Controlling Fairness and Bias in Dynamic Learning-to-Rank

Marco Morik[*][†]
m.morik@tu-berlin.de
Technische Univerität Berlin
Berlin, Germany

Ashudeep Singh[*]
ashudeep@cs.cornell.edu
Cornell University
Ithaca, NY

Jessica Hong
jwh296@cornell.edu
Cornell University
Ithaca, NY

Thorsten Joachims
tj@cs.cornell.edu
Cornell University
Ithaca, NY

ABSTRACT

Rankings are the primary interface through which many online platforms match users to items (e.g. news, products, music, video). In these two-sided markets, not only the users draw utility from the rankings, but the rankings also determine the utility (e.g. exposure, revenue) for the item providers (e.g. publishers, sellers, artists, studios). It has already been noted that myopically optimizing utility to the users – as done by virtually all learning-to-rank algorithms – can be unfair to the item providers. We, therefore, present a learning-to-rank approach for explicitly enforcing merit-based fairness guarantees to groups of items (e.g. articles by the same publisher, tracks by the same artist). In particular, we propose a learning algorithm that ensures notions of amortized group fairness, while simultaneously learning the ranking function from implicit feedback data. The algorithm takes the form of a controller that integrates unbiased estimators for both fairness and utility, dynamically adapting both as more data becomes available. In addition to its rigorous theoretical foundation and convergence guarantees, we find empirically that the algorithm is highly practical and robust.

CCS CONCEPTS

• **Information systems** → *Learning to rank.*

KEYWORDS

ranking; learning-to-rank; fairness; bias; selection bias; exposure

ACM Reference Format:
Marco Morik, Ashudeep Singh, Jessica Hong, and Thorsten Joachims. 2020. Controlling Fairness and Bias in Dynamic Learning-to-Rank. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20), July 25–30, 2020, Virtual Event, China.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3397271.3401100

1 INTRODUCTION

We consider the problem of dynamic Learning-to-Rank (LTR), where the ranking function dynamically adapts based on the feedback that users provide. Such dynamic LTR problems are ubiquitous in online systems — news-feed rankings that adapt to the number of "likes" an article receives, online stores that adapt to the number of positive reviews for a product, or movie-recommendation systems that adapt to who has watched a movie. In all of these systems, learning and prediction are dynamically intertwined, where past feedback influences future rankings in a specific form of online learning with partial-information feedback [18].

While dynamic LTR systems are in widespread use and unquestionably useful, there are at least two issues that require careful design considerations. First, the ranking system induces a bias through the rankings it presents. In particular, items ranked highly are more likely to collect additional feedback, which in turn can influence future rankings and promote misleading rich-get-richer dynamics [3, 32, 33, 40]. Second, the ranking system is the arbiter of how much exposure each item receives, where exposure directly influences opinion (e.g. ideological orientation of presented news articles) or economic gain (e.g. revenue from product sales or streaming) for the provider of the item. This raises fairness considerations about how exposure should be allocated based on the merit of the items [14, 42]. We will show in the following that naive dynamic LTR methods that are oblivious to these issues can lead to economic disparity, unfairness, and polarization.

In this paper, we present the first dynamic LTR algorithm – called FairCo – that overcomes rich-get-richer dynamics while enforcing a configurable allocation-of-exposure scheme. Unlike existing fair LTR algorithms [14, 42, 43, 48], FairCo explicitly addresses the dynamic nature of the learning problem, where the system is unbiased and fair even though the relevance and the merit of items are still being learned. At the core of our approach lies a merit-based exposure-allocation criterion that is amortized over the learning process [14, 42]. We view the enforcement of this merit-based exposure criterion as a control problem and derive a P-controller that optimizes both the fairness of exposure as well as the quality of the rankings. A crucial component of the controller is the ability to estimate merit (i.e. relevance) accurately, even though the feedback is only revealed incrementally as the system operates, and the feedback is biased by the rankings shown in the process [32]. To this effect, FairCo includes a new unbiased cardinal relevance

[*] Equal contribution.
[†] Work conducted while at Cornell University.

SIGIR '20, July 25–30, 2020, Virtual Event, China

ACM ISBN 978-1-4503-8016-4/20/07…$15.00
https://doi.org/10.1145/3397271.3401100

estimator – as opposed to existing ordinal methods [4, 33] –, which can be used both as an unbiased merit estimator for fairness and as a ranking criterion.

In addition to the theoretical justification of FairCo, we provide empirical results on both synthetic news-feed data and real-world movie recommendation data. We find that FairCo is effective at enforcing fairness while providing good ranking performance. Furthermore, FairCo is efficient, robust, and easy to implement.

2 MOTIVATION

Consider the following illustrative example of a dynamic LTR problem. An online news-aggregation platform wants to present a ranking of the top news articles on its front page. Through some external mechanism, it identifies a set $\mathcal{D} = \{d_1, ..., d_{20}\}$ of 20 articles at the beginning of each day, but it is left with the learning problem of how to rank these 20 articles on its front page. As users start coming to the platform, the platform uses the following naive algorithm to learn the ranking.

Algorithm 1: Naive Dynamic LTR Algorithm

```
Initialize counters $C(d) = 0$ for each $d \in \mathcal{D}$;
foreach user do
    present ranking $\boldsymbol{\sigma} = \mathrm{argsort}_{\mathcal{D}}[C(d)]$ (random tiebreak);
    increment $C(d)$ for the articles read by the user.
```

Executing this algorithm at the beginning of a day, the platform starts by presenting the 20 articles in random order for the first user. It may then observe that the user reads the article in position 3 and increments the counter $C(d)$ for this article. For the next user, this article now gets ranked first and the counters are updated based on what the second user reads. This cycle continues for each subsequent user. Unfortunately, this naive algorithm has at least two deficiencies that make it suboptimal or unsuitable for many ranking applications.

The first deficiency lies in the choice of $C(d)$ as an estimate of average relevance for each article – namely the fraction of users that want to read the article. Unfortunately, even with infinite amounts of user feedback, the counters $C(d)$ are not consistent estimators of average relevance [32, 33, 40]. In particular, items that happened to get more reads in early iterations get ranked highly, where more users find them and thus have the opportunity to provide more positive feedback for them. This perpetuates a rich-get-richer dynamic, where the feedback count $C(d)$ recorded for each article does not reflect how many users actually wanted to read the article.

The second deficiency of the naive algorithm lies in the ranking policy itself, creating a source of unfairness even if the true average relevance of each article was accurately known [7, 14, 42]. Consider the following omniscient variant of the naive algorithm that ranks the articles by their true average relevance (i.e. the true fraction of users who want to read each article). How can this ranking be unfair? Let us assume that we have two groups of articles, G_{right} and G_{left}, with 10 items each (i.e. articles from politically right- and left-leaning sources). 51% of the users (right-leaning) want to read the articles in group G_{right}, but not the articles in group G_{left}. In reverse, the remaining 49% of the users (left-leaning) like only the articles in G_{left}. Ranking articles solely by their true average relevance puts items from G_{right} into positions 1-10 and the items from G_{left} in positions 11-20. This means the platform gives the articles in G_{left} vastly less exposure than those in G_{right}. We argue that this can be considered unfair since the two groups receive disproportionately different outcomes despite having similar merit (i.e. relevance). Here, a 2% difference in average relevance leads to a much larger difference in exposure between the groups.

