stochastic matching with timeouts. *Algorithmica* 82, 1 (2020), 64–87.

[9] Brian Brubach, Karthik Abinav Sankararaman, Aravind Srinivasan, and Pan Xu. 2020. Online stochastic matching: New algorithms and bounds. *Algorithmica* 82, 10 (2020), 2737–2783.

[10] Michael B. Cohen, Y. Lee, and Zhao Song. 2019. Solving linear programs in the current matrix multiplication time. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing* (2019).

[11] Cody Cook, Rebecca Diamond, Jonathan Hall, John A. List, and Paul Oyer. 2018. *The Gender Earnings Gap in the Gig Economy: Evidence from over a Million Rideshare Drivers*. Technical Report. National Bureau of Economic Research.

[12] John P. Dickerson, Ariel D. Procaccia, and Tuomas Sandholm. 2014. Price of fairness in kidney exchange. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'14)* (Paris, France). International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 1013–1020.

[13] John P. Dickerson, Karthik Abinav Sankararaman, Kanthi Kiran Sarpatwar, Aravind Srinivasan, Kun-Lung Wu, and Pan Xu. 2019. Online resource allocation with matching constraints. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 1681–1689.

[14] John P. Dickerson, Karthik Abinav Sankararaman, Aravind Srinivasan, and Pan Xu. 2018. Assigning tasks to workers based on historical data: Online task assignment with two-sided arrivals. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 318–326.

[15] John P. Dickerson, Karthik Abinav Sankararaman, Aravind Srinivasan, and Pan Xu. 2019. Balancing relevance and diversity in online bipartite matching via submodularity. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 1877–1884.

[16] John P. Dickerson, Karthik A. Sankararaman, Aravind Srinivasan, and Pan Xu. 2021. Allocation problems in ride-sharing platforms: Online matching with offline reusable resources. *ACM Transactions on Economics and Computation (TEAC)* 9, 3 (2021), 1–17.

[17] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*. 214–226.

[18] Elaheh Fata, Will Ma, and David Simchi-Levi. 2019. Multi-stage and multi-customer assortment optimization with inventory constraints. *Available at SSRN 3443109* (2019).

[19] Jon Feldman, Aranyak Mehta, Vahab Mirrokni, and S. Muthukrishnan. 2009. Online stochastic matching: Beating 1-1/e. In *Foundations of Computer Science, 2009. FOCS'09. 50th Annual IEEE Symposium on*. IEEE, 117–126.

[20] Yiding Feng, Rad Niazadeh, and Amin Saberi. 2019. Linear programming based online policies for real-time assortment of reusable resources. *Available at SSRN 3421227* (2019).

[21] David García-Soriano and Francesco Bonchi. 2020. Fair-by-design matching. *Data Mining and Knowledge Discovery* 34, 5 (2020), 1291–1335.

[22] Stephen Gillen, Christopher Jung, Michael Kearns, and Aaron Roth. 2018. Online learning with an unknown fairness metric. *arXiv preprint arXiv:1802.06936* (2018).

[23] Gagan Goel and Aranyak Mehta. 2008. Online budgeted matching in random input models with applications to Adwords. In *Proceedings of the 19th Annual ACM-SIAM Symposium on Discrete Algorithms*, Vol. 8. 982–991.

[24] Bernhard Haeupler, Vahab S. Mirrokni, and Morteza Zadimoghaddam. 2011. Online stochastic weighted matching: Improved approximation algorithms. In *Internet and Network Economics*. Lecture Notes in Computer Science, Vol. 7090. Springer, Berlin, 170–181.

[25] Bernhard Haeupler, Vahab S. Mirrokni, and Morteza Zadimoghaddam. 2011. Online stochastic weighted matching: Improved approximation algorithms. In *Internet and Network Economics - 7th International Workshop (WINE'11)*. 170–181.

[26] Emma Hinchliffe. 2017. Yes, there's a wage gap for Uber and Lyft drivers based on age, gender and race. https://mashable.com/2017/01/18/uber-lyft-wage-gap-rideshare/. Accessed: 2019-12-27.

[27] Zhiyi Huang and Xinkai Shu. 2021. Online stochastic matching, poisson arrivals, and the natural linear program. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*. 682–693.

[28] Patrick Jaillet and Xin Lu. 2013. Online stochastic matching: New algorithms with better bounds. *Mathematics of Operations Research* 39, 3 (2013), 624–646.

[29] Kumar Joag-Dev and Frank Proschan. 1983. Negative association of random variables with applications. *The Annals of Statistics* (1983), 286–295.

[30] Matthew Joseph, Michael J. Kearns, Jamie H. Morgenstern, and Aaron Roth. 2016. Fairness in learning: Classic and contextual bandits. In *NIPAdvances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems*. 325–333.

[31] Richard M. Karp, Umesh V. Vazirani, and Vijay V. Vazirani. 1990. An optimal algorithm for on-line bipartite matching. In *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing (STOC'90)*. 352–358.

[32] Nixie S. Lesmana, Xuan Zhang, and Xiaohui Bei. 2019. Balancing efficiency and fairness in on-demand ridesourcing. In *Advances in Neural Information Processing Systems*. 5310–5320.

