Rawlsian Fairness in Online Bipartite Matching: Two-sided, Group, and Individual

Seyed A. Esmaeili[1], Sharmila Duppala[1], Davidson Cheng[2], Vedant Nanda[1], Aravind Srinivasan[1], and John Dickerson[2]

[1]University of Maryland, College Park
[2]Colorado College

Abstract

Online bipartite-matching platforms are ubiquitous and find applications in important areas such as crowdsourcing and ridesharing. In the most general form, the platform consists of three entities: two sides to be matched and a platform operator that decides the matching. The design of algorithms for such platforms has traditionally focused on the operator's (expected) profit. Since fairness has become an important consideration that was ignored in the existing algorithms a collection of online matching algorithms have been developed that give a fair treatment guarantee for one side of the market at the expense of a drop in the operator's profit. In this paper, we generalize the existing work to offer fair treatment guarantees to *both* sides of the market *simultaneously*, at a calculated worst case drop to operator profit. We consider group and individual Rawlsian fairness criteria. Moreover, our algorithms have theoretical guarantees and have adjustable parameters that can be tuned as desired to balance the trade-off between the utilities of the three sides. We also derive hardness results that give clear upper bounds over the performance of any algorithm.

1 Introduction

Online bipartite matching has been used to model many important applications such as crowdsourcing (Ho and Vaughan 2012; Tong et al. 2016; Dickerson et al. 2019b), rideshare (Lowalekar, Varakantham, and Jaillet 2018; Dickerson et al. 2021; Ma, Xu, and Xu 2021), and online ad allocation (Goel and Mehta 2008; Mehta 2013). In the most general version of the problem, there are three interacting entities: two sides of the market to be matched and a platform operator which assigns the matches. For example, in rideshare, riders on one side of the market submit requests, drivers on the other side of the market can take requests, and a platform operator such as Uber or Lyft matches the riders' requests to one or more available drivers. In the case of crowdsourcing, organizations offer tasks, workers look for tasks to complete, and a platform operator such as Amazon Mechanical Turk (MTurk) or Upwork matches tasks to workers.

Online bipartite matching algorithms are often designed to optimize a performance measure—usually, maximizing overall profit for the platform operator or a proxy of that objective. However, fairness considerations were largely ignored. This is troubling especially given that recent reports have indicated that different demographic groups may not receive similar treatment. For example, in rideshare platforms once the platform assigns a driver to a rider's request, both the rider and the driver have the option of rejecting the assignment and it has been observed that membership in a demographic group may cause adverse treatment in the form of higher rejection. Indeed, (Cook 2018; White 2016; Wirtschafter 2019) report that drivers could reject riders based on attributes such as gender, race, or disability. Conversely, (Rosenblat et al. 2016) reports that drivers are likely to receive less favorable ratings if they belong to certain demographic groups. A similar phenomenon exists in crowdsourcing (Galperin and Greppi 2017). Moreover, even in the absence of such evidence of discrimination, as algorithms become more prevalent in making decisions that directly affect the welfare of individuals (Barocas, Hardt, and Narayanan 2019; Dwork et al. 2012), it becomes important to guarantee a standard of fairness. Also, while much of our discussion focuses on the for-profit setting for concreteness, similar fairness issues hold in not-for-profit scenarios such as the fair matching of individuals with health-care facilities, e.g., in the time of a pandemic.

In response, a recent line of research has been concerned with the issue of designing fair algorithms for online bipartite matching. (Lesmana, Zhang, and Bei 2019; Ma and Xu 2022; Xu and Xu 2020) present algorithms which give a minimum utility guarantee for the drivers at a bounded drop to the operator's profit. Conversely, (Nanda et al. 2020) give guarantees for both the platform operator and the riders instead. Finally, (Sühr et al. 2019) shows empirical methods that achieve fairness for both the riders and drivers simultaneously but lacks theoretical guarantees and ignores the operator's profit.

Nevertheless, the existing work has a major drawback in terms of optimality guarantees. Specifically, the two sides being matched along with the platform operator constitute the three main interacting entities in online matching and despite the significant progress in fair online matching none of the previous work considers all three sides simultaneously. In this paper, we derive algorithms with theoretical guarantees for the platform operator's profit as well as fairness guarantees for the two sides of the market. Unlike the previous work we not only consider the size of the matching but also its quality. Further, we consider two online arrival settings: the **KIID** and the richer **KAD** setting (see Section 3 for definitions). We consider both group and individual notions of Rawlsian fairness and interestingly show a reduction from individual fairness to group fairness in the **KAD** setting. Moreover, we show upper bounds on the optimality guarantees of any algorithm and derive impossibility results that show a conflict between group and individual notions of fairness. Finally, we empirically test our algorithms on a real-world dataset.

2 Related Work

It is worth noting that similar to our work, (Patro et al. 2020) and (Basu et al. 2020) have considered two-sided fairness as well, although in the setting of recommendation systems where a different model is applied—and, critically, a separate objective for the operator's profit was not considered.

Fairness in bipartite matching has seen significant interest recently. The fairness definition employed has consistently been the well-known Rawlsian fairness (Rawls 1958) (i.e. max-min fairness) or its generalization Leximin fairness.[*] We note that the objective to be maximized (other than the fairness objective) represents operator profit in our setting.

The case of offline and unweighted maximum cardinality matching is addressed by (García-Soriano and Bonchi 2020), who give an algorithm with Leximin fairness guarantees for one side of the market (one side of the bipartite graph) and show that this can be achieved without sacrificing the size of the match. Motivated by fairness consideration for drivers in ridesharing, (Lesmana, Zhang, and Bei 2019) considers the problem of offline and weighted matching. Specifically, they show an algorithm with a provable trade-off between the operator's profit and the minimum utility guaranteed to any vertex in one-side of the market.

Recently, (Ma, Xu, and Xu 2020) considered fairness for the online part of the graph through a group notion of fairness. In particular, the utility for a group is added across the different types and is minimized for the group worst off, in rough terms their notion translates to maximizing the minimum utility accumulated by a group throughout the matching. Their notion of fairness is very similar to the one we consider here. However, (Ma, Xu, and Xu 2020) considers fairness only on one side of the graph and ignores the operator's profit. Further, only the matching size is considered to measure utility, i.e. edges are unweighted.

A new notion of group fairness in online matching is considered in (Sankar et al. 2021). In rough terms, their group fairness criterion amounts to establishing a quota for each group and ensuring that the matching does not exceed that quota. This notion can be seen as ensuring that the system is not dominated by a specific group and is in some sense an opposite to max-min fairness as the utility is upper bounded instead of being lower bounded. Further, the fairness guarantees considered are one-sided as well.

On the empirical side of fair online matching, (Mattei, Saffidine, and Walsh 2017) and (Lee et al. 2019) give application-specific treatments in the context of deceased-donor organ allocation and food bank provisioning, respectively. More related to our work is that of (Sühr et al. 2019; Zhou, Marecek, and Shorten 2021) which consider the rideshare problem and provide algorithms to achieve fairness for both sides of the graph simultaneously, however both papers lack theoretical guarantees and in the case of (Sühr et al. 2019) the operator's profit is not considered.

3 Online Model & Optimization Objectives

Our model follows that of (Mehta 2013; Feldman et al. 2009; Bansal et al. 2010; Alaei, Hajiaghayi, and Liaghat 2013) and others. We have a bipartite graph $G = (U, V, E)$ where U represents the set of static (offline) vertices (workers) and V represents the set of online vertex types (job types) which arrive dynamically in each round. The online matching is done over T rounds. In a given round t, a vertex of type v is sampled from V with probability $p_{v,t}$ with $\sum_{v \in V} p_{v,t} = 1, \forall t \in [T]$ the probability $p_{v,t}$ is known beforehand for each type v and each round t. This arrival setting is referred to as the known adversarial distribution (**KAD**) setting (Alaei, Hajiaghayi, and Liaghat 2013; Dickerson et al. 2021). When the distribution is stationary, i.e. $p_{v,t} = p_v, \forall t \in [T]$, we have the arrival setting of the known independent identical distribution (**KIID**). Accordingly, the expected number of arrivals of type v in T rounds is $n_v = \sum_{t \in [T]} p_{v,t}$, which reduces to $n_v = Tp_v$ in the **KIID** setting. We assume that $n_v \in \mathbb{Z}^+$ for **KIID** (Bansal et al. 2010). Every vertex u (v) has a group membership,[†] with $\mathcal{G}$ being the set of all group memberships; for any vertex $u \in U$, we denote its group memberships by $g(u) \in \mathcal{G}$ (similarly, we have $g(v)$ for $v \in V$). Conversely, for a group g, $U(g)$ ($V(g)$) denotes the subset of U (V) with group membership g. A vertex u (v) has a set of incident edges E_u (E_v) which connect it to vertices in the opposite side of the graph. In a given round, once a vertex (job) v arrives, an irrevocable decision has to be made on whether to reject v or assign it to a neighbouring vertex u (where $(u, v) \in E_v$) which has not been matched before. Suppose, that v is assigned to u, then the assignment is not necessarily successful rather it succeeds with probability $p_e = p_{(u,v)} \in [0, 1]$. This models the fact that an assignment could fail for some reason such as the worker refusing the assigned job. Furthermore, each vertex u has patience parameter $\Delta_u \in \mathbb{Z}^+$ which indicates the number of failed assignments it can tolerate before leaving the system, i.e. if u receives Δ_u failed assignments then it is deleted from the graph. Similarly, a vertex v has patience $\Delta_v \in \mathbb{Z}^+$, if a vertex v arrives in a given round, then it would tolerate at most Δ_v many failed assignments in that round before leaving the system.

[*] Leximin fairness maximizes the minimum utility like max-min fairness. However, it proceeds to maximize the second worst utility, and so on until the list is exhausted.

[†] For a clearer representation we assume each vertex belongs to one group although our algorithms apply to the case where a vertex can belong to multiple groups.

For a given edge $e = (u, v) \in E$, we let each entity assign its own utility to that edge. In particular, the platform operator assigns a utility of w_e^O, whereas the offline vertex u assigns a utility of w_e^U, and the online vertex v assigns a utility of w_e^V. This captures entities' heterogeneous wants. For example, in ridesharing, drivers may desire long trips from nearby riders, whereas the platform operator would not be concerned with the driver's proximity to the rider, although this maybe the only consideration the rider has. Similar motivations exist in crowdsourcing as well. We finally note that most of the details of our model such the **KIID** and **KAD** arrival settings as well as the vertex patience follow well-established and pratically motivated model choices in online matching, see Appendix (A) for more details.

Letting $\mathcal{M}$ denote the set of successful matchings made in the T rounds, then we consider the following optimization objectives:

- **Operator's Utility (Profit):** The operator's expected profit is simply the expected sums of the profits across the matched edges, this leads to $\mathbb{E}[\sum_{e\in\mathcal{M}} w_e^O]$.
- **Rawlsian Group Fairness:**
  - **Offline Side:** Denote by $\mathcal{M}_u$ the subset of edges in the matching that are incident on u. Then our fairness criterion is equal to

    $$\min_{g\in\mathcal{G}} \frac{\mathbb{E}[\sum_{u\in U(g)}(\sum_{e\in\mathcal{M}_u} w_e^U)]}{|U(g)|}.$$

    this value equals the minimum average expected utility received by a group in the offline side U.
  - **Online Side:** Similarly, we denote by $\mathcal{M}_v$ the subset of edges in the matching that are incident on vertex v, and define the fairness criterion to be

    $$\min_{g\in\mathcal{G}} \frac{\mathbb{E}[\sum_{v\in V(g)}(\sum_{e\in\mathcal{M}_v} w_e^V)]}{\sum_{v\in V(g)} n_v}.$$

    this value equals the minimum average expected utility received throughout the matching by any group in the online side V.
- **Rawlsian Individual Fairness:**
  - **Offline Side:** The definition here follows from the group fairness definition for the offline side by simply considering that each vertex u belongs to its own distinct group. Therefore, the objective is $\min_{u\in U} \mathbb{E}[\sum_{e\in\mathcal{M}_u} w_e^U]$.
  - **Online Side:** Unlike the offline side, the definition does not follow as straightforwardly. Here we cannot obtain a valid definition by simply assigning each vertex type its own group. Rather, we note that a given individual is actually a given arriving vertex at a given round $t \in [T]$, accordingly our fairness criterion is the minimum expected utility an individual receives in a given round, i.e. $\min_{t\in[T]} \mathbb{E}[\sum_{e\in\mathcal{M}_{v_t}} w_e^V)]$, where v_t is the vertex that arrived in round t.