We argue that these two deficiencies – namely bias and unfairness – are not just undesirable in themselves, but that they have undesirable consequences. For example, biased estimates lead to poor ranking quality, and unfairness is likely to alienate the left-leaning users in our example, driving them off the platform and encouraging polarization.

Furthermore, note that these two deficiencies are not specific to the news example, but that the naive algorithm leads to analogous problems in many other domains. For example, consider a ranking system for job applicants, where rich-get-richer dynamics and exposure allocation may perpetuate and even amplify existing unfairness (e.g. disparity between male and female applicants). Similarly, consider an online marketplace where products of different sellers (i.e. groups) are ranked. Here rich-get-richer dynamics and unfair exposure allocation can encourage monopolies and drive some sellers out of the market.

These examples illustrate the following two desiderata that a less naive dynamic LTR algorithm should fulfill.

Unbiasedness: The algorithm should not be biased or subject to rich-get-richer dynamics.

Fairness: The algorithm should enforce a fair allocation of exposure based on merit (e.g. relevance).

With these two desiderata in mind, this paper develops alternatives to the Naive algorithm. In particular, after introducing the dynamic learning-to-rank setting in Section 4, Section 5 formalizes an amortized notion of merit-based fairness, accounting for the fact that merit itself is unknown at the beginning of the learning process and is only learned throughout. Section 6 then addresses the bias problem, providing estimators that eliminate the presentation bias for both global and personalized ranking policies. Finally, Section 7 proposes a control-based algorithm that is designed to optimize ranking quality while dynamically enforcing fairness.

3 RELATED WORK

Ranking algorithms are widely recognized for their potential for societal impact [8], as they form the core of many online systems, including search engines, recommendation systems, news feeds, and online voting. Controlling rich-get-richer phenomena in recommendations and rankings has been studied from the perspective of both optimizing utility through exploration as well as ensuring fairness of such systems [2, 41, 49]. There are several adverse consequences of naive ranking systems [20], such as political polarization [11], misinformation [46], unfair allocation of exposure [43], and biased judgment [8] through phenomena such as the Matthew effect [3, 24]. Viewing such ranking problems as two-sided markets of users and items that each derive utility from the ranking system brings a novel perspective to tackling such problems [1, 42]. In this work, we take inspiration from these works to develop methods for mitigating bias and unfairness in a dynamic setting.

Machine learning methods underlie most ranking algorithms. There has been a growing concern around the question of how machine learning algorithms can be unfair, especially given their numerous real-world applications [10]. There have been several definitions proposed for fairness in the binary classification setting [9], as well as recently in the domains of rankings in recommendations and information retrieval [13, 14, 17, 42]. The definitions of fairness in ranking span from ones purely based on the composition of the top-k [17], to relevance-based definitions such as fairness of exposure [42], and amortized attention equity [14]. We will discuss these definitions in greater detail in Section 5. Our work also relates to the recent interest in studying the impact of fairness when learning algorithms are applied in dynamic settings [22, 36, 44].

In information retrieval, there has been a long-standing interest in learning to rank from biased click data. As already argued above, the bias in logged click data occurs because the feedback is incomplete and biased by the presentation. Numerous approaches based on preferences (e.g. [26, 31]), click models (e.g. [19]), and randomized interventions (e.g. [37]) exist. Most recently, a new approach for de-biasing feedback data using techniques from causal inference and missing data analysis was proposed to provably eliminate selection biases [6, 33]. We follow this approach in this paper, extend it to the dynamic ranking setting, and propose a new unbiased regression objective in Section 6.

Learning in our dynamic ranking setting is related to the conventional learning-to-rank algorithms such as LambdaRank, LambdaMART, RankNet, Softrank etc. [16, 45]. However, to implement fairness constraints based on merit, we need to explicitly estimate relevance to the user as a measure of merit while the scores estimated by these methods don't necessarily have a meaning. Our setting is also closely related to online learning to rank for top-k ranking where feedback is observed only on the top-k items, and hence exploration interventions are necessary to ensure convergence [27, 35, 38, 50]. These algorithms are designed with respect to a click-model assumption [50] or learning in the presence of document features [35]. A key difference in our method is that we do not consider exploration through explicit interventions, but merely exploit user-driven exploration. However, explicit exploration could also be incorporated into our algorithms to improve the convergence rate of our methods.

4 DYNAMIC LEARNING-TO-RANK

We begin by formally defining the dynamic LTR problem. Given is a set of items $\mathcal{D}$ that needs to be ranked in response to incoming requests. At each time step t, a request

$$\boldsymbol{x}_t, \mathbf{r}_t \sim \mathrm{P}(\boldsymbol{x}, \mathbf{r}) \tag{1}$$

arrives i.i.d. at the ranking system. Each request consists of a feature vector describing the user's information need $\boldsymbol{x}_t$ (e.g. query, user profile), and the user's vector of true relevance ratings $\mathbf{r}_t$ for all items in the collection $\mathcal{D}$. Only the feature vector $\boldsymbol{x}_t$ is visible to the system, while the true relevance ratings $\mathbf{r}_t$ are hidden. Based on the information in $\boldsymbol{x}_t$, a ranking policy $\pi_t(\boldsymbol{x})$ produces a ranking $\boldsymbol{\sigma}_t$ that is presented to the user. Note that the policy may ignore the information in $\boldsymbol{x}_t$, if we want to learn a single global ranking like in the introductory news example.

After presenting the ranking $\boldsymbol{\sigma}_t$, the system receives a feedback vector $\mathbf{c}_t$ from the user with a non-negative value $\mathbf{c}_t(d)$ for every $d \in \mathcal{D}$. In the simplest case, it is 1 for click and 0 for no click, and we will use the word "click" as a placeholder throughout this paper for simplicity. But the feedback may take many other forms and does not have to be binary. For example, in a video streaming service, the feedback may be the percentage the user watched of each video.

After the feedback $\mathbf{c}_t$ was received, the dynamic LTR algorithm $\mathcal{A}$ now updates the ranking policy and produces the policy π_{t+1} that is used in the next time step.

$$\pi_{t+1} \longleftarrow \mathcal{A}((\boldsymbol{x}_1, \boldsymbol{\sigma}_1, \mathbf{c}_1), ..., (\boldsymbol{x}_t, \boldsymbol{\sigma}_t, \mathbf{c}_t))$$

An instance of such a dynamic LTR algorithm is the Naive algorithm already outlined in Section 2. It merely computes $\sum \mathbf{c}_t$ to produce a new ranking policy for $t + 1$ (here a global ranking independent of $\boldsymbol{x}$).