[33] Z. Li, Kelsey Lieberman, William Macke, Sofia Carrillo, C. Ho, J. Wellen, and Sanmay Das. 2019. Incorporating compatible pairs in kidney exchange: A dynamic weighted matching model. In *Proceedings of the 2019 ACM Conference on Economics and Computation* (2019).

[34] Hongyao Ma, Fei Fang, and David C. Parkes. 2019. Spatio-temporal pricing for ridesharing platforms. In *Proceedings of the 2019 ACM Conference on Economics and Computation (EC'19)*. 583.

[35] Will Ma. 2018. Improvements and generalizations of stochastic knapsack and Markovian bandits approximation algorithms. *Mathematics of Operations Research* 43, 3 (2018), 789–812.

[36] Will Ma, Pan Xu, and Yifan Xu. 2021. Group-level fairness maximization in online bipartite matching. *arXiv preprint arXiv:2011.13908* (2021).

[37] Vahideh Manshadi, Rad Niazadeh, and Scott Rodilitz. 2021. Fair dynamic rationing. *Available at SSRN 3775895* (2021).

[38] Vahideh Manshadi and Scott Rodilitz. 2020. Online policies for efficient volunteer crowdsourcing. In *Proceedings of the 21st ACM Conference on Economics and Computation* (2020).

[39] Vahideh H. Manshadi, Shayan Oveis Gharan, and Amin Saberi. 2012. Online stochastic matching: Online actions based on offline statistics. *Mathematics of Operations Research* 37, 4 (2012), 559–573.

[40] Duncan C. McElfresh, Christian Kroer, Sergey Pupyrev, Eric Sodomka, Karthik Abinav Sankararaman, Zack Chauvin, Neil Dexter, and John P. Dickerson. 2020. Matching algorithms for blood donation. In *Proceedings of the 21st ACM Conference on Economics and Computation* (2020).

[41] Aranyak Mehta. 2013. Online matching and ad allocation. *Foundations and Trends in Theoretical Computer Science* 8, 4 (2013), 265–368.

[42] Michael Mitzenmacher and Eli Upfal. 2017. *Probability and Computing: Randomization and Probabilistic Techniques in Algorithms and Data Analysis*. Cambridge University Press.

[43] Vedant Nanda, Pan Xu, Karthik Abhinav Sankararaman, John Dickerson, and Aravind Srinivasan. 2020. Balancing the tradeoff between profit and fairness in rideshare platforms during high-demand hours. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 2210–2217.

[44] Vishakha Patil, Ganesh Ghalme, Vineet Nair, and Y. Narahari. 2020. Achieving fairness in the stochastic multi-armed bandit problem. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 5379–5386.

[45] Alex Rosenblat, Karen Levy, Solon Barocas, and Tim Hwang. 2016. Discriminating tastes: Customer ratings as vehicles for bias. *Available at SSRN 2858946* (2016).

[46] Ryan A. Rossi and Nesreen K. Ahmed. 2015. The network data repository with interactive graph analytics and visualization. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI'15)*, Blai Bonet and Sven Koenig (Eds.). 4292–4293.

[47] Jad Salem and Swati Gupta. 2019. Closing the GAP: Group-aware parallelization for online selection of candidates with biased evaluations. *Available at SSRN 3444283* (2019).

[48] Govind S. Sankar, Anand Louis, Meghana Nasre, and Prajakta Nimbhorkar. 2021. Matchings with group fairness constraints: Online and offline algorithms. *arXiv preprint arXiv:2105.09522* (2021).

[49] Sean R. Sinclair, Siddhartha Banerjee, and Christina Lee Yu. 2021. Sequential fair allocation: Achieving the optimal envy-efficiency tradeoff curve. *arXiv preprint arXiv:2105.05308* (2021).

[50] Tom Sühr, Asia J. Biega, Meike Zehlike, Krishna P. Gummadi, and Abhijnan Chakraborty. 2019. Two-sided fairness for repeated matchings in two-sided markets: A case study of a ride-hailing platform. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 3082–3092.

[51] Jacob Thebault-Spieker, Loren G. Terveen, and Brent Hecht. 2015. Avoiding the south side and the suburbs: The geography of mobile crowdsourcing markets. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*. ACM, 265–275.

[52] Alan Tsang, Bryan Wilder, Eric Rice, Milind Tambe, and Yair Zick. 2019. Group-fairness in influence maximization. *arXiv preprint arXiv:1903.00967* (2019).

[53] Yifan Xu and Pan Xu. 2020. Trade the system efficiency for the income equality of drivers in rideshare. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence*. 4199–4205.

[54] Boming Zhao, Pan Xu, Yexuan Shi, Yongxin Tong, Zimu Zhou, and Yuxiang Zeng. 2019. Preference-aware task assignment in on-demand taxi dispatching: An online stable matching approach. In *Proceedings of the 33rd Conference on Artificial Intelligence (AAAI'19)*. 2245–2252.

Received 7 February 2022; accepted 24 October 2022