4 Main Results

Performance Criterion: We note that we are in the online setting, therefore our performance criterion is the competitive ratio. Denote by $\mathcal{I}$ the distribution for the instances of matching problems, then $\mathrm{OPT}(\mathcal{I}) = \mathbb{E}_{I\sim\mathcal{I}}[\mathrm{OPT}(I)]$ where $\mathrm{OPT}(I)$ is the optimal value of the sampled instance I. Similarly, for a given algorithm ALG, we define the value of its objective over the distribution $\mathcal{I}$ by $\mathrm{ALG}(\mathcal{I}) = \mathbb{E}_{\mathcal{D}}[\mathrm{ALG}(I)]$ where the expectation $\mathbb{E}_{\mathcal{D}}[.]$ is over the randomness of the instance and the algorithm. The competitive ratio is then defined as $\min_{\mathcal{I}} \frac{\mathrm{ALG}(\mathcal{I})}{\mathrm{OPT}(\mathcal{I})}$.

In our work, we address optimality guarantees for each of the three sides of the matching market by providing algorithms with competitive ratio guarantees for the operator's profit and the fairness objectives of the static and online side of the market simultaneously. Specifically, for the **KIID** arrival setting we have:

Theorem 4.1. *For the **KIID** setting, algorithm* $\mathrm{TSGF}_{\boldsymbol{KIID}}(\alpha, \beta, \gamma)$ *achieves a competitive ratio of* $(\frac{\alpha}{2e}, \frac{\beta}{2e}, \frac{\gamma}{2e})$[†] *simultaneously over the operator's profit, the group fairness objective for the offline side, and the group fairness objective for the online side, where* $\alpha, \beta, \gamma > 0$ *and* $\alpha + \beta + \gamma \leq 1$.

The following two theorems hold under the condition that $p_e = 1, \forall e \in E$. Specifically for the **KAD** setting we have:

Theorem 4.2. *For the **KAD** setting, algorithm* $\mathrm{TSGF}_{\boldsymbol{KAD}}(\alpha, \beta, \gamma)$ *achieves a competitive ratio of* $(\frac{\alpha}{2}, \frac{\beta}{2}, \frac{\gamma}{2})$ *simultaneously over the operator's profit, the group fairness objective for the offline side, and the group fairness objective for the online side, where* $\alpha, \beta, \gamma > 0$ *and* $\alpha + \beta + \gamma \leq 1$.

Moreover, for the case of individual fairness whether in the **KIID** or **KAD** arrival setting we have:

Theorem 4.3. *For the **KIID** or **KAD** setting, we can achieve a competitive ratio of* $(\frac{\alpha}{2}, \frac{\beta}{2}, \frac{\gamma}{2})$ *simultaneously over the operator's profit, the individual fairness objective for the offline side, and the individual fairness objective for the online side, where* $\alpha, \beta, \gamma > 0$ *and* $\alpha + \beta + \gamma \leq 1$.

We also give the following hardness results. In particular, for a given arrival (**KIID** or **KAD**) setting and fairness criterion (group or individual), the competitive ratios for all sides cannot exceed 1 simultaneously:

Theorem 4.4. *For all arrival models, given the three objectives: operator's profit, offline side group (individual) fairness, and online side group (individual) fairness. No algorithm can achieve a competitive ratio of* (α, β, γ) *over the three objectives simultaneously such that* $\alpha + \beta + \gamma > 1$.

It is natural to wonder if we can combine individual and group fairness. Though it is possible to extend our algorithms to this setting. The follow theorem shows that they can conflict with one another:

Theorem 4.5. *Ignoring the operator's profit and focusing either on the offline side alone or the online side alone.*

[†] Here, e denotes the Euler number, not an edge in the graph.

With α_G and α_I denoting the group and individual fairness competitive ratios, respectively. No algorithm can achieve competitive ratios (α_G, α_I) over the group and individual fairness objectives of one side simultaneously such that $\alpha_G + \alpha_I > 1$.

Finally, we carry experiments on real-world datasets in Section 6.

5 Algorithms and Theoretical Guarantees

Our algorithms use linear programming (LP) based techniques (Bansal et al. 2010; Nanda et al. 2020; Xu and Xu 2020; Brubach et al. 2016b) where first a *benchmark* LP is set up to upper bound the optimal value of the problem, then an LP solution is sampled from to produce an algorithm with guarantees. Due to space constraints, all proofs and the technical details of Theorems (4.4 and 4.5) are in Appendix (B).

5.1 Group Fairness for the KIID Setting:

Before we discuss the details of the algorithm, we note that for a given vertex type $v \in V$, the expected arrival rate n_v could be greater than one. However, it is not difficult to modify the instance by "fragmenting" each type with $n_v > 1$ such that in the new instance $n_v = 1$ for each type. This can be done with the operator's profit, offline group fairness, and online group fairness having the same values. Therefore, in what remains for the **KIID** setting $n_v = 1, \forall v \in V$ and therefore for any round t, each vertex v arrives with probability $\frac{1}{T}$. It also follows that for a given group g, $\sum_{v \in V(g)} n_v = \sum_{v \in V(g)} 1 = |V(g)|$.

For each edge $e = (u, v) \in E$ we use x_e to denote the expected number of probes (i.e, assignments from u to type v not necessarily successful) made to edge e in the LP benchmark. We have a total of three LPs each having the same set of constraints of (4), but differing by the objective. For compactness we do not repeat these constraints and instead write them once. Specifically, LP objective (1) along with the constraints of (4) give the optimal benchmark value of the operator's profit. Similarly, with the same set of constraints (4) LP objective (2) and LP objective (3) give the optimal group max-min fair assignment for the offline and online sides, respectively. Note that the expected max-min objectives of (2) and (3), can be written in the form of a linear objective. For example, the max-min objective of (2) can be replaced with an LP with objective $\max \eta$ subject to the additional constraints that $\forall g \in \mathcal{G}$, $\eta \leq \frac{\sum_{u \in U(g)} \sum_{e \in E_u} w_e^U x_e p_e}{|U(g)|}$. Having introduced the LPs, we will use LP(1), LP(2), and LP(3) to refer to the platform's profit LP, the offline side group fairness LP, and the online side group fairness LP, respectively.

$$\max \textstyle\sum_{e \in E} w_e^O x_e p_e \tag{1}$$

$$\max \min_{g \in \mathcal{G}} \frac{\sum_{u \in U(g)} \sum_{e \in E_u} w_e^U x_e p_e}{|U(g)|} \tag{2}$$

$$\max \min_{g \in \mathcal{G}} \frac{\sum_{v \in V(g)} \sum_{e \in E_v} w_e^V x_e p_e}{|V(g)|} \tag{3}$$

$$\text{s.t} \quad \forall e \in E : 0 \leq x_e \leq 1 \tag{4a}$$

$$\forall u \in U : \textstyle\sum_{e \in E_u} x_e p_e \leq 1 \tag{4b}$$

$$\forall u \in U : \textstyle\sum_{e \in E_u} x_e \leq \Delta_u \tag{4c}$$

$$\forall v \in V : \textstyle\sum_{e \in E_v} x_e p_e \leq 1 \tag{4d}$$

$$\forall v \in V : \textstyle\sum_{e \in E_v} x_e \leq \Delta_v \tag{4e}$$

Now we prove that LP(1), LP(2) and LP(3) indeed provide valid upper bounds (benchmarks) for the optimal solution for the operator's profit and expected max-min fairness for the offline and online sides of the matching.

Lemma 5.1. *For the **KIID** setting, the optimal solutions of LP (1), LP (2), and LP (3) are upper bounds on the expected optimal that can be achieved by any algorithm for the operator's profit, the offline side group fairness objective, and the online side group fairness objective, respectively.*

Our algorithm makes use of the dependent rounding subroutine (Gandhi et al. 2006). We mention the main properties of dependent rounding. In particular, given a fractional vector $\vec{x} = (x_1, x_2, \ldots, x_t)$ where each $x_i \in [0, 1]$, let $k = \sum_{i \in [t]} x_i$, dependent rounding rounds x_i (possibly fractional) to $X_i \in \{0, 1\}$ for each $i \in [t]$ such that the resulting vector $\vec{X} = (X_1, X_2, X_3, \ldots, X_t)$ has the following properties: (1) **Marginal Distribution**: The probability that $X_i = 1$ is equal to x_i, i.e. $Pr[X_i = 1] = x_i$ for each $i \in [t]$. (2) **Degree Preservation**: Sum of X_i's should be equal to either $\lfloor k \rfloor$ or $\lceil k \rceil$ with probability one, i.e. $Pr[\sum_{i \in [t]} X_i \in \{\lfloor k \rfloor, \lceil k \rceil\}] = 1$. (3) **Negative Correlation**: For any $S \subseteq [t]$, (1) $Pr[\wedge_{i \in S} X_i = 0] \leq \Pi_{i \in S} Pr[X_i = 0]$ (2) $Pr[\wedge_{i \in S} X_i = 1] \leq \Pi_{i \in S} Pr[X_i = 1]$. It follows that for any $x_i, x_j \in \vec{x}$, $\mathbb{E}[X_i = 1 | X_j = 1] \leq x_i$.

Going back to the LPs (1,2,3), we denote the optimal solutions to LP (1), LP (2), and LP (3) by $\vec{x}^*$,$\vec{y}^*$ and $\vec{z}^*$ respectively. Further, we introduce the parameters $\alpha, \beta, \gamma \in [0, 1]$ where $\alpha + \beta + \gamma \leq 1$ and each of these parameters decide the "weight" the algorithm places on each objective (the operator's profit, the offline group fairness, and the online group fairness objectives). We note that our algorithm makes use of the subroutine **PPDR** (Probe with Permuted Dependent Rounding) shown in Algorithm 1.

```
Algorithm 1: PPDR($\vec{x}_v$)
Apply dependent rounding to the fractional solution $\vec{x}_v$
   to get a binary vector $\vec{X}_v$.
Choose a random permutation $\pi$ over the set $E_v$.
for $i = 1$ to $|E_v|$ do
    Probe vertex $\pi(i)$ if it is available and $\vec{X}_v(\pi(i)) = 1$
    if Probe is successful (i.e., a match) then
        break
```

The procedure of our parameterized sampling algorithm TSGF$_{\textbf{KIID}}$ is shown in Algorithm 2. Specifically, when a vertex of type v arrives at any time step we run **PPDR**$(\vec{x}_v^*)$, **PPDR**$(\vec{y}_v^*)$, or **PPDR**$(\vec{z}_v^*)$ with probabilities α, β, and γ, respectively. We do not run any of the **PPDR** subroutines and instead reject the vertex with probability $1-(\alpha+\beta+\gamma)$. The LP constraint (4e) guarantees that $\forall v \in V : \sum_{e \in E_r} s_e^* \leq$

Δ_v where $\vec{s}^*$ could be $\vec{x}^*$, $\vec{y}^*$, or $\vec{z}^*$. Therefore, when **PPDR** is invoked by the **degree preservation** property of dependent rounding the number of edges probed will not exceed Δ_v, i.e. it would be within the patience limit.