4.1 Partial and Biased Feedback

A key challenge of dynamic LTR lies in the fact that the feedback $\mathbf{c}_t$ provides meaningful feedback only for the items that the user examined. Following a large body of work on click models [19], we model this as a censoring process. Specifically, for a binary vector $\mathbf{e}_t$ indicating which items were examined by the user, we model the relationship between $\mathbf{c}_t$ and $\mathbf{r}_t$ as follows.

$$\mathbf{c}_t(d) = \begin{cases} \mathbf{r}_t(d) & \text{if } \mathbf{e}_t(d) = 1 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Coming back to the running example of news ranking, $\mathbf{r}_t$ contains the full information about which articles the user is interested in reading, while $\mathbf{c}_t$ reveals this information only for the articles d examined by the user (i.e. $\mathbf{e}_t(d) = 1$). Analogously, in the job placement application $\mathbf{r}_t$ indicates for all candidates d whether they are qualified to receive an interview call, but $\mathbf{c}_t$ reveals this information only for those candidates examined by the employer.

A second challenge lies in the fact that the examination vector $\mathbf{e}_t$ cannot be observed. This implies that a feedback value of $\mathbf{c}_t(d) = 0$ is ambiguous – it may either indicate lack of examination (i.e. $\mathbf{e}_t(d) = 0$) or negative feedback (i.e. $\mathbf{r}_t(d) = 0$). This would not be problematic if $\mathbf{e}_t$ was uniformly random, but which items get examined is strongly biased by the ranking $\boldsymbol{\sigma}_t$ presented to the user in the current iteration. Specifically, users are more likely to look at an item high in the ranking than at one that is lower down [32]. We model this position bias as a probability distribution on the examination vector

$$\mathbf{e}_t \sim \mathrm{P}(\mathbf{e} \,|\boldsymbol{\sigma}_t, \boldsymbol{x}_t, \mathbf{r}_t). \tag{3}$$

Most click models can be brought into this form [19]. For the simplicity of this paper, we merely use the Position-Based Model (PBM) [21]. It assumes that the marginal probability of examination $\mathbf{p}_t(d)$ for each item d depends only on the rank $\mathrm{rank}(d|\boldsymbol{\sigma})$ of d in the presented ranking $\boldsymbol{\sigma}$. Despite its simplicity, it was found that the PBM can capture the main effect of position bias accurately enough to be reliable in practice [5, 33, 47].

4.2 Evaluating Ranking Performance

We measure the quality of a ranking policy π by its utility to the users. Virtually all ranking metrics used in information retrieval define the utility $U(\boldsymbol{\sigma}|\,\mathbf{r})$ of a ranking $\boldsymbol{\sigma}$ as a function of the relevances of the individual items $\mathbf{r}$. In our case, these item-based relevances $\mathbf{r}$ represent which articles the user likes to read, or which candidates are qualified for an interview. A commonly used utility measure is the DCG [30]

$$U^{DCG}(\boldsymbol{\sigma}|\,\mathbf{r}) = \sum_{d\in\boldsymbol{\sigma}} \frac{\mathbf{r}(d)}{\log_2(1+\mathrm{rank}(d|\boldsymbol{\sigma}))},$$

or the NDCG when normalized by the DCG of the optimal ranking. Over a distribution of requests $\mathrm{P}(\boldsymbol{x},\mathbf{r})$, a ranking policy $\pi(\boldsymbol{x})$ is evaluated by its expected utility

$$U(\pi) \quad = \quad \int U(\pi(\boldsymbol{x})|\,\mathbf{r})\, d\,\mathrm{P}(\boldsymbol{x},\mathbf{r}). \tag{4}$$

4.3 Optimizing Ranking Performance

The user-facing goal of dynamic LTR is to converge to the policy $\pi^* = \mathrm{argmax}_\pi\, U(\pi)$ that maximizes utility. Even if we solve the problem of estimating $U(\pi)$ despite our lack of knowledge of $\mathbf{e}$, this maximization problem could be computationally challenging, since the space of ranking policies is exponential even when learning just a single global ranking. Fortunately, it is easy to show [39] that sorting-based policies

$$\pi(\boldsymbol{x}) \equiv \underset{d\in\mathcal{D}}{\mathrm{argsort}}\left[R(d|\boldsymbol{x})\right], \tag{5}$$

where

$$R(d|\boldsymbol{x}) = \int \mathbf{r}(d)\, d\,\mathrm{P}(\mathbf{r}\,|\boldsymbol{x}), \tag{6}$$

are optimal for virtually all $U(\boldsymbol{\sigma}|\,\mathbf{r})$ commonly used in IR (e.g. DCG). So, the problem lies in estimating the expected relevance $R(d|\boldsymbol{x})$ of each item d conditioned on $\boldsymbol{x}$. When learning a single global ranking, this further simplifies to estimating the expected average relevance $R(d) = \int \mathbf{r}(d)\, d\,\mathrm{P}(\mathbf{r},\boldsymbol{x})$ for each item d. The global ranking can then be derived via

$$\boldsymbol{\sigma} = \underset{d\in\mathcal{D}}{\mathrm{argsort}}\left[R(d)\right] \tag{7}$$

In Section 6, we will use techniques from causal inference and missing-data analysis to design unbiased and consistent estimators for $R(d|\boldsymbol{x})$ and $R(d)$ that only require access to the observed feedback $\mathbf{c}_t$.

5 FAIRNESS IN DYNAMIC LTR

While sorting by $R(d|\boldsymbol{x})$ (or $R(d)$ for global rankings) may provide optimal utility to the user, the introductory example has already illustrated that this ranking can be unfair. There is a growing body of literature to address this unfairness in ranking, and we now extend merit-based fairness [14, 42] to the dynamic LTR setting.

The key scarce resource that a ranking policy allocates among the items is exposure. Based on the model introduced in the previous section, we define the exposure of an item d as the marginal probability of examination $\mathbf{p}_t(d) = \mathrm{P}(\mathbf{e}_t(d) = 1|\boldsymbol{\sigma}_t, \boldsymbol{x}_t, \mathbf{r}_t)$. It is the probability that the user will see d and thus have the opportunity to read that article, buy that product, or interview that candidate. We discuss in Section 6 how to estimate $\mathbf{p}_t(d)$. Taking a group-based approach to fairness, we aggregate exposure by groups $\mathcal{G} = \{G_1, \ldots, G_m\}$.

$$Exp_t(G_i) \quad = \quad \frac{1}{|G_i|}\sum_{d\in G_i} \mathbf{p}_t(d). \tag{8}$$

These groups can be legally protected groups (e.g. gender, race), reflect some other structure (e.g. items sold by a particular seller), or simply put each item in its own group (i.e. individual fairness).

In order to formulate fairness criteria that relate exposure to merit, we define the merit of an item as its expected average relevance $R(d)$ and again aggregate over groups.

$$Merit(G_i) = \frac{1}{|G_i|}\sum_{d\in G_i} R(d) \tag{9}$$

In Section 6, we will discuss how to get unbiased estimates of $Merit(G_i)$ using the biased feedback data $\mathbf{c}_t$.