Algorithm 2: $\mathrm{TSGF}_{\textbf{KIID}}(\alpha, \beta, \gamma)$

```
Let $v$ be the vertex type arriving at time $t$.
With probability $\alpha$ run the subroutine, PPDR$(\vec{x}_v^*)$.
With probability $\beta$ run the subroutine, PPDR$(\vec{y}_v^*)$.
With probability $\gamma$ run the subroutine, PPDR$(\vec{z}_v^*)$.
Reject the arriving vertex with probability $1 - (\alpha + \beta + \gamma)$.
```

Now we analyze $\mathrm{TSGF}_{\textbf{KIID}}$ to prove Theorem 4.1. It would suffice to prove that for each edge e the expected number of successful probes is at least $\alpha\frac{x_e^*}{2e}$, $\beta\frac{y_e^*}{2e}$ and $\gamma\frac{z_e^*}{2e}$. And finally from the linearity of expectation we show that the worst case competitive ratio of the proposed online algorithm with parameters α, β and γ is at least $(\frac{\alpha}{2e}, \frac{\beta}{2e}, \frac{\gamma}{2e})$ for the operator's profit and group fairness objectives on the offline and online sides of the matching, respectively.

A critical step is to lower bound the probability that a vertex u is available (safe) at the beginning of round $t \in [T]$. Let us denote the indicator random variable for that event by $SF_{u,t}$. The following lemma enables us to lower bound for the probability of $SF_{u,t}$.

Lemma 5.2. $Pr[SF_{u,t}] \geq \left(1 - \frac{t-1}{T}\right)\left(1 - \frac{1}{T}\right)^{t-1}$.

Now that we have established a lower bound on $Pr[SF_{u,t}]$, we lower bound the probability that an edge e is probed by one of the **PPDR** subroutines conditioned on the fact that u is available (Lemma 5.3). Let $1_{e,t}$ be the indicator that $e = (u, v)$ is probed by the $\mathrm{TSGF}_{\textbf{KIID}}$ Algorithm at time t. Note that event $1_{e,t}$ occurs when (1) a vertex of type v arrives at time t and (2) e is sampled by **PPDR**($\vec{x_v}$), **PPDR**($\vec{y_v}$), or **PPDR**($\vec{z_v}$).

Lemma 5.3. $Pr[1_{e,t} \mid SF_{u,t}] \geq \alpha\frac{x_e^*}{2T}$,$Pr[1_{e,t} \mid SF_{u,t}] \geq \beta\frac{y_e^*}{2T}$, $Pr[1_{e,t} \mid SF_{u,t}] \geq \gamma\frac{z_e^*}{2T}$

Given the above lemmas Theorem 4.1 can be proved.

5.2 Group Fairness for the KAD Setting:

For the **KAD** setting, the distribution over V is time dependent and hence the probability of sampling a type v in round t is $p_{v,t} \in [0, 1]$ with $\sum_{v \in V} p_{v,t} = 1$. Further, we assume for the **KAD** setting that for every edge $e \in E$ we have $p_e = 1$. This means that whenever an incoming vertex v is assigned to a safe-to-add vertex u the assignment is successful. This also means that any non-trivial values for the patience parameters Δ_u and Δ_v become meaningless and hence we can WLOG assume that $\forall u \in U, \forall v \in V, \Delta_u = \Delta_v = 1$. From the above discussion, we have the following LP benchmarks for the operator's profit, the group fairness for the offline side and the group fairness for the online side:

$$\max \sum_{t \in [T]} \sum_{e \in E} w_e^O x_{e,t} \tag{5}$$

$$\max \min_{g \in \mathcal{G}} \frac{\sum\limits_{t \in [T]} \sum\limits_{u \in U(g)} \sum\limits_{e \in E_u} w_e^U x_{e,t}}{|U(g)|} \tag{6}$$

$$\max \min_{g \in \mathcal{G}} \frac{\sum\limits_{t \in [T]} \sum\limits_{v \in V(g)} \sum\limits_{e \in E_v} w_e^V x_{e,t}}{\sum\limits_{v \in V(g)} n_v} \tag{7}$$

$$\text{s.t} \quad \forall e \in E, \forall t \in [T] : 0 \leq x_{e,t} \leq 1 \tag{8a}$$

$$\forall u \in U : \sum_{t \in [T]} \sum_{e \in E_u} x_{e,t} \leq 1 \tag{8b}$$

$$\forall v \in V, \forall t \in [T] : \textstyle\sum_{e \in E_v} x_{e,t} \leq p_{v,t} \tag{8c}$$

Lemma 5.4. *For the **KAD** setting, the optimal solutions of LP (5), LP (6) and LP (7) are upper bounds on the expected optimal that can be achieved by any algorithm for the operator's profit, the offline side group fairness objective, and the online side group fairness objective, respectively.*

Note that in the above LP we have $x_{e,t}$ as the probability for successfully assigning an edge in round t (with an explicit dependence on t), unlike in the **KIID** setting where we had x_e instead to denote the expected number of times edge e is probed through all rounds. Similar to our solution for the **KIID** setting, we denote by $x_{e,t}^*$, $y_{e,t}^*$, and $z_{e,t}^*$ the optimal solutions of the LP benchmarks for the operator's profit, offline side group fairness, and online side group fairness, respectively.

Having the optimal solutions to the LPs, we use algorithm $\mathrm{TSGF}_{\textbf{KAD}}$ shown in Algorithm 3. In $\mathrm{TSGF}_{\textbf{KAD}}$ new parameters are introduced, specifically λ and $\rho_{e,t}$ where $\rho_{e,t}$ is the probability that edge $e = (u, v)$ is safe to add in round t, i.e. the probability that u is unmatched at the beginning of round t. For now we assume that we have the precise values of $\rho_{e,t}$ for all rounds and discuss how to obtain these values at the end of this subsection. Now conditioned on v arriving at round t and $e = (u, v)$ being safe to add, it follows that e is sampled with probability $\alpha\frac{x_{e,t}^*}{p_{v,t}}\frac{\lambda}{\rho_{e,t}} + \beta\frac{y_{e,t}^*}{p_{v,t}}\frac{\lambda}{\rho_{e,t}} + \gamma\frac{z_{e,t}^*}{p_{v,t}}\frac{\lambda}{\rho_{e,t}}$ which would be a valid probability (positive and not exceeding 1) if $\rho_{e,t} \geq \lambda$. This follows from the fact that $\alpha, \beta, \gamma \in [0, 1]$ and $\alpha + \beta + \gamma \leq 1$ and also by constraint (8c) which leads to $\frac{\sum_{e \in E_v} x_{e,t}}{p_{v,t}} \leq 1$. Further, if $\rho_{e,t} \geq \lambda$ then by constraint (8c) we have $\sum_{e \in E_v}\left(\alpha\frac{x_{e,t}^*}{p_{v,t}}\frac{\lambda}{\rho_{e,t}} + \beta\frac{y_{e,t}^*}{p_{v,t}}\frac{\lambda}{\rho_{e,t}} + \gamma\frac{z_{e,t}^*}{p_{v,t}}\frac{\lambda}{\rho_{e,t}}\right) \leq 1$ and therefore the distribution is valid. Clearly, the value of λ is important for the validity of the algorithm, the following lemma shows that $\lambda = \frac{1}{2}$ leads to a valid algorithm.

Lemma 5.5. *Algorithm $\mathrm{TSGF}_{\textbf{KAD}}$ is valid for $\lambda = \frac{1}{2}$.*

We now return to the issue of how to obtain the values of $\rho_{e,t}$ for all rounds. This can be done by using the simulation technique as done in (Dickerson et al. 2021; Adamczyk, Grandoni, and Mukherjee 2015). To elaborate, we note that we first solve the LPs (5,6,7) and hence have the values of $x_{e,t}^*$, $y_{e,t}^*$, and $z_{e,t}^*$. Now, for the first round $t = 1$, clearly $\rho_{e,t} = 1, \forall e \in E$. To obtain $\rho_{e,t}$ for $t = 2$, we simulate the arrivals and algorithm a collection of times, and use the empirically estimated probability. More precisely, for $t = 1$

```
Algorithm 3: TSGF$_{\textbf{KAD}}(\alpha, \beta, \gamma)$
Let $v$ be the vertex type arriving at time $t$.
if $E_{v,t} = \phi$ then
    Reject $v$
else
    With probability $\alpha$ probe $e$ with probability $\frac{x^*_{e,t}}{p_{v,t}} \frac{\lambda}{\rho_{e,t}}$.
    With probability $\beta$ probe $e$ with probability $\frac{y^*_{e,t}}{p_{v,t}} \frac{\lambda}{\rho_{e,t}}$.
    With probability $\gamma$ probe $e$ with probability $\frac{z^*_{e,t}}{p_{v,t}} \frac{\lambda}{\rho_{e,t}}$.
    With probability $[1 - (\alpha + \beta + \gamma)]$ reject $v$ .
```

we sample the arrival of vertex v from $p_{v,t}$ with $t = 1$ ($p_{v,t}$ values are given as part of the model), then we run our algorithm for the values of α, β, γ that we have chosen. Accordingly, at $t = 2$ some vertex in U might be matched. We do this simulation a number of times and then we take $\rho_{e,t}$ for $t = 2$ to be the average of all runs. Now having the values of $\rho_{e,t}$ for $t = 1$ and $t = 2$, we further simulate the arrivals and the algorithm to obtain $\rho_{e,t}$ for $t = 3$ and so on until we get $\rho_{e,t}$ for the last round T. We note that using the Chernoff bound (Mitzenmacher and Upfal 2017) we can rigorously characterize the error in this estimation, however by doing this simulation a number of times that is polynomial in the problem size, the error in the estimation would only affect the lower order terms in the competitive ration analysis (Dickerson et al. 2021) and hence for simplicity it is ignored. Now, with Lemma 5.5 Theorem 4.2 can be proved (see Appendix (B)).

5.3 Individual Fairness KIID and KAD Settings:

For the case of Rawlsian (max-min) individual fairness, we consider each vertex of the offline side to belong to its own distinct group and the definition of group max-min fairness would lead to individual max-min fairness. On the other hand, for the online side a similar trick would not yield a meaningful criterion, we instead define the individual max-min fairness for the online side to equal $\min\limits_{t \in [T]} \mathbb{E}[\text{util}(v_t)] = \min\limits_{t \in [T]} \mathbb{E}[\sum_{e \in \mathcal{M}_{v_t}} w^V_e)]$ where util(v_t) is the utility received by the vertex arriving in round t. If we were to denote by $x_{e,t}$ the probability that the algorithm would successfully match e in round t, then it follows straightforwardly that $\mathbb{E}[\text{util}(v_t)] = \sum_{e \in E_{v_t}} w^V_e x_{e,t}$. We consider this definition to be the valid extension of max-min fairness for the online side as we are now concerned with the minimum utility across the online individuals (arriving vertices) which are T many. The following lemma shows that we can solve two-sided individual max-min fairness by a reduction to two-sided group max-min fairness in the **KAD** arrival setting:

Lemma 5.6. *Whether in the **KIID** or **KAD** setting, a given instance of two-sided individual max-min fairness can be converted to an instance of two-sided group max-min fairness in the **KAD** setting.*

The above Lemma with algorithm TSGF$_{\textbf{KAD}}$ can be used to prove Theorem 4.3 as shown in Appendix (B).

6 Experiments

In this section, we verify the performance of our algorithm and our theoretical lower bounds for the KIID and group fairness setting using algorithm TSGF$_{\textbf{KIID}}$ (Section 5.1). We note that none of the previous work consider our three-sided setting. We use rideshare as an application example of online bipartite matching (see also, e.g., Dickerson et al. 2021; Nanda et al. 2020; Xu and Xu 2020; Barann, Beverungen, and Müller 2017). We expect similar results and performance to hold in other matching applications such as crowdsourcing.

Experimental Setup: As done in previous work, the drivers' side is the offline (static) side whereas the riders' side is the online side. We run our experiments over the widely used New York City (NYC) yellow cabs dataset (Sekulić, Long, and Demšar 2021; Nanda et al. 2020; Xu and Xu 2020; Alonso-Mora, Wallar, and Rus 2017) which contains records of taxi trips in the NYC area from 2013. Each record contains a unique (anonymized) ID of the driver, the coordinates of start and end locations of the trip, distance of the trip, and additional metadata.

Similar to (Dickerson et al. 2021; Nanda et al. 2020), we bin the starting and ending latitudes and longitudes by dividing the latitudes from $40.4°$ to $40.95°$ and longitudes from $-73°$ to $-75°$ into equally spaced grids of step size 0.005. This enables us to define each driver and request type based on its starting and ending bins. We pick out the trips between 7pm and 8pm on January 31, 2013, which is a rush hour with 10,814 drivers and 35,109 trips. We set driver patience Δ_u to 3. Following (Xu and Xu 2020), we uniformly sample rider patience Δ_v from $\{1, 2\}$.

Since the dataset does not include demographic information, for each vertex we randomly sample the group membership (Nanda et al. 2020). Specifically, we randomly assign 70% of the riders and drivers to be advantaged and the rest to be disadvantaged. The value of p_e for $e = (u, v)$ depends on whether the vertices belong to the advantaged or disadvantaged group. Specifically, $p_e = 0.6$ if both vertices are advantaged, $p_e = 0.3$ if both are disadvantaged, and $p_e = 0.1$ for other cases.

In addition to this, a key component of our work is the use of driver and rider specific utilities. We follow the work of (Sühr et al. 2019) to set the utilities. We adopt the Manhattan distance metric rather than the Euclidean distance metric since the former is a better proxy for length of taxi trips in New York City. We set the operator's utility to the rider's trip length $w^O_e = \text{tripLength}(v)$—a rough proxy for profit. In addition, the rider's utility over an edge $e = (u, v)$ is set to $w^V_e = -\text{dist}(u, v)$ where $\text{dist}(u, v)$ is the distance between the rider and the driver. The driver's utility is set to $w^U_e = \text{tripLength}(v) - \text{dist}(u, v)$. Whereas the trip length $\text{tripLength}(v)$ is available in the dataset, the distance between the rider and the driver $\text{dist}(u, v)$ is not. We therefore simulate the distance, by creating an equally spaced grid with step size 0.005 around the starting coordinates of the trip. This results in 81 possible coordinates in the vicin-

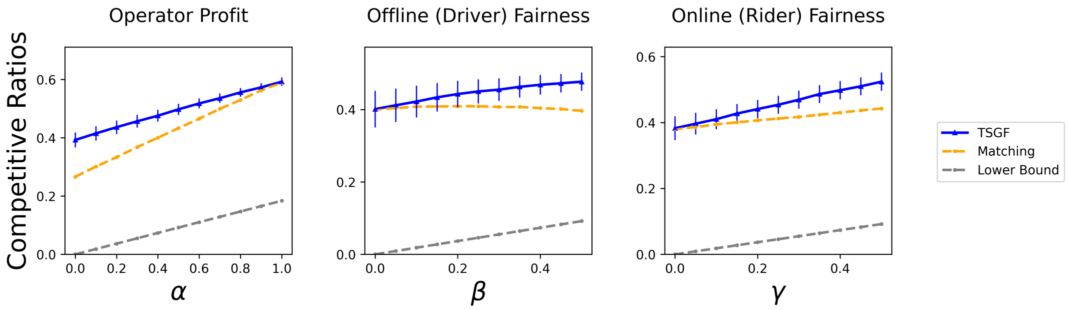

Figure 1: Competitive ratios for TSGF$_{\textbf{KIID}}$ over the operator's profit, offline (driver) fairness objective, and online (rider) fairness objective with different values of α, β, γ. Note that "Matching" refers to the case where driver and rider utilities are set to 1 across all edges. The experiment is run with $\alpha = \{0, 0.1, 0.2, ..., 1\}$, and $\beta = \gamma = \frac{1-\alpha}{2}$. Higher competitive ratio indicates better performance.

ity of the starting coordinates of the trip. We then randomly choose one of these 81 coordinates to be the location of the driver when the trip was requested. Then $dist(u, v)$ is the distance between this coordinate to the start coordinate of the trip. This is a valid approximation since the platform would not assign drivers unreasonably far away to pickup a rider. Lastly, we scale the utilities by a constant to prevent them from being negative.

We run TSGF$_{\textbf{KIID}}$ at the scale of $|U| = 49, |V| = 172$ for 100 trials. During each trial, we randomly sample 49 drivers and 172 requests between 7 and 8pm, and run TSGF$_{\textbf{KIID}}$ 100 times to measure the expected competitive ratios of this trial. We then averaged the competitive ratios over all trials, and the results are reported in figure 1. Code to reproduce our experiments is available in the blinded format[‡]; we will release that code in deblinded form upon acceptance.

Performance of TSGF$_{\textbf{KIID}}$ with Varied Parameters: Figure 1 shows the performance of our algorithm over the three objectives: operator's profit, offline (driver) group fairness, and online (rider) group fairness. It is clear that the algorithm behaves as expected with all objectives being steadily above their theoretical lower bound. More importantly, we see that increasing the weight for an objective leads to better performance for that objective. I.e., a higher weight for β leads to better performance for the offline side fairness and similar observations follow in the case of α for the operator's objective and in the case of γ for the online-fairness. This also indicates the limitation in previous work which only considered fairness for one-side since their algorithms would not be able to improve the fairness for the other ignored side.

Furthermore, previous work (e.g., Nanda et al. 2020; Xu and Xu 2020; Ma and Xu 2022) only considered the matching size when optimizing the fairness objective for the offline (drivers) or online (riders) side. This is in contrast to our setting where we consider the matching quality. To see the effect of ignoring the matching quality and only considering the size, we run the same experiments with $w_e^U = w_e^V = 1, \forall e \in E$, i.e. the quality is ignored. The results are shown shown in the graph labelled "Matching" in figure 1, it is clear that ignoring the match quality leads to noticeably worse results.

Table 1: Competitive ratios of TSGF$_{\textbf{KIID}}$ with Greedy heuristics on the NYC dataset at $|U| = 49$, $|V| = 172$. Higher competitive ratio indicates better performance.

	Profit	**Driver Fairness**	**Rider Fairness**
Greedy-O	0.431	0.549	0.503
TSGF$_{\textbf{KIID}}$ ($\alpha = 1$)	0.595	0.398	0.384
Greedy-D	0.371	0.609	0.563
TSGF$_{\textbf{KIID}}$ ($\beta = 1$)	0.517	0.571	0.44
Greedy-R	0.316	0.504	0.513
TSGF$_{\textbf{KIID}}$ ($\gamma = 1$)	0.252	0.353	0.574

Comparison to Heuristics: We also compare the performance of TSGF$_{\textbf{KIID}}$ against three other heuristics. In particular, we consider Greedy-O which is a greedy algorithm that upon the arrival of an online vertex (rider) v picks the edge $e \in E_v$ with maximum value of $p_e w_e^O$ until it either results in a match or the patience quota is reached. We also consider Greedy-R which is identical to Greedy-O except that it greedily picks the edge with maximum value of $p_e w_e^V$ instead, therefore maximizing the rider's utility in a greedy fashion. Moreover, we consider Greedy-D which is a greedy algorithm that upon the arrival of an online vertex v, first finds the group on the offline side with the lowest average utility so far, then it greedily picks an offline vertex (driver) $u \in E_v$ from this group (if possible) which has the maximum utility until it either results in a match or the patience limit is reached. We carried out 100 trials to compare the performance of TSGF$_{\textbf{KIID}}$ with the greedy algorithms, where each trial contains 49 randomly sampled drivers and 172 requests and is repeated 100 times. The aggregated results are displayed in table 1. We see that TSGF$_{\textbf{KIID}}$ outperforms the heuristics with the exception of a small under-performance in comparison to Greedy-D. However, using Greedy-D we cannot tune the weights (α, β, and γ) to balance the objectives as we can in the case of TSGF$_{\textbf{KIID}}$.

[‡] https://github.com/anonymousUser634534/TSGF

7 Conclusion

In this paper, we considered the problem of two-sided fairness in online matching platforms where there are three participating entities: the platform and the two sides to be matched. Unlike previous work where the fairness guarantees are provided for one particular entity, we give fairness guarantees to both offline and online sides of the matching simultaneously at a bounded drop to the platform operator's profit. We considered both group and individual fairness objectives. Our algorithms have theoretical guarantees and involve three tunable parameters which can be adjusted according to the utility trade-off desired. In addition, we showed hardness results which impose upper bounds on the achievable competitive ratios of any algorithm.

.

Acknowledgments

This research was supported in part by NSF CAREER Award IIS-1846237, NSF Award CCF-1918749, NSF Award CCF-1852352, NSF Award SMA-2039862, NIST MSE Award #20126334, DARPA GARD #HR00112020007, DARPA SI3-CMD #S4761, DoD WHS Award #HQ003420F0035, ARPA-E DIFFERENTIATE Award #1257037, ARL Award W911NF2120076, and gifts by research awards from Amazon, and Google. We are grateful to Pan Xu for advice and comments on earlier versions of this work.

References

Adamczyk, M.; Grandoni, F.; and Mukherjee, J. 2015. Improved approximation algorithms for stochastic matching. In *Algorithms-ESA 2015*, 1–12. Springer.

Alaei, S.; Hajiaghayi, M.; and Liaghat, V. 2012. Online prophet-inequality matching with applications to ad allocation. In *Proceedings of the 13th ACM Conference on Electronic Commerce*, 18–35.

Alaei, S.; Hajiaghayi, M.; and Liaghat, V. 2013. The online stochastic generalized assignment problem. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, 11–25. Springer.

Alonso-Mora, J.; Wallar, A.; and Rus, D. 2017. Predictive routing for autonomous mobility-on-demand systems with ride-sharing. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 3583–3590.

Bahmani, B.; and Kapralov, M. 2010. Improved bounds for online stochastic matching. In *European Symposium on Algorithms*, 170–181. Springer.

Bansal, N.; Gupta, A.; Li, J.; Mestre, J.; Nagarajan, V.; and Rudra, A. 2010. When lp is the cure for your matching woes: Improved bounds for stochastic matchings. In *European Symposium on Algorithms*, 218–229. Springer.

Barann, B.; Beverungen, D.; and Müller, O. 2017. An open-data approach for quantifying the potential of taxi ridesharing. *Decision Support Systems*, 99: 86–95.

Barocas, S.; Hardt, M.; and Narayanan, A. 2019. *Fairness and Machine Learning*. fairmlbook.org. http://www.fairmlbook.org.

Basu, K.; DiCiccio, C.; Logan, H.; and Karoui, N. E. 2020. A Framework for Fairness in Two-Sided Marketplaces. *arXiv preprint arXiv:2006.12756*.

Brubach, B.; Sankararaman, K. A.; Srinivasan, A.; and Xu, P. 2016a. New algorithms, better bounds, and a novel model for online stochastic matching. In *24th Annual European Symposium on Algorithms (ESA 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Brubach, B.; Sankararaman, K. A.; Srinivasan, A.; and Xu, P. 2016b. Online Stochastic Matching: New Algorithms and Bounds. arXiv:1606.06395.

Cook, G. 2018. Woman Says Uber Driver Denied Her Ride Because of Her Wheelchair.

Dickerson, J.; Sankararaman, K.; Sarpatwar, K.; Srinivasan, A.; Wu, K.-L.; and Xu, P. 2019a. Online resource allocation with matching constraints. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*.

Dickerson, J. P.; Sankararaman, K. A.; Srinivasan, A.; and Xu, P. 2019b. Balancing relevance and diversity in online bipartite matching via submodularity. In *AAAI*.

Dickerson, J. P.; Sankararaman, K. A.; Srinivasan, A.; and Xu, P. 2021. Allocation Problems in Ride-sharing Platforms: Online Matching with Offline Reusable Resources. *ACM Transactions on Economics and Computation (TEAC)*, 9(3): 1–17.

Dwork, C.; Hardt, M.; Pitassi, T.; Reingold, O.; and Zemel, R. 2012. Fairness through awareness. In *ITCS*.

Feldman, J.; Mehta, A.; Mirrokni, V.; and Muthukrishnan, S. 2009. Online stochastic matching: Beating 1-1/e. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science*, 117–126. IEEE.

Galperin, H.; and Greppi, C. 2017. Geographical discrimination in the gig economy. *Available at SSRN 2922874*.

Gandhi, R.; Khuller, S.; Parthasarathy, S.; and Srinivasan, A. 2006. Dependent rounding and its applications to approximation algorithms. *Journal of the ACM (JACM)*, 53(3): 324–360.