With these definitions in hand, we can address the types of disparities identified in Section 2. Specifically, we extend the Disparity of Treatment criterion of [42] to the dynamic ranking problem, using an amortized notion of fairness as in [14]. In particular, for any two groups G_i and G_j the disparity

$$D^E_\tau(G_i,G_j) = \frac{\frac{1}{\tau}\sum_{t=1}^{\tau} Exp_t(G_i)}{Merit(G_i)} - \frac{\frac{1}{\tau}\sum_{t=1}^{\tau} Exp_t(G_j)}{Merit(G_j)} \tag{10}$$

measures in how far amortized exposure over τ time steps was fulfilled. This **exposure-based fairness disparity** expresses in how far, averaged over all time steps, each group of items got exposure proportional to its relevance. The further the disparity is from zero, the greater is the violation of fairness. Note that other allocation strategies beyond proportionality could be implemented as well by using alternate definitions of disparity [42].

Exposure can also be allocated based on other fairness criteria, for example, a Disparity of Impact that a specific exposure allocation implies [42]. If we consider the feedback $\mathbf{c}_t$ (e.g. clicks, purchases, votes) as a measure of impact

$$Imp_t(G_i) \quad = \quad \frac{1}{|G_i|}\sum_{d\in G_i} \mathbf{c}_t(d), \tag{11}$$

then keeping the following disparity close to zero controls how exposure is allocated to make impact proportional to relevance.

$$D^I_\tau(G_i,G_j) = \frac{\frac{1}{\tau}\sum_{t=1}^{\tau} Imp_t(G_i)}{Merit(G_i)} - \frac{\frac{1}{\tau}\sum_{t=1}^{\tau} Imp_t(G_j)}{Merit(G_j)} \tag{12}$$

We refer to this as the **impact-based fairness disparity**. In Section 7 we will derive a controller that drives such exposure and impact disparities to zero.

6 UNBIASED ESTIMATORS

To be able to implement the ranking policies in Equation (5) and the fairness disparities in Equations (10) and (12), we need accurate estimates of the position bias $\mathbf{p}_t$, the expected conditional relevances $R(d|\boldsymbol{x})$, and the expected average relevances $R(d)$. We consider these estimation problems in the following.

6.1 Estimating the Position Bias

Learning a model for $\mathbf{p}_t$ is not part of our dynamic LTR problem, as the position-bias model is merely an input to our dynamic LTR algorithms. Fortunately, several techniques for estimating position-bias models already exist in the literature [5, 23, 33, 47], and we are agnostic to which of these is used. In the simplest case, the examination probabilities $\mathbf{p}_t(d)$ only depend on the rank of the item in $\boldsymbol{\sigma}$, analogous to a Position-Based Click Model [21] with a fixed probability for each rank. It was shown in [5, 33, 47] how these position-based probabilities can be estimated from explicit and implicit swap interventions. Furthermore, it was shown in [23] how the contextual features $\boldsymbol{x}$ about the users and query can be incorporated in a neural-network based propensity model, allowing it to capture that certain users may explore further down the ranking for some queries. Once any of these propensity models are learned, they can be applied to predict $\mathbf{p}_t$ for any new query $\boldsymbol{x}_t$ and ranking $\boldsymbol{\sigma}_t$.

6.2 Estimating Conditional Relevances

The key challenge in estimating $R(d|\boldsymbol{x})$ from Equation (6) lies in our inability to directly observe the true relevances $\mathbf{r}_t$. Instead, the only data we have is the partial and biased feedback $\mathbf{c}_t$. To overcome this problem, we take an approach inspired by [33] and extend it to the dynamic ranking setting. The key idea is to correct for the selection bias with which relevance labels are observed in $\mathbf{c}_t$ using techniques from survey sampling and causal inference [28, 29]. However, unlike the ordinal estimators proposed in [33], we need cardinal relevance estimates since our fairness disparities are cardinal in nature. We, therefore, propose the following cardinal relevance estimator.

The key idea behind this estimator lies in a training objective that only uses $\mathbf{c}_t$, but that in expectation is equivalent to a least-squares objective that has access to $\mathbf{r}_t$. To start the derivation, let's consider how we would estimate $R(d|\boldsymbol{x})$, if we had access to the relevance labels $(\mathbf{r}_1, ..., \mathbf{r}_\tau)$ of the previous τ time steps. A straightforward solution would be to solve the following least-squares objective for a given regression model $\hat{R}^{\mathrm{w}}(d|\boldsymbol{x}_t)$ (e.g. a neural network), where w are the parameters of the model.

$$\mathcal{L}^{\mathbf{r}}(w) \quad = \quad \sum_{t=1}^{\tau}\sum_{d}\left(\mathbf{r}_t(d) - \hat{R}^{\mathrm{w}}(d|\boldsymbol{x}_t)\right)^2 \tag{13}$$

The minimum w^* of this objective is the least-squares regression estimator of $R(d|\boldsymbol{x}_t)$. Since the $(\mathbf{r}_1, ..., \mathbf{r}_\tau)$ are not available, we define an asymptotically equivalent objective that merely uses the biased feedback $(\mathbf{c}_1, ..., \mathbf{c}_\tau)$. The new objective corrects for the position bias using Inverse Propensity Score (IPS) weighting [28, 29], where the position bias $(\mathbf{p}_1, ..., \mathbf{p}_\tau)$ takes the role of the missingness model.

$$\mathcal{L}^{\mathbf{c}}(w) \;=\; \sum_{t=1}^{\tau}\sum_{d}\hat{R}^{\mathrm{w}}(d|\boldsymbol{x}_t)^2 + \frac{\mathbf{c}_t(d)}{\mathbf{p}_t(d)}(\mathbf{c}_t(d) - 2\hat{R}^{\mathrm{w}}(d|\boldsymbol{x}_t)) \tag{14}$$

We denote the regression estimator defined by the minimum of this objective as $\hat{R}^{\mathrm{Reg}}(d|\boldsymbol{x}_t)$. The regression objective in (14) is unbiased, meaning that its expectation is equal to the regression objective $\mathcal{L}^{\mathbf{r}}(w)$ that uses the unobserved true relevances $(\mathbf{r}_1, ..., \mathbf{r}_\tau)$.

$$\begin{aligned}
&\mathbb{E}_{\mathbf{e}}\left[\mathcal{L}^{\mathbf{c}}(w)\right] \\
&= \sum_{t=1}^{\tau}\sum_{d}\sum_{\mathbf{e}_t(d)}\left[\hat{R}^{\mathrm{w}}(d|\boldsymbol{x}_t)^2 + \frac{\mathbf{c}_t(d)}{\mathbf{p}_t(d)}(\mathbf{c}_t(d) - 2\hat{R}^{\mathrm{w}}(d|\boldsymbol{x}_t))\right]\mathrm{P}(\mathbf{e}_t(d)|\boldsymbol{\sigma}_t, \boldsymbol{x}_t) \\
&= \sum_{t=1}^{\tau}\sum_{d}\hat{R}^{\mathrm{w}}(d|\boldsymbol{x}_t)^2 + \frac{1}{\mathbf{p}_t(d)}\,\mathbf{r}_t(d)(\mathbf{r}_t(d) - 2\hat{R}^{\mathrm{w}}(d|\boldsymbol{x}_t))\,\mathbf{p}_t(d) \\
&= \sum_{t=1}^{\tau}\sum_{d}\hat{R}^{\mathrm{w}}(d|\boldsymbol{x}_t)^2 + \mathbf{r}_t(d)^2 - 2\,\mathbf{r}_t(d)\hat{R}^{\mathrm{w}}(d|\boldsymbol{x}_t) \\
&= \sum_{t=1}^{\tau}\sum_{d}\left(\mathbf{r}_t(d) - \hat{R}^{\mathrm{w}}(d|\boldsymbol{x}_t)\right)^2 \\
&= \mathcal{L}^{\mathbf{r}}(w)
\end{aligned}$$