García-Soriano, D.; and Bonchi, F. 2020. Fair-by-design matching. *Data Mining and Knowledge Discovery*, 1–45.

Goel, G.; and Mehta, A. 2008. Online budgeted matching in random input models with applications to Adwords. In *SODA*.

Ho, C.-J.; and Vaughan, J. 2012. Online task assignment in crowdsourcing markets. In *AAAI*.

Karp, R. M.; Vazirani, U. V.; and Vazirani, V. V. 1990. An optimal algorithm for on-line bipartite matching. In *Proceedings of the twenty-second annual ACM symposium on Theory of computing*, 352–358.

Lee, M. K.; Kusbit, D.; Kahng, A.; Kim, J. T.; Yuan, X.; Chan, A.; See, D.; Noothigattu, R.; Lee, S.; Psomas, A.; ; and Procaccia, A. D. 2019. WeBuildAI: Participatory framework for algorithmic governance. In *CSCW*.

Lesmana, N. S.; Zhang, X.; and Bei, X. 2019. Balancing efficiency and fairness in on-demand ridesourcing. In *NeurIPS*.

Lowalekar, M.; Varakantham, P.; and Jaillet, P. 2018. Online spatio-temporal matching in stochastic and dynamic domains. *Artificial Intelligence (AIJ)*, 261: 71–112.

Ma, W.; and Xu, P. 2022. Group-level fairness maximization in online bipartite matching. In *AAMAS*.

Ma, W.; Xu, P.; and Xu, Y. 2020. Group-level fairness maximization in online bipartite matching. *arXiv preprint arXiv:2011.13908*.

Ma, W.; Xu, P.; and Xu, Y. 2021. Fairness Maximization among Offline Agents in Online-Matching Markets. In *WINE*.

Manshadi, V. H.; Gharan, S. O.; and Saberi, A. 2012. Online stochastic matching: Online actions based on offline statistics. *Mathematics of Operations Research*, 37(4): 559–573.

Mattei, N.; Saffidine, A.; and Walsh, T. 2017. Mechanisms for Online Organ Matching. In *IJCAI*.

Mehta, A. 2013. Online Matching and Ad Allocation. *Foundations and Trends in Theoretical Computer Science*, 8(4): 265–368.

Mitzenmacher, M.; and Upfal, E. 2017. *Probability and computing: Randomization and probabilistic techniques in algorithms and data analysis*. Cambridge university press.

Nanda, V.; Xu, P.; Sankararaman, K. A.; Dickerson, J. P.; and Srinivasan, A. 2020. Balancing the Tradeoff between Profit and Fairness in Rideshare Platforms During High-Demand Hours. In *AAAI*.

Patro, G. K.; Biswas, A.; Ganguly, N.; Gummadi, K. P.; and Chakraborty, A. 2020. FairRec: Two-Sided Fairness for Personalized Recommendations in Two-Sided Platforms. In *Proceedings of The Web Conference 2020*, 1194–1204.

Rawls, J. 1958. Justice as fairness. *The philosophical review*, 67(2): 164–194.

Rosenblat, A.; Levy, K. E.; Barocas, S.; and Hwang, T. 2016. Discriminating tastes: Customer ratings as vehicles for bias. *Available at SSRN 2858946*.

Sankar, G. S.; Louis, A.; Nasre, M.; and Nimbhorkar, P. 2021. Matchings with Group Fairness Constraints: Online and Offline Algorithms. *arXiv preprint arXiv:2105.09522*.

Sekulić, S.; Long, J.; and Demšar, U. 2021. A spatially aware method for mapping movement-based and place-based regions from spatial flow networks. *Transactions in GIS*, 25(4): 2104–2124.

Sühr, T.; Biega, A. J.; Zehlike, M.; Gummadi, K. P.; and Chakraborty, A. 2019. Two-sided fairness for repeated matchings in two-sided markets: A case study of a ride-hailing platform. In *KDD*.

Tong, Y.; She, J.; Ding, B.; Wang, L.; and Chen, L. 2016. Online mobile micro-task allocation in spatial crowdsourcing. In *ICDE*.

White, G. B. 2016. Uber and Lyft Are Failing Black Riders.

Wirtschafter, E. 2019. Driver discrimination still a problem as Uber and Lyft prepare to go public.

Xu, Y.; and Xu, P. 2020. Trade the System Efficiency for the Income Equality of Drivers in Rideshare. In *IJCAI*.

Zhou, Q.; Marecek, J.; and Shorten, R. N. 2021. Subgroup Fairness in Two-Sided Markets. *arXiv preprint arXiv:2106.02702*.

A Online Matching Model Details

A.1 Arrival Setting (KIID and KAD):

The modelling choices we have made follow standard settings in online matching (Mehta 2013; Alaei, Hajiaghayi, and Liaghat 2013). To elaborate further, the initial seminal paper on online matching (Karp, Vazirani, and Vazirani 1990) does not assume any prior knowledge on the arrival of the online vertices of V and follows adversarial analysis to establish theoretical guarantees on the competitive ratio. In addition to overly pessimistic theoretical results, the lack of prior knowledge is often an unrealistic assumption. Most decision makers in online matching settings are able to gain knowledge on the arrival rates of the online vertices and this knowledge can be used to build more realistic probabilistic knowledge of the arrival.

Specifically, the Known Independent and Identically Distributed **KIID** model is an established model in online matching (Feldman et al. 2009; Mehta 2013; Bahmani and Kapralov 2010; Manshadi, Gharan, and Saberi 2012; Dickerson et al. 2019b). In this model, the collection of arriving vertices on the online side belong to a finite set of known types where the type of a vertex v decides the edge connections E_v it has to the vertices of U along with the weights $w_e, \forall e \in E_v$ of those edges. Further, a given vertex of type v arrives with the same probability p_v in every round. These arrival probabilities can be estimated easily from historical data based on previous matchings.

While the **KIID** model utilizes prior knowledge which is frequently available in practical applications, it is still restrictive since it assumes that the probabilities do not vary through time. The Known Adversarial Arrival **KAD** model (also known as prophet inequality matching) on the other hand, takes into account the dynamic variation in the probabilities. Therefore, the probability a vertex of type v arrives in round t is $p_{v,t}$ instead of being constant for every round t. This model is also well-established in the matching literature and has been used in a collection of papers such as (Alaei, Hajiaghayi, and Liaghat 2012; Brubach et al. 2016a; Dickerson et al. 2021, 2019a). Despite the fact that the **KAD** model is well-motivated and richer than the **KIID** model it was not used in the one-sided online fair matching papers of (Nanda et al. 2020; Xu and Xu 2020).

A.2 Patience:

The patience parameter of a vertex Δ_u (or Δ_v) for an offline vertex u (or an online vertex v) models its tolerance for unsuccessful probes (match attempts) before leaving the system. We note that this is an important detail in the online matching model since it is frequently the case that the vertices in the online matching applications (such as advertising, crowdsourcing, and ridesharing) represent human participants who would only tolerate a fixed number of failed matching attempts before leaving the system. Like the **KIID** and **KAD** arrival models, the patience parameter is also well-established in online matching, see for example (Mehta 2013; Bansal et al. 2010; Adamczyk, Grandoni, and Mukherjee 2015). Despite the importance of this parameter, the previous work in fair online matching did not consider the patience issue for both sides simultaneously (Nanda et al. 2020; Xu and Xu 2020), handling both parameters at the same time is more challenging and leads to more tedious derivations.

We further elaborate on the meaning of the patience for both the online and offline sides, we note again that this is following the research literature on online matching:

Offline Patience: Consider a vertex u with patience Δ_u, then vertex u will remain on the offline side U unless it is successfully matched or it receives Δ_u many failed matching attempts. As a concrete example, consider a vertex u_1 with patience $\Delta_{u_1} = 2$. Clearly, in the first round ($t = 1$) u_1 will be in the offline side U, suppose an unsuccessful matching attempt (unsuccessful probe) is made in this round, then in the next round u_1 will still be there. Suppose that the next round when u_1 is probed is in the fifth round ($t = 5$), then if the probe is successful then u_1 is matched and will be removed from the offline side in the next rounds ($t > 5$), but also if the match is unsuccessful then u_1 will not be matched but will still be removed for all of the next rounds ($t > 5$) since it has a patience $\Delta_{u_1} = 2$ and therefore can only take two failed matching attempts before leaving.

Online Patience: Unlike the offline side, an online vertex v would arrive in a round t and must be matched or rejected in that given round. While in a round t we can at most match one online vertex (which is the arriving vertex v) to some offline vertex u, we can make multiple match attempts (probes) from v to the vertices it is connected to in U in that round t. The patience Δ_v of v decides the upper limit on the number of failed attempts we can make in round t before v leaves the system and can no longer be matched even if a possible match was still not attempted. As a concrete example, suppose vertex of type v_1 with $\Delta_{v_1} = 3$ arrives in round $t = 7$ and that v_1 is connected to a total of four vertices $\{u_1, u_2, u_3, u_4\}$ in U all of which are still available (i.e. unmatched and still have not ran out of patience), suppose we make match attempts (probes) to u_1 then u_2 then u_3, it follows since $\Delta_{v_1} = 3$ that v_1 has left the system and we can no longer even attempt to match it to u_4 despite that fact that its available. Further, if at any probe attempt v_1 was matched then no further probe attempts are made to v_1, e.g. if the first probe (v_1 to u_1) in the above discussion was successful, then v_1 and u_1 are matched to each other and we cannot attempt to match v_1 to $u_2, u_3,$ or u_4.

B Proofs

Here we include the missing proofs. Each lemma/theorem is restated followed by its proof.

B.1 Proofs for Section 5.1

Lemma 5.1. *For the **KIID** setting, the optimal solutions of LP (1), LP (2), and LP (3) are upper bounds on the expected optimal that can be achieved by any algorithm for the operator's profit, the offline side group fairness objective, and the online side group fairness objective, respectively.*

Proof. We follow a similar proof to that used in (Bansal et al. 2010). We shall focus on the operator's profit objective since the other objectives follow by very similar arguments. First, we note that LP(1) uses the expected values of the problem parameters, i.e. if we consider a specific graph realization G, then let N_v^G be the number of arrival for vertex type v, then it follows that LP(1) uses the expected values since $\mathbb{E}_{\mathcal{I}}[N_v^G] = 1, \forall v \in V$ where $\mathbb{E}_{\mathcal{I}}[.]$ is an expectation over the randomness of the instance. We shall therefore refer to the value of LP(1) as $LP(\mathbb{E}_{\mathcal{I}}[G])$.

To prove that $LP(\mathbb{E}_{\mathcal{I}}(G))$ is a valid upper bound it suffices to show that $LP(\mathbb{E}_{\mathcal{I}}[G[) \geq \mathbb{E}_{\mathcal{I}}[LP(G)]$ where $LP(G)$ is the optimal LP value of a realized instance G and $\mathbb{E}_{\mathcal{I}}[LP(G)]$ is the expected value of that optimal LP value. Let us then consider a specific realization G', its corresponding LP would be the following:

$$\max \textstyle\sum_{e' \in E'} w_{e'}^O p_{e'} x_{e'} \tag{9}$$

$$\text{s.t} \quad \forall e' \in E' : 0 \leq x_{e'} \leq 1 \tag{10a}$$

$$\forall u \in U : \textstyle\sum_{e' \in E'_u} x_{e'} p_{e'} \leq 1 \tag{10b}$$

$$\forall u \in U : \textstyle\sum_{e' \in E'_u} x_{e'} \leq \Delta_u \tag{10c}$$

$$\forall v' \in V' : \textstyle\sum_{e' \in E'_{v'}} x_{e'} p_{e'} \leq 1 \tag{10d}$$

$$\forall v' \in V' : \textstyle\sum_{e' \in E'_{v'}} x_{e'} \leq \Delta_{v'} \tag{10e}$$

where V' is the realization of the online side. It is clear that for a given realization $G' = (U, V', E')$ the above LP(9) is an upper bound on the operator's objective value for that realization.

Now we prove that $LP(\mathbb{E}_{\mathcal{I}}[G]) \geq \mathbb{E}_{\mathcal{I}}[LP(G)]$. The dual of the LP for the realization G' is the following:

$$\min \textstyle\sum_{u \in U} (\alpha_u + \Delta_u \beta_u) + \sum_{v' \in V'} (\alpha_{v'} + \Delta_{v'} \beta_{v'}) + \sum_{(u,v')} \gamma_{u,v'} \tag{11}$$

$$\text{s.t.} \quad \forall u \in U, \forall v' \in V' :$$

$$\beta_u + \beta_{v'} + p_{(u,v')}(\alpha_u + \alpha_{v'}) + \gamma_{(u,v')} \geq w_{(u,v')}^O p_{(u,v')} \tag{12a}$$

$$\alpha_u, \alpha_{v'}, \beta_u, \beta_{v'}, \gamma_{(u,v')} \geq 0 \tag{12b}$$

Consider the graph with the expected number of arrival $\mathbb{E}_{\mathcal{I}}(G)$ it would have a dual of the above form, let $\vec{\alpha}^*, \vec{\beta}^*, \vec{\gamma}^*$ be the optimal solution of its corresponding dual. Then it follows by the strong duality of LPs that solution $\vec{\alpha}^*, \vec{\beta}^*, \vec{\gamma}^*$ would have a value of $LP(\mathbb{E}_{\mathcal{I}}[G])$. Now for the instance G', we shall use the following dual solution $\vec{\hat{\alpha}}, \vec{\hat{\beta}}, \vec{\hat{\gamma}}$ which is set as follows:

- $\forall u \in U : \hat{\alpha}_u = \alpha_u^*, \hat{\beta}_u = \alpha_u^*$.
- $\forall v' \in V'$ of type v: $\hat{\alpha}_{v'} = \alpha_v^*, \hat{\beta}_{v'} = \beta_v^*$.
- $\forall u \in U, \forall v' \in V'$ of type v: $\hat{\gamma}_{(u,v')} = \gamma_{(u,v)}^*$.

Note that the new solution $\vec{\hat{\alpha}}, \vec{\hat{\beta}}, \vec{\hat{\gamma}}$ is a feasible dual solution since it satisfies constraints 12a and 12b. By weak duality the value of the solution $\vec{\hat{\alpha}}, \vec{\hat{\beta}}, \vec{\hat{\gamma}}$ upper bounds $LP(G')$. Now if we were to denote the number of vertices of type v that arrived in instance G' by $n_v^{G'}$, then the value of the solution $\vec{\hat{\alpha}}, \vec{\hat{\beta}}, \vec{\hat{\gamma}}$ satisfies:

$$\begin{aligned}
&\sum_{u \in U} (\hat{\alpha}_u + \Delta_u \hat{\beta}_u) + \sum_{v' \in V'} (\hat{\alpha}_{v'} + \Delta_{v'} \hat{\beta}_{v'}) + \sum_{(u,v')} \hat{\gamma}_{u,v'} \\
&= \sum_{u \in U} (\alpha_u^* + \Delta_u \beta_u^*) + \sum_{v \in V} n_v^{G'} (\alpha_v^* + \Delta_v \beta_v^*) + \sum_{(u,v)} n_v^{G'} \gamma_{u,v}^* \\
&\geq LP(G')
\end{aligned}$$

Now taking the expectation, we get:

$$\begin{aligned}
&\mathbb{E}_{\mathcal{I}}[LP(G')] \\
&\leq \mathbb{E}_{\mathcal{I}} \Big[\sum_{u \in U} (\hat{\alpha}_u + \Delta_u \hat{\beta}_u) + \sum_{v' \in V'} (\hat{\alpha}_{v'} + \Delta_{v'} \hat{\beta}_{v'}) + \sum_{(u,v')} \hat{\gamma}_{u,v'} \Big] \\
&= \mathbb{E}_{\mathcal{I}} \Big[\sum_{u \in U} (\alpha_u^* + \Delta_u \beta_u^*) + \sum_{v \in V} n_v^{G'} (\alpha_v^* + \Delta_v \beta_v^*) + \sum_{(u,v)} n_v^{G'} \gamma_{u,v}^* \Big] \\
&= \sum_{u \in U} (\alpha_u^* + \Delta_u \beta_u^*) + \sum_{v \in V} \mathbb{E}_{\mathcal{I}}[n_v^{G'}] (\alpha_v^* + \Delta_v \beta_v^*) + \sum_{(u,v)} \mathbb{E}_{\mathcal{I}}[n_v^{G'}] \gamma_{u,v}^* \\
&= \sum_{u \in U} (\alpha_u^* + \Delta_u \beta_u^*) + \sum_{v \in V} (\alpha_v^* + \Delta_v \beta_v^*) + \sum_{(u,v)} \gamma_{u,v}^* \\
&= LP(\mathbb{E}_{\mathcal{I}}[G])
\end{aligned}$$

For the offline and online group fairness objectives, we use the same steps. The difference would be in the constraints of the dual program, however following a similar assignment as done from $\vec{\alpha}^*, \vec{\beta}^*, \vec{\gamma}^*$ to $\vec{\hat{\alpha}}, \vec{\hat{\beta}}, \vec{\hat{\gamma}}$ is sufficient to prove the lemma for both fairness objectives. □

Before we prove Lemma 5.2 for the lower bound on the probability of $SF_{u,t}$. We have to first introduce the following two lemmas. Specifically, let $A_{u,t}$ be the number of successful assignments that u received and accepted before round t. Then the following lemma holds.

Lemma B.1. *For any given vertex u at time $t \in [T]$, $P[A_{u,t} = 0] \geq \left(1 - \frac{1}{T}\right)^{t-1}$.*

Proof. Let $X_{e,k}$ be the indicator random variable for u receiving an arrival request of type v where $e \in E_u$ and $k < t$. Let $Y_{e,k}$ be the indicator random variable that the edge e gets sampled by the TSGF$_{\textbf{KIID}}(\alpha, \beta, \gamma)$ algorithm at time $k < t$. Let $Z_{e,k}$ be the indicator random variable that assignment $e = (u, v)$ is successful (a match) at time $k < t$. Then $A_{u,t} = \sum_{k<t} \sum_{e \in E_u} X_{e,k} Y_{e,k} Z_{e,k}$.

$$\begin{aligned}
&Pr[A_{u,t} = 0] = \Pi_{k<t} Pr\Big[\sum_{e=(u,v) \in E_u} X_{e,k} Y_{e,k} Z_{e,k} = 0 \Big] \\
&= \Pi_{k<t} \Big(1 - Pr\Big[\sum_{e \in E_u} X_{e,k} Y_{e,k} Z_{e,k} \geq 1 \Big] \Big) \\
&\geq \Pi_{k<t} \Big(1 - \sum_{e \in E_u} \frac{1}{T} \cdot \big(\alpha x_e^* + \beta \frac{y_e^*}{q_v} + \gamma \frac{z_e^*}{q_v} \big) \cdot p_e \Big) \\
&= \Pi_{k<t} \Big(1 - \frac{1}{T} \cdot \big(\alpha \sum_{e \in E_u} x_e^* p_e + \beta \sum_{e \in E_u} y_e^* p_e + \gamma \sum_{e \in E_u} z_e^* p_e \big) \Big) \\
&\geq \Pi_{k<t} \Big(1 - \frac{1}{T} \cdot \big(\alpha \cdot 1 + \beta \cdot 1 + \gamma \cdot 1 \big) \Big) \\
&\geq \Pi_{k<t} \Big(1 - \frac{1}{T} \Big) = \Big(1 - \frac{1}{T} \Big)^{t-1}
\end{aligned}$$

□

Now we lower bound the probability that u was probed less than Δ_u times prior to t. Denote the number of probes received by u before t by $B_{u,t}$, then the following lemma holds:

Lemma B.2. $Pr[B_{u,t} < \Delta_u] \geq 1 - \frac{t-1}{T}$.

Proof. First it is clear that $B_{u,t} = \sum_{k<t} \sum_{e \in E_u} X_{e,k} Y_{e,k}$.

$$\begin{aligned}
\mathbb{E}[B_{u,t}] &= \sum_{k<t} \sum_{e \in E_u} \mathbb{E}[X_{e,k} Y_{e,k}] \\
&\leq \sum_{k<t} \sum_{e \in E_u} \frac{1}{T}\Big(\alpha x_e^* + \beta y_e^* + \gamma z_e^*\Big) \\
&\leq \sum_{k<t} \frac{1}{T}\Big(\alpha \sum_{e \in E_d} x_e^* + \beta \sum_{e \in E_u} y_e^* + \gamma \sum_{e \in E_u} z_e^*\Big) \\
&\leq \sum_{k<t} \frac{\Delta_u}{T}(\alpha + \beta + \gamma) \leq \frac{(t-1)\Delta_u}{T}
\end{aligned}$$

The inequality before the last follows from $(\alpha+\beta+\gamma) \leq 1$. Now using Markov's inequality: $Pr[B_{u,t} < \Delta_u] \geq 1 - \frac{\mathbb{E}[B_{u,t}]}{\Delta_u}$, we get $\implies Pr[B_{u,t} < \Delta_u] \geq 1 - \frac{t-1}{T}$. □

Now we restate Lemma 5.2 and prove it.

Lemma 5.2. $Pr[SF_{u,t}] \geq \Big(1 - \frac{t-1}{T}\Big)\Big(1 - \frac{1}{T}\Big)^{t-1}$.

Proof. Consider a given edge $e \in E_u$ where $k < t$

$$\begin{aligned}
&\mathbb{E}[X_{e,k} Y_{e,k} \mid A_{u,t} = 0] = \mathbb{E}[X_{e,k} Y_{e,k} \mid A_{u,k} = 0] \\
&= \frac{Pr[X_{e,k} = 1, Y_{e,k} = 1, Z_{e,k} = 0]}{Pr[A_{u,k} = 0]} \\
&\leq \frac{\frac{1}{T} \cdot \big(\alpha x_e^* + \beta y_e^* + \gamma z_e^*\big) \cdot (1 - p_e)}{1 - \sum_{e \in E_d} \frac{1}{T} \cdot \big(\alpha x_e^* + \beta y_e^* + \gamma z_e^*\big) \cdot p_e} \\
&= \frac{\frac{1}{T} \cdot \big(\alpha x_e^* + \beta y_e^* + \gamma z_e^*\big) \cdot (1 - p_e)}{1 - p_e + p_e\Big(1 - \sum_{e \in E_d} \frac{1}{T} \cdot \big(\alpha x_e^* + \beta y_e^* + \gamma z_e^*\big)\Big)} \\
&\leq \frac{1}{T} \cdot \big(\alpha x_e^* + \beta y_e^* + \gamma z_e^*\big) \cdot
\end{aligned}$$

The above inequality is due to the fact that $\sum_{e \in E_u} \frac{1}{T}\big(\alpha x_e^* + \beta y_e^* + \gamma z_e^*\big) \leq \frac{\Delta_u}{T} < 1$.

$$\begin{aligned}
\mathbb{E}[B_{u,t} | A_{u,t} = 0] &= \sum_{k<t} \sum_{e \in E_u} \mathbb{E}[X_{e,k} Y_{e,k} | A_{u,k} = 0] \\
&\leq \sum_{k<t} \sum_{e \in E_u} \frac{1}{T}\Big(\alpha x_e^* + \beta y_e^* + \gamma z_e^*\Big) \\
&\leq \sum_{k<t} \frac{1}{T}\Big(\alpha \sum_{e \in E_u} x_e^* + \beta \sum_{e \in E_u} y_e^* + \gamma \sum_{e \in E_u} z_e^*\Big) \\
&\leq \sum_{k<t} \frac{1}{T}\Big(\alpha \cdot \Delta_u + \beta \cdot \Delta_d + \gamma \cdot \Delta_u\Big) \\
&= \sum_{k<t} \frac{\Delta_u}{T}(\alpha + \beta + \gamma) \leq \frac{(t-1)\Delta_u}{T}
\end{aligned}$$

Therefore the expected number of assignments (probes) to vertex u until time t is at most $\frac{(t-1)\Delta_u}{T}$. Therefore, we have:

$$\begin{aligned}
&Pr[B_{u,t} < \Delta_u | A_{u,t} = 0] \geq 1 - \frac{\mathbb{E}[B_{u,t} | A_{u,t} = 0]}{\Delta_d} \\
&\geq 1 - \frac{(t-1)\Delta_u}{T\Delta_u} \geq 1 - \frac{t-1}{T}
\end{aligned}$$

It is to be noted that $B_{u,t}$ is the total number of probes u received before round t. Thus, we have that the events $(B_{u,t} < \Delta_u)$ and $(A_{u,t} = 0)$ are positively correlated. Therefore,

$$\begin{aligned}
&Pr[SF_{u,t}] \geq Pr[(B_{u,t} < \Delta_u) \wedge (A_{u,t} = 0)] \\
&\geq Pr[B_{u,t} < \Delta_d | A_{u,t} = 0] Pr[A_{u,t} = 0] \\
&Pr[SF_{u,t}] \geq \Big(1 - \frac{t-1}{T}\Big)\Big(1 - \frac{1}{T}\Big)^{t-1}
\end{aligned}$$

□

.