Line 2 formulates the expectation in terms of the marginal exposure probabilities $\mathrm{P}(\mathbf{e}_t(d)|\boldsymbol{\sigma}_t, \boldsymbol{x}_t)$, which decomposes the expectation as the objective is additive in d. Note that $\mathrm{P}(\mathbf{e}_t(d) = 1|\boldsymbol{\sigma}_t, \boldsymbol{x}_t)$ is therefore equal to $\mathbf{p}_t(d)$ under our exposure model. Line 3 substitutes $\mathbf{c}_t(d) = \mathbf{e}_t(d)\,\mathbf{r}_t(d)$ and simplifies the expression, since $\mathbf{e}_t(d)\,\mathbf{r}_t(d) = 0$ whenever the user is not exposed to an item. Note that the propensities $\mathbf{p}_t(\boldsymbol{\sigma})$ for the exposed items now cancel, as long as they are bounded away from zero – meaning that all items have some probability of being found by the user. In case users do not naturally explore low enough in the ranking, active interventions can be used to stochastically promote items in order to ensure non-zero examination propensities (e.g. [27]). Note that unbiasedness holds for any sequence of $(\boldsymbol{x}_1, \mathbf{r}_1, \boldsymbol{\sigma}_1)..., (\boldsymbol{x}_T, \mathbf{r}_T, \boldsymbol{\sigma}_T)$, no matter how complex the dependencies between the rankings $\boldsymbol{\sigma}_t$ are.

Beyond this proof of unbiasedness, it is possible to use standard concentration inequalities to show that $\mathcal{L}^{\mathbf{c}}(w)$ converges to $\mathcal{L}^{\mathbf{r}}(w)$ as the size τ of the training sequence increases. Thus, under standard conditions on the capacity for uniform convergence, it is possible to show convergence of the minimizer of $\mathcal{L}^{\mathbf{c}}(w)$ to the least-squares regressor as the size τ of the training sequence increases. We will use this regression objective to learn neural-network rankers in Section 8.2.

6.3 Estimating Average Relevances

The conditional relevances $R(d|\boldsymbol{x})$ are used in the ranking policies from Equation (5). But when defining merit in Equation (9) for the fairness disparities, the average relevance $R(d)$ is needed. Furthermore, $R(d)$ serves as the ranking criterion for global rankings in Equation (7). While we could marginalize $R(d|\boldsymbol{x})$ over $\mathrm{P}(\boldsymbol{x})$ to derive $R(d)$, we argue that the following is a more direct way to get an unbiased estimate.

$$\hat{R}^{\mathrm{IPS}}(d) \quad = \quad \frac{1}{\tau}\sum_{t=1}^{\tau}\frac{\mathbf{c}_t(d)}{\mathbf{p}_t(d)}. \tag{15}$$

The following shows that this estimator is unbiased as long as the propensities are bounded away from zero.

$$\begin{aligned}\mathbb{E}_{\mathbf{e}}\left[\hat{R}^{\text{IPS}}(d)\right] &= \frac{1}{\tau}\sum_{t=1}^{\tau}\sum_{\mathbf{e}_t(d)}\frac{\mathbf{e}_t(d)\,\mathbf{r}_t(d)}{\mathbf{p}_t(d)}\,\mathrm{P}(\mathbf{e}_t(d)|\boldsymbol{\sigma}_t,\boldsymbol{x}_t)\\ &= \frac{1}{\tau}\sum_{t=1}^{\tau}\frac{\mathbf{r}_t(d)}{\mathbf{p}_t(d)}\,\mathbf{p}_t(d)\\ &= \frac{1}{\tau}\sum_{t=1}^{\tau}\mathbf{r}_t(d)\\ &= R(d)\end{aligned}$$

In the following experiments, we will use this estimator whenever a direct estimate of $R(d)$ is needed for the fairness disparities or as a global ranking criterion.

7 DYNAMICALLY CONTROLLING FAIRNESS

Given the formalization of the dynamic LTR problem, our definition of fairness, and our derivation of estimators for all relevant parameters, we are now in the position to tackle the problem of ranking while enforcing the fairness conditions. We view this as a control problem since we need to be robust to the uncertainty in the estimates $\hat{R}(d|\boldsymbol{x})$ and $\hat{R}(d)$ at the beginning of the learning process. Specifically, we propose a controller that is able to make up for the initial uncertainty as these estimates converge during the learning process.

Following our pairwise definitions of amortized fairness from Section 5, we quantify by how much fairness between all classes is violated using the following overall disparity metric.

$$\overline{D}_{\tau} = \frac{2}{m(m-1)}\sum_{i=0}^{m}\sum_{j=i+1}^{m}\left|D_{\tau}(G_i,G_j)\right| \tag{16}$$

This metric can be instantiated with the disparity $D^{E}_{\tau}(G_i,G_j)$ from Equation (10) for exposure-based fairness, or $D^{I}_{\tau}(G_i,G_j)$ from Equation (12) for impact-based fairness. Since optimal fairness is achieved for $\overline{D}_{\tau}=0$, we seek to minimize $\overline{D}_{\tau}$.

To this end, we now derive a method we call *FairCo*, which takes the form of a Proportional Controller (a.k.a. P-Controller) [12]. A P-controller is a widely used control-loop mechanism that applies feedback through a correction term that is proportional to the error. In our application, the error corresponds to the violation of our amortized fairness disparity from Equations (10) and (12). Specifically, for any set of disjoint groups $\mathcal{G}=\{G_1,\ldots,G_m\}$, the error term of the controller for any item d is defined as

$$\forall G\in\mathcal{G}\;\forall d\in G:\mathbf{err}_{\tau}(d)=(\tau-1)\cdot\max_{G_i}\left(\hat{D}_{\tau-1}(G_i,G)\right).$$

The error term $\mathbf{err}_{\tau}(G)$ is zero for the group that already has the maximum exposure/impact w.r.t. its merit. For items in the other groups, the error term grows with increasing disparity.

Note that the disparity $\hat{D}_{\tau-1}(G_i,G)$ in the error term uses the estimated $\hat{Merit}(G)$ from Equation (15), which converges to $Merit(G)$ as the sample size τ increases. To avoid division by zero, $\hat{Merit}(G)$ can be set to some minimum constant.