Lemma 5.3. $Pr[1_{e,t} \mid SF_{u,t}] \geq \alpha\frac{x_e^*}{2T}$,$Pr[1_{e,t} \mid SF_{u,t}] \geq \beta\frac{y_e^*}{2T}$, $Pr[1_{e,t} \mid SF_{u,t}] \geq \gamma\frac{z_e^*}{2T}$

Proof. In this part we prove that the probability that edge e is probed at time t is at least $\alpha\frac{x_e^*}{2T}$. Note that the probability that a vertex of type v arrives at time t and that Algorithm 2 calls the subroutine **PPDR**($\vec{x_r}$) is $\alpha\frac{1}{T}$. Let $E_{v,\bar{e}}$ be the set of edges in E_v excluding $e = (u, v)$. For each edge $e' \in E_{v,\bar{e}}$ let $Y_{e'}$ be the indicator for e' being before e in the random order of π (in algorithm 1) and let $Z_{e'}$ be the probability that the assignment is successful for e'. It is clear that $\mathbb{E}[Y_{e'}] = 1/2$ and that $\mathbb{E}[Z_{e'}] = p_{e'}$. Now we have:

$$Pr[1_{e,t} \mid SF_{u,t}] \tag{13}$$

$$\geq \alpha\frac{1}{T} Pr[X_e = 1] Pr\big[\sum_{e' \in E_{r,\bar{e}}} X_{e'} Y_{e'} Z_{e'} \mid X_e = 1\big] \tag{14}$$

$$= \alpha\frac{Pr[X_e = 1]}{T}\big(1 - Pr\big[\sum_{e' \in E_{v,\bar{e}}} X_{e'} Y_{e'} Z_{e'} \geq 1 \mid X_e = 1\big]\big) \tag{15}$$

$$\geq \alpha\frac{Pr[X_e = 1]}{T}\big(1 - \mathbb{E}\big[\sum_{e' \in E_{v,\bar{e}}} X_{e'} Y_{e'} Z_{e'} \geq 1 \mid X_e = 1\big]\big) \tag{16}$$

$$\geq \alpha\frac{Pr[X_e = 1]}{T}\big(1 - \sum_{e' \in E_{v,\bar{e}}} \mathbb{E}\big[X_{e'} Y_{e'} Z_{e'} \geq 1 \mid X_e = 1\big]\big) \tag{17}$$

$$\geq \alpha\frac{x_e^*}{T}\big(1 - \sum_{e' \in E_{v,\bar{e}}} x_{e'}^* \frac{1}{2} p_{e'}\big) \tag{18}$$

$$\geq \alpha\frac{x_e^*}{T}\big(1 - \frac{1}{2}\big) = \alpha\frac{x_e^*}{2T} \tag{19}$$

Applying Markov inequality we get the inequality (16). By linearity of expectation we get inequality (17). Since X_e and $X_{e'}$ are negatively correlated to each other from the

Negative Correlation property of Dependent Rounding we have $\mathbb{E}[X_{e'} \mid X_e = 1] \le x_e^*$ and we get (18). The last inequality (19) is due the fact that for any feasible solution $\{x_e^*\}$ the constraints imply that $\sum_{e\in E_v} x_e^* p_e \le 1$ for all $v \in V$. Using similar analysis we can also prove that $Pr[1_{e,t} \mid SF_{u,t}] \ge \beta \frac{y_e^*}{2T}$ and $Pr[1_{e,t} \mid SF_{u,t}] \ge \gamma \frac{z_e^*}{2T}$. □

Now we restate and prove Theorem 4.1.

Theorem 4.1. *For the **KIID** setting, algorithm* $\text{TSGF}_{\textbf{\textit{KIID}}}(\alpha,\beta,\gamma)$ *achieves a competitive ratio of* $(\frac{\alpha}{2e}, \frac{\beta}{2e}, \frac{\gamma}{2e})^{\dagger}$ *simultaneously over the operator's profit, the group fairness objective for the offline side, and the group fairness objective for the online side, where* $\alpha,\beta,\gamma > 0$ *and* $\alpha+\beta+\gamma \le 1$.

Proof. Denote the expected number of probes on each edge $e \in E$ resulting from **PPDR**$(\vec{x}_v^*)$ by n_e^x. It follows that:

$$n_e^x \ge \sum_{t=1}^{T} Pr[1_{e,t}] = \sum_{t=1}^{T} Pr[1_{e,t} \mid SF_{u,t}] Pr[SF_{u,t}]$$

$$\ge \sum_{t=1}^{T} \Big(1-\frac{1}{T}\Big)^{t-1}\Big(1-\frac{t-1}{T}\Big)\Big(\alpha\frac{x_e^*}{2T}\Big) \xrightarrow{T\to\infty} \frac{\alpha x_e^*}{2e}$$

Denote the optimal solution for the operator's profit LP by OPT_O. Let ALG_O be operator's profit obtained by our online algorithm. Using the linearity of expectation we get: $ALG_O = \mathbb{E}\Big[\sum_{e\in E} w_e^O n_e^x p_e\Big] \ge \sum_{e\in E} w_e^O p_e \frac{\alpha x_e^*}{2e} \ge \sum_{e\in E} w_e^O p_e \Big(\frac{1}{e}\Big)\frac{\alpha x_e^*}{2} \ge \frac{\alpha}{2e}(OPT_O)$. Similarly, we can obtain $\frac{\beta}{2e}$ and $\frac{\gamma}{2e}$ competitive ratios for the expected max-min group fairness guarantees on the offline and online sides, respectively. □

B.2 Proofs for Section 5.2

Lemma 5.4. *For the **KAD** setting, the optimal solutions of LP (5), LP (6) and LP (7) are upper bounds on the expected optimal that can be achieved by any algorithm for the operator's profit, the offline side group fairness objective, and the online side group fairness objective, respectively.*

Proof. We shall consider only the operator's profit objective as the other objectives follow through an identical argument. Let $1_{v,t}$ be the indicator random variable for the arrival for vertex type v in round t. Then we can obtain a realization and solve the corresponding LP and then take the expected value of LP as an upper bound on the operator's profit objective, i.e. the value $\mathbb{E}_{\mathcal{I}}[LP(G)]$ where $\mathbb{E}_{\mathcal{I}}$ is an expectation with respect to the randomness of the problem. This means replacing $1_{v,t}$ by its realization in the LP below:

$$\max \sum_{t\in[T]}\sum_{e\in E} w_e^O x_{e,t} \tag{20}$$

$$\text{s.t} \quad \forall e \in E, \forall t \in [T] : 0 \le x_{e,t} \le 1 \tag{21a}$$

$$\forall u \in U : \sum_{t\in[T]}\sum_{e\in E_u} x_{e,t} \le 1 \tag{21b}$$

$$\forall v \in V, \forall t \in [T] : \textstyle\sum_{e\in E_v} x_{e,t} \le 1_{v,t} \tag{21c}$$

If we were to replace the random variables $1_{v,t}$ by their expected value, then we would retrieve LP(5) where $\mathbb{E}_{\mathcal{I}}[1_{v,t}] = p_{v,t}$. It suffices to show that the value of LP(5) which is the LP value over the "expected" graph (the parameters replaced by their expected value) which we now denote by $LP(\mathbb{E}_{\mathcal{I}}[G])$ is an upper bound to $\mathbb{E}_{\mathcal{I}}[LP(G)]$, i.e. $LP(\mathbb{E}_{\mathcal{I}}[G]) \ge \mathbb{E}_{\mathcal{I}}[LP(G)]$. Let $x_{e,t}^{*;G}$ be the optimal solution for a given realization G and $1_{v,t}^G$ be the realization of the random variables over the instance, then we have that $\sum_{e\in E_v} x_{e,t}^{*;G} \le 1_{v,t}^G$. It follows that $\mathbb{E}_{\mathcal{I}}[x_{e,t}^{*;G}]$ is a feasible solution for LP(5), since $\mathbb{E}_{\mathcal{I}}[\sum_{e\in E_v} x_{e,t}^{*,G}] \le \mathbb{E}_{\mathcal{I}}[1_{v,t}^G] = p_{v,t}$ and the rest of the constraints are satisfied as well since they are the same in every realization. However, we have that $\mathbb{E}_{\mathcal{I}}[LP(G)] = \mathbb{E}_{\mathcal{I}}[\sum_{t\in[T]}\sum_{e\in E} w_e^O x_{e,t}^{*,G}] = \sum_{t\in[T]}\sum_{e\in E} w_e^O\, \mathbb{E}_{\mathcal{I}}[x_{e,t}^{*,G}] \le \sum_{t\in[T]}\sum_{e\in E} w_e^O x_{e,t}^* = LP(\mathbb{E}_{\mathcal{I}}[G])$ where $x_{e,t}^*$ is the optimal solution for LP(5) over the "expected" graph. The inequality followed since a feasible solution to a problem cannot exceed its optimal solution. □

Lemma 5.5. *Algorithm* $\text{TSGF}_{\textbf{\textit{KAD}}}$ *is valid for* $\lambda = \frac{1}{2}$.

Proof. We prove the validity of the algorithm for $\lambda = \frac{1}{2}$ by induction. For the base case, it is clear that $\forall e \in E, \rho_{e,t} = 1$, hence $\rho_{e,t} \ge \lambda = \frac{1}{2}$. Assume for $t' < t$, that $\rho_{e,t'} \ge \lambda = \frac{1}{2}$, then at round t we have:

$$\begin{aligned}
1 - \rho_{e,t} &= \Pr[e \text{ is not available at } t] \\
&= \Pr[e \text{ is matched in } [T-1]] \\
&\le \sum_{t'<t} \Pr[e \text{ is matched in } t'] \\
&= \sum_{t'<t} \Pr[(e \text{ is chosen by the algorithm}) \\
&\quad \wedge (u \text{ is unmatched at the beginning of } t) \\
&\quad \wedge (v \text{ arrives at } t)] \\
&= \sum_{t'<t} p_{v,t}\rho_{e,t}\Big(\alpha\frac{x_{e,t}^*}{p_{v,t}}\frac{\lambda}{\rho_{e,t}} + \beta\frac{y_{e,t}^*}{p_{v,t}}\frac{\lambda}{\rho_{e,t}} + \gamma\frac{z_{e,t}^*}{p_{v,t}}\frac{\lambda}{\rho_{e,t}}\Big) \\
&= \sum_{t'<t} \lambda(\alpha x_{e,t'}^* + \beta y_{e,t'}^* + \gamma z_{e,t'}^*) \\
&\le \lambda \sum_{t'<t}(\alpha x_{e,t'}^* + \beta y_{e,t'}^* + \gamma z_{e,t'}^*) \\
&\le \lambda(\alpha+\beta+\gamma) \le \lambda \le \frac{1}{2}
\end{aligned}$$

where we used the fact that $x_{e,t'}^*, y_{e,t'}^*, z_{e,t'}^* \le 1$ from constraint (8a) and the fact that $\alpha+\beta+\gamma \le 1$. From the above, it follows that $\rho_{e,t} \ge \frac{1}{2} \ge \lambda$. □

Now we restate and prove Theorem 4.2 using Lemma 5.5:

Theorem 4.2. *For the **KAD** setting, algorithm* $\text{TSGF}_{\textbf{\textit{KAD}}}(\alpha,\beta,\gamma)$ *achieves a competitive ratio of* $(\frac{\alpha}{2}, \frac{\beta}{2}, \frac{\gamma}{2})$ *simultaneously over the operator's profit, the group fairness objective for the offline side, and the group*

[†] Here, e denotes the Euler number, not an edge in the graph.

fairness objective for the online side, where $\alpha, \beta, \gamma > 0$ and $\alpha + \beta + \gamma \leq 1$.