We are now in a position to state the FairCo ranking policy as

$$\text{FairCo:}\qquad \boldsymbol{\sigma}_{\tau}=\underset{d\in\mathcal{D}}{\operatorname{argsort}}\left(\hat{R}(d|\boldsymbol{x})+\lambda\,\mathbf{err}_{\tau}(d)\right). \tag{17}$$

When the exposure-based disparity $\hat{D}^{E}_{\tau-1}(G_i,G)$ is used in the error term, we refer to this policy as FairCo(Exp). If the impact-based disparity $\hat{D}^{I}_{\tau-1}(G_i,G)$ is used, we refer to it as FairCo(Imp).

Like the policies in Section 4.3, FairCo is a sort-based policy. However, the sorting criterion is a combination of relevance $\hat{R}(d|\boldsymbol{x})$ and an error term representing the fairness violation. The idea behind FairCo is that the error term pushes the items from the underexposed groups upwards in the ranking. The parameter λ can be chosen to be any positive constant. While any choice of λ leads to asymptotic convergence as shown by the theorem below for exposure fairness, a suitable choice of λ can have influence on the finite-sample behavior of FairCo: a higher λ can lead to an oscillating behavior, while a smaller λ makes the convergence smoother but slower. We explore the role of λ in the experiments, but find that keeping it fixed at $\lambda=0.01$ works well across all of our experiments. Another key quality of FairCo is that it is agnostic to the choice of error metric, and we conjecture that it can easily be adapted to other types of fairness disparities. Furthermore, it is easy to implement and it is very efficient, making it well suited for practical applications.

To illustrate the theoretical properties of FairCo, we now analyze its convergence for the case of exposure-based fairness. To disentangle the convergence of the estimator for $\hat{Merit}(G)$ from the convergence of FairCo, consider a time point τ_0 where $\hat{Merit}(G)$ is already close to $Merit(G)$ for all $G\in\mathcal{G}$. We can thus focus on the question whether FairCo can drive $\overline{D}^{E}_{\tau}$ to zero starting from any unfairness that may have persisted at time τ_0. To make this problem well-posed, we need to assume that exposure is not available in overabundance, otherwise it may be unavoidable to give some groups more exposure than they deserve even if they are put at the bottom of the ranking. A sufficient condition for excluding this case is to only consider problems for which the following is true: for all pairs of groups G_i, G_j, if G_i is ranked entirely above G_j at any time point t, then

$$\frac{Exp_t(G_i)}{\hat{Merit}(G_i)}\geq\frac{Exp_t(G_j)}{\hat{Merit}(G_j)}. \tag{18}$$

Intuitively, the condition states that ranking G_i ahead of G_j reduces the disparity if G_i has been underexposed in the past. We can now state the following theorem.

Theorem 7.1. *For any set of disjoint groups $\mathcal{G}=\{G_1,\ldots,G_m\}$ with any fixed target merits $\hat{Merit}(G_i)>0$ that fulfill* (18)*, any relevance model $\hat{R}(d|\boldsymbol{x})\in[0,1]$, any exposure model $\mathbf{p}_t(d)$ with $0\leq\mathbf{p}_t(d)\leq\mathbf{p}_{\max}$, and any value $\lambda>0$, running FairCo(Exp) from time τ_0 will always ensure that the overall disparity $\overline{D}^{E}_{\tau}$ with respect to the target merits converges to zero at a rate of $\mathcal{O}\left(\frac{1}{\tau}\right)$, no matter how unfair the exposures $\frac{1}{\tau_0}\sum_{t=1}^{\tau_0}Exp_t(G_j)$ up to τ_0 have been.*

The proof of the theorem is included in Appendix B. Note that this theorem holds for any time point τ_0, even if the estimated merits change substantially up to τ_0. So, once the estimated merits have converged to the true merits, FairCo(Exp) will ensure that the amortized disparity $\overline{D}_\tau^E$ converges to zero as well.

8 EMPIRICAL EVALUATION

In addition to the theoretical justification of our approach, we also conducted an empirical evaluation[1]. We first present experiments on a semi-synthetic news dataset to investigate different aspects of the proposed methods under controlled conditions. After that we evaluate the methods on real-world movie preference data for external validity.

8.1 Robustness Analysis on News Data

To be able to evaluate the methods in a variety of specifically designed test settings, we created the following simulation environment from articles in the Ad Fontes Media Bias dataset[2]. It simulates a dynamic ranking problem on a set of news articles belonging to two groups G_{left} and G_{right} (e.g. left-leaning and right-leaning news articles).

In each trial, we sample a set of 30 news articles $\mathcal{D}$. For each article, the dataset contains a polarity value ρ^d that we rescale to the interval between -1 and 1, while the user polarities are simulated. Each user has a polarity that is drawn from a mixture of two normal distributions clipped to $[-1, 1]$

$$\rho^{u_t} \sim \text{clip}_{[-1,1]}\left(p_{neg}\mathcal{N}(-0.5, 0.2) + (1 - p_{neg})\mathcal{N}(0.5, 0.2)\right) \quad (19)$$

where p_{neg} is the probability of the user to be left-leaning (mean=-0.5). We use $p_{neg} = 0.5$ unless specified. In addition, each user has an openness parameter $o^{u_t} \sim \mathcal{U}(0.05, 0.55)$, indicating on the breadth of interest outside their polarity. Based on the polarities of the user u_t and the item d, the true relevance is drawn from the Bernoulli distribution

$$\mathbf{r}_t(d) \sim \text{Bernoulli}\left[p = exp\left(\frac{-(\rho^{u_t} - \rho^d)^2}{2(o^{u_t})^2}\right)\right].$$

As the model of user behavior, we use the Position-based click model (PBM [19]), where the marginal probability that user u_t examines an article only depends only on its position. We choose an exposure drop-off analogous to the gain function in DCG as

$$\mathbf{p}_t(d) = \frac{1}{\log_2(\text{rank}(d|\boldsymbol{\sigma}_t) + 1)}. \quad (20)$$

The remainder of the simulation follows the dynamic ranking setup. At each time step t a user u_t arrives to the system, the algorithm presents an unpersonalized ranking $\boldsymbol{\sigma}_t$, and the user provides feedback $\mathbf{c}_t$ according to $\mathbf{p}_t$ and $\mathbf{r}_t$. The algorithm only observes $\mathbf{c}_t$ and not $\mathbf{r}_t$.

To investigate group-fairness, we group the items according to their polarity, where items with a polarity $\rho^d \in [-1, 0)$ belong to the *left-leaning* group G_{left} and items with a polarity $\rho^d \in [0, 1]$ belong to the *right-leaning* group G_{right}.

We measure ranking quality by the average cumulative NDCG $\frac{1}{\tau}\sum_{t=1}^{\tau} U^{DCG}(\boldsymbol{\sigma}_t|\mathbf{r}_t)$ over all the users up to time τ. We measure Exposure Unfairness via $\overline{D}_\tau^E$ and Impact Unfairness via $\overline{D}_\tau^I$ as defined in Equation (16).

In all news experiments, we learn a global ranking and compare the following methods.

Naive: Rank by the sum of the observed feedback $\mathbf{c}_t$.

D-ULTR(Glob): Dynamic LTR by sorting via the unbiased estimates $\hat{R}^{\text{IPS}}(d)$ from Eq. (15).

FairCo(Imp): Fairness controller from Eq. (17) for impact fairness.

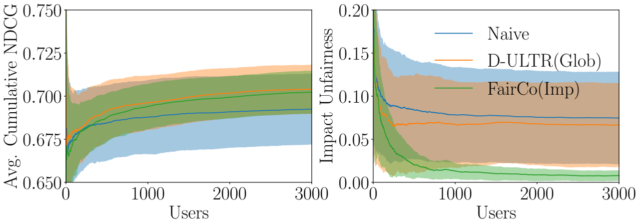

Figure 1: Convergence of NDCG (left) and Unfairness (right) as the number of users increases. (100 trials)

8.1.1 Can FairCo reduce unfairness while maintaining good ranking quality? This is the key question in evaluating FairCo, and Figure 1 shows how NDCG and Unfairness converge for Naive, D-ULTR(Glob), and FairCo(Imp). The plots show that Naive achieves the lowest NDCG and that its unfairness remains high as the number of user interactions increases. D-ULTR(Glob) achieve the best NDCG, as predicted by the theory, but its unfairness is only marginally better than that of Naive. Only FairCo manages to substantially reduce unfairness, and this comes only at a small decrease in NDCG compared to D-ULTR(Glob).

The following questions will provide further insight into these results, evaluating the components of the FairCo and exploring its robustness.

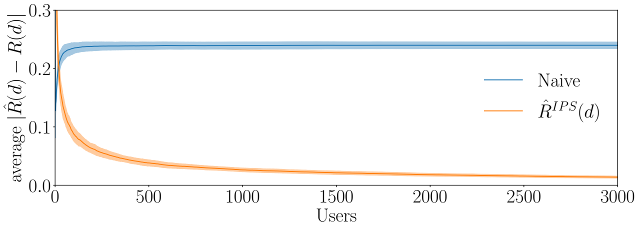

Figure 2: Error of relevance estimators as the number of users increases ($|\mathcal{D}| = 30$, 10 trials)

8.1.2 Do the unbiased estimates converge to the true relevances? The first component of FairCo we evaluate is the unbiased IPS estimator $\hat{R}^{\text{IPS}}(d)$ from Equation (15). Figure 1 shows the absolute difference between the estimated global relevance and true global relevance for $\hat{R}^{\text{IPS}}(d)$ and the estimator used in the Naive. While the error for Naive stagnates at around 0.25, the estimation error of $\hat{R}^{\text{IPS}}(d)$ approaches zero as the number of users increases. This verifies that IPS eliminates the effect of position bias and learns accurate estimates of the true expected relevance for each news article so that we can use them for the fairness and ranking criteria.

[1] The implementation is available at https://github.com/MarcoMorik/Dynamic-Fairness.

[2] https://www.adfontesmedia.com/interactive-media-bias-chart/

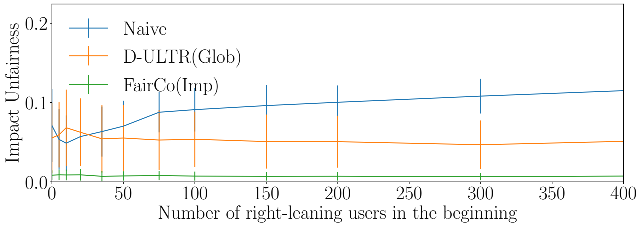

Figure 3: The effect of a block of right-leaning users on the Unfairness of Impact. (50 trials, 3000 users)

8.1.3 Does FairCo overcome the rich-get-richer dynamic? The illustrating example in Section 2 argues that naively ranking items is highly sensitive to the initial conditions (e.g. which items get the first clicks), leading to a rich-get-richer dynamic. We now test whether FairCo overcomes this problem. In particular, we adversarially modify the user distribution so that the first x users are right-leaning ($p_{neg} = 0$), followed by x left-leaning users ($p_{neg} = 1$), before we continue with a balanced user distribution ($p_{neg} = 0.5$). Figure 3 shows the unfairness after 3000 user interactions. As expected, Naive is the most sensitive to the head-start that the right-leaning articles are getting. D-ULTR(Glob) fares better and its unfairness remains constant (but high) independent of the initial user distribution since the unbiased estimator $\hat{R}^{\text{IPS}}(d)$ corrects for the presentation bias so that the estimates still converge to the true relevance. FairCo inherits this robustness to initial conditions since it uses the same $\hat{R}^{\text{IPS}}(d)$ estimator, and its active control for unfairness makes it the only method that achieves low unfairness across the whole range.

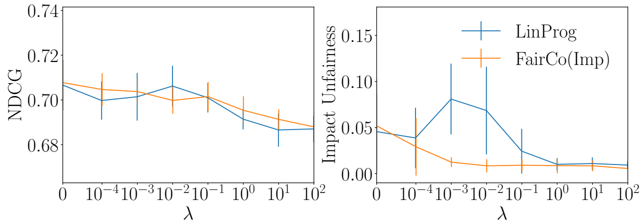

Figure 4: Comparing the LP Baseline and the P-Controller in terms of NDCG (left) and Unfairness (right) for different values of λ. (15 trials, 3000 users)

8.1.4 How effective is the FairCo compared to a more expensive Linear-Programming Baseline? As a baseline, we adapt the linear programming method from [42] to the dynamic LTR setting to minimize the amortized fairness disparities that we consider in this work. The method uses the current relevance and disparity estimates to solve a linear programming problem whose solution is a stochastic ranking policy that satisfies the fairness constraints in expectation at each τ. The details of this method are described in Appendix A. Figure 4 shows NDCG and Impact Unfairness after 3000 users averaged over 15 trials for both LinProg and FairCo for different values of their hyperparameter λ. For $\lambda = 0$, both methods reduce to D-ULTR(Glob) and we can see that their solutions are unfair. As λ increases, both methods start enforcing fairness at the expense of NDCG. In these and other experiments, we found no evidence that the LinProg baseline is superior to FairCo. However, LinProg is substantially more expensive to compute, which makes FairCo preferable in practice.

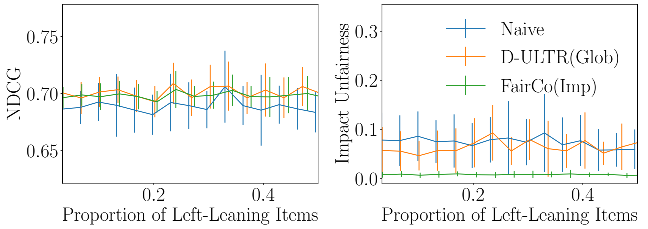

Figure 5: NDCG (left) and Unfairness (right) for varying proportion of G_{left} (20 trials, 3000 users)

8.1.5 Is FairCo effective for different group sizes? In this experiment, we vary asymmetry of the polarity within the set of 30 news articles, ranging from $G_{\text{left}} = 1$ to $G_{\text{left}} = 15$ news articles. For each group size, we run 20 trials for 3000 users each. Figure 5 shows that regardless of the group ratio, FairCo reduces unfairness for the whole range while maintaining NDCG. This is in contrast to Naive and D-ULTR(Glob), which suffer from high unfairness.