Proof. For an edge e the probability that it is matched (successfully probed) is the following:

$$\begin{aligned}
&\Pr[e \text{ is successfully probed in round } t] \\
&= \Pr[(e \text{ is chosen by the algorithm}) \\
&\wedge (u \text{ is unmatched at the beginning of } t) \wedge (v \text{ arrives at } t)] \\
&= p_{v,t}\rho_{e,t}(\alpha \frac{x^*_{e,t}}{p_{v,t}} \frac{\lambda}{\rho_{e,t}} + \beta \frac{y^*_{e,t}}{p_{v,t}} \frac{\lambda}{\rho_{e,t}} + \gamma \frac{z^*_{e,t}}{p_{v,t}} \frac{\lambda}{\rho_{e,t}}) = \\
&= \alpha\lambda x^*_{e,t} + \beta\lambda y^*_{e,t} + \gamma\lambda z^*_{e,t}
\end{aligned}$$

Setting $\lambda = \frac{1}{2}$, it follows from the above that e is successfully matched with probability at least $\frac{1}{2}\alpha x^*_{e,t}$, at least $\frac{1}{2}\beta y^*_{e,t}$, and at least $\frac{1}{2}\gamma z^*_{e,t}$. Hence, the guarantees on the competitive ratios follow by linearity of the expectation. □

B.3 Proofs for Section 5.3

We restate Lemma 5.6 and give its proof:

Lemma 5.6. *Whether in the **KIID** or **KAD** setting, a given instance of two-sided individual max-min fairness can be converted to an instance of two-sided group max-min fairness in the **KAD** setting.*

Proof. Given an instance with individual fairness, define $\mathcal{G} = \{g_1, \dots, g_T\} \cup \{g'_1, \dots, g'_{|U|}\}$ as the set of all groups, thus $|\mathcal{G}| = T + |U|$, i.e. one group for each time round and one group for each offline vertex. Further given the online side types V, create a new online side V' where $|V'| = T|V|$ and $V' = V'_1 \cup V'_2 \cdots \cup V'_t \cdots \cup V'_T$ where V'_t consists of the same types as V. Moreover, $\forall v' \in V'_t, p_{v',t} = p_{v,t}$ and $p_{v',\bar{t}} = 0, \forall \bar{t} \in [T] - \{t\}$, finally $\forall v' \in V'_t, g(v') = g_t$. For the offline side U, we let each vertex have its own distinct group membership, i.e. for vertex $u_i \in U$, $g(u_i) = g'_i$.

Based on the above, it is not difficult to see that both problems have the same operator profit, and that the individual max-min fairness objectives of the original instance equal the group max-min fairness objectives of the new instance. □

From the above Lemma, applying algorithm $\text{TSGF}_{\textbf{KAD}}$ to the reduced instance leads to the following corollary:

Corollary B.1. *Given an instance of two-sided individual max-min fairness, applying $\text{TSGF}_{\textbf{KAD}}(\alpha, \beta, \gamma)$ to the reduction from Theorem 5.6 leads to a competitive ratio of $(\frac{\alpha}{2}, \frac{\beta}{2}, \frac{\gamma}{2})$ simultaneously over the operator's profit, the individual fairness objective for the offline side, and the individual fairness objective for the online side, where $\alpha, \beta, \gamma > 0$ and $\alpha + \beta + \gamma \leq 1$.*

The proof of Theorem 4.3 is immediate from the above corollary.

B.4 Proofs for Theorems 4.4 and 4.5

We now restate and prove the hardness result of Theorem 4.4:

Theorem 4.4. *For all arrival models, given the three objectives: operator's profit, offline side group (individual) fairness, and online side group (individual) fairness. No algorithm can achieve a competitive ratio of (α, β, γ) over the three objectives simultaneously such that $\alpha + \beta + \gamma > 1$.*

Proof. We prove it for group fairness in the **KIID** setting, since the **KIID** setting is a special case of the **KAD** setting, then this also proves the upper bound for the **KAD** setting.

Consider the graph $G = (U, V, E)$ which consists of three offline vertices and three online vertex types, i.e. $|U| = |V| = 3$. Each vertex in U (V) belongs to its own distinct group. The time horizon T is set to an arbitrarily large value. The arrival rate for each $v \in V$ is uniform and independent of time, i.e. **KIID** with $p_v = \frac{1}{3}$. Further, the bipartite graph is complete, i.e. each vertex of U is connected to all of the vertices of V with $p_e = 1$ for all $e \in E$. We also let $\Delta_u = 1$ for each $u \in U$, $n_v = \frac{T}{3}$ and $\Delta_v = 1$ for each $v \in V$. We represent the utilities on the edges of E with matrices where the (i, j) element gives the utility of the edge connecting vertex $u_i \in U$ and vertex $v_j \in V$. The utility matrices for the platform operator, offline, and online sides are following, respectively:

$$M_O = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, M_U = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, M_V = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}.$$

It can be seen that the utility assignments in the above example conflict between the three entities.

Let $\text{OPT}_O, \text{OPT}_U$, and OPT_V be the optimal values for the operator's profit, offline group fairness, and online group fairness, respectively. It is not difficult to see that $\text{OPT}_O = 3$, $\text{OPT}_U = 1$, and $\text{OPT}_V = 1$. Now, denote by A, B, and C the edges with values of 1 for M_O, M_U, and M_V in the graph, respectively. Further, for a given online algorithm, let a_j, b_k, and c_ℓ be the expected number of probes received by edges $j \in A, k \in B$, and $\ell \in C$, respectively. Moreover, denote the algorithm's expected value over the operator's profit, expected fairness for offline vertices, and expected fairness for online vertices by $\text{ALG}_O, \text{ALG}_U$, and ALG_V, respectively. We can upper bound the sum of the competitive ratios as follows:

$$\begin{aligned}
&\frac{\text{ALG}_O}{\text{OPT}_O} + \frac{\text{ALG}_U}{\text{OPT}_U} + \frac{ALG_V}{\text{OPT}_V} \\
&\leq \frac{\sum_{j \in A} a_j}{3} + \frac{\min_{k \in B} b_j}{1} + \frac{\min_{\ell \in C} c_j}{1} \\
&\leq \frac{\sum_{j \in A} a_i}{3} + \frac{\left(\sum_{k \in B} b_i\right)/3}{1} + \frac{\left(\sum_{\ell \in C} c_i\right)/3}{1} \\
&\leq \frac{\sum_{j \in A} a_i + \sum_{k \in B} b_i + \sum_{\ell \in C} c_i}{3} \leq \frac{3}{3} = 1
\end{aligned}$$

in the above, the second inequality follows since the minimum value is upper bounded by the average. The last inequality follows since $\Delta_u = 1$ and therefore the expected number of probes any offline vertex receives cannot exceed 1 and we have $|U| = 3$ many vertices.

To prove the same result for individual fairness we use the same graph. We note that the arrival of vertices in V is **KAD** instead with the i^{th} vertex v_i having $p_{v_i,i} = 1$ and $p_{v_i,t} = 0, \forall t \neq i$. Then we follow an argument similar to the above. □

The following proves Theorem 4.5 therefore showing that there is indeed a conflict between achieving group and individual fairness even if we were to consider only one side of the graph.

Theorem 4.5. *Ignoring the operator's profit and focusing either on the offline side alone or the online side alone. With α_G and α_I denoting the group and individual fairness competitive ratios, respectively. No algorithm can achieve competitive ratios (α_G, α_I) over the group and individual fairness objectives of one side simultaneously such that $\alpha_G + \alpha_I > 1$.*

Proof. Let us focus on the offline side, i.e. we consider α_G and α_I that are the competitive ratios for the group and individual fairness of the offline side.

Consider a graph which consists of two offline vertices and one online vertex, i.e. $|U| = 2$ and $|V| = 1$. Further, there is only one group. Let $p_e = 1, \forall e \in E$ and $\forall u \in U, \forall v \in V : \Delta_u = \Delta_v = 1$. U has two vertices u_1 and u_2 both connected to the same vertex $v \in V$. For edge (u_1, v), we let $w^U_{(u_1,v)} = 1$ and for edge (u_2, v), we let $w^U_{(u_2,v)} = L$ where L is an arbitrarily large number. Note that both of these weights are for the utility of the offline side. Finally, we only have one round so $T = 1$.

Let θ_1 and θ_2 be the expected number of probes edges (u_1, v) and (u_2, v) receive, respectively. Note that $\theta_1 = 1 - \theta_2$. It follows that the optimal offline group fairness objective is $\text{OPT}^U_G = \max_{\theta_1,\theta_2}(\theta_1 + L\theta_2) = \max_{\theta_2}((1-\theta_2) + L\theta_2) = L$. Further, the optimal offline individual fairness objective is $\text{OPT}^U_I = \min\{\theta_1, L\theta_2\}$, it is not difficult to show that $\text{OPT}^U_I = \frac{L}{L+1}$. Now consider the sum of competitive ratios, we have:

$$\begin{aligned}
\frac{\text{ALG}^U_G}{\text{OPT}^U_G} + \frac{\text{ALG}^U_I}{\text{OPT}^U_I} &= \frac{\theta_1 + L\theta_2}{L} + \frac{\min\{\theta_1, L\theta_2\}}{\frac{L}{L+1}} \\
&\leq \frac{\theta_1 + L\theta_2}{L} + \frac{\theta_1(L+1)}{L} \\
&= \frac{(L+2)\theta_1 + L\theta_2}{L} \\
&= (\theta_1 + \theta_2) + \frac{2\theta_1}{L} \\
&\leq 1 + \frac{2\theta_1}{L} \xrightarrow{L\to\infty} 1
\end{aligned}$$

this proves the result for the offline side of the graph.

To prove the result for the online side, we reverse the graph construction, i.e. having one vertex in U and two vertex types in V which arrive with equal probability. It now holds that $\text{OPT}^V_I = \min\{\theta_1, L\theta_2\}$ and by setting T to an arbitrarily large value $\text{OPT}^V_G = L$. Then we follow an identical argument to the above. □

C Additonal Experimental Results

As mentioned before one of the major contributions of our work is that we consider the operator's profit and fairness for both sides *simultaneously* instead of fairness for only one side. To further see the effects of ignoring one side, we run $\text{TSGF}_{\textbf{KIID}}$ with one side ignored (see table 2). It is clear that the fairness objective for the ignored side is indeed lower in comparison to what can be achieved in figure 1. More precisely, we can see that the Offline (Driver) and Online (Rider) fairness can be simultaneously improved to around 0.5 by setting $\alpha = 0.5, \beta = \gamma = 0.25$ (figure 1) whereas their values when their optimization weight is set to zero is 0.387 and 0.41 , respectively (see table 2).

Table 2: Results of running $\text{TSGF}_{\textbf{KIID}}$ on the NYC dataset with the fairness on one side ignored, i.e. its optimization weight set to 0: (**Top Row**) Offline (Driver) fairness ignored ($\alpha = \gamma = 0.5, \beta = 0$) and (**Bottom Row**) Online (Rider) fairness ignored ($\alpha = 0.5, \beta = 0.5, \gamma = 0$).

	Profit	**Driver Fairness**	**Rider Fairness**
$\alpha = \gamma = 0.5, \beta = 0$	0.43	0.387	0.509
$\alpha = \beta = 0.5, \gamma = 0$	0.564	0.498	0.41