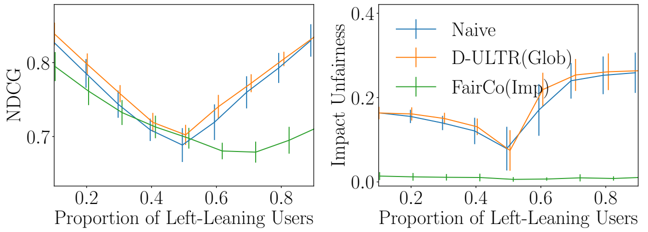

Figure 6: NDCG (left) and Unfairness (right) with varying user distributions. (20 trials, 3000 users)

8.1.6 Is FairCo effective for different user distributions? Finally, to examine the robustness to varying user distributions, we control the polarity distribution of the users by varying p_{neg} in Equation (19). We run 20 trials each on 3000 users. In Figure 6, observe that Naive and D-ULTR(Glob) suffer from high unfairness when there is a large imbalance between the minority and the majority group, while FairCo is able to control the unfairness in all settings.

8.2 Evaluation on Real-World Preference Data

To evaluate our method on a real-world preference data, we adopt the ML-20M dataset [25]. We select the five production companies with the most movies in the dataset — *MGM, Warner Bros, Paramount, 20th Century Fox, Columbia.* These production companies form the groups for which we aim to ensure fairness of exposure. To exclude movies with only a few ratings and have a diverse user population, from the set of 300 most rated movies by these production companies, we select 100 movies with the highest standard deviation in the rating across users. For the users, we select 10^4 users who have rated the most number of the chosen movies. This leaves us with a partially filled ratings matrix with 10^4 users and 100 movies. To avoid missing data for the ease of evaluation, we use an off-the-shelf matrix factorization algorithm[3] to fill in the missing entries. We then normalize the ratings to $[0, 1]$ by apply a Sigmoid function centered at rating $b = 3$ with slope $a = 10$. These serve as relevance probabilities where higher star ratings correspond to a higher likelihood of positive feedback. Finally, for each trial we obtain a binary relevance matrix by drawing a Bernoulli sample for each user and movie pair with these probabilities. We use the user embeddings from the matrix factorization model as the user features $\boldsymbol{x}_t$.

In the following experiments we use FairCo to learn a sequence of ranking policies $\pi_t(\boldsymbol{x})$ that are personalized based on $\boldsymbol{x}$. The goal is to maximize NDCG while providing fairness of exposure to the production companies. User interactions are simulated analogously to the previous experiments. At each time step t, we sample a user $\boldsymbol{x}_t$ and the ranking algorithm presents a ranking of the 100 movies. The user follows the position-based model from Equation (20) and reveal $\mathbf{c}_t$ accordingly.

For the conditional relevance model $\hat{R}^{\mathrm{Reg}}(d|\boldsymbol{x})$ used by FairCo and D-ULTR, we use a one hidden-layer neural network that consists of $D = 50$ input nodes fully connected to 64 nodes in the hidden layer with ReLU activation, which is connected to 100 output nodes with Sigmoid to output the predicted probability of relevance of each movie. Since training this network with less than 100 observations is unreliable, we use the global ranker D-ULTR(Glob) for the first 100 users. We then train the network at $\tau = 100$ users, and then update the network after every 10 users on all previously collected feedback i.e. $\mathbf{c}_1, \ldots, \mathbf{c}_\tau$ using the unbiased regression objective, $\mathcal{L}^{\mathrm{c}}(w)$, from Eq. (14) with the Adam optimizer [34].

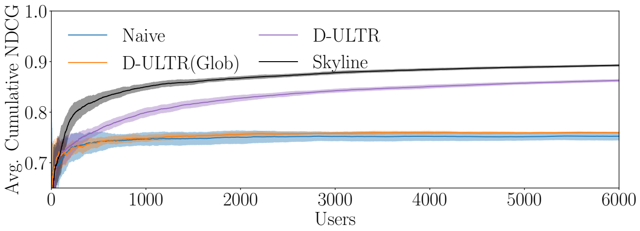

Figure 7: Comparing the NDCG of personalized and non-personalized rankings on the Movie data. (10 trials)

8.2.1 Does personalization via unbiased objective improve NDCG?. We first evaluate whether training a personalized model using the de-biased $\hat{R}^{\mathrm{Reg}}(d|\boldsymbol{x})$ regression estimator improves ranking performance over a non-personalized model. Figure 7 shows that ranking by $\hat{R}^{\mathrm{Reg}}(d|\boldsymbol{x})$ (i.e. D-ULTR) provides substantially higher NDCG than the unbiased global ranking D-ULTR(Glob) and the Naive ranking. To get an upper bound on the performance of the personalization models, we also train a Skyline model using the (in practice unobserved) true relevances $\mathbf{r}_t$ with the least-squares objective from Eq. (13). Even though the unbiased regression estimator $\hat{R}^{\mathrm{Reg}}(d|\boldsymbol{x})$ only has access to the partial feedback $\mathbf{c}_t$, it tracks the performance of Skyline. As predicted by the theory, they appear to converge asymptotically.

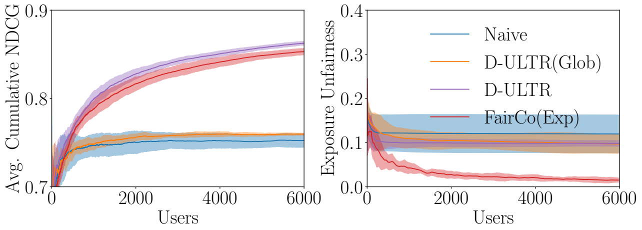

Figure 8: NDCG (left) and Exposure Unfairness (right) on the Movie data as the number of user interactions increases. (10 trials)

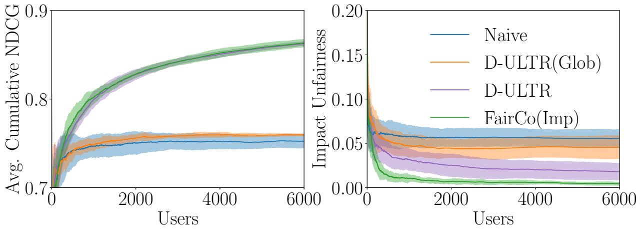

Figure 9: NDCG (left) and Impact Unfairness (right) on the Movie data as the number of user interactions increases. (10 trials)

8.2.2 Can FairCo reduce unfairness? Figure 8 shows that FairCo(Exp) can effectively control Exposure Unfairness, unlike the other methods that do not actively consider fairness. Similarly, Figure 9 shows that FairCo(Imp) is effective at controlling Impact Unfairness. As expected, the improvement in fairness comes at a reduction in NDCG, but this reduction is small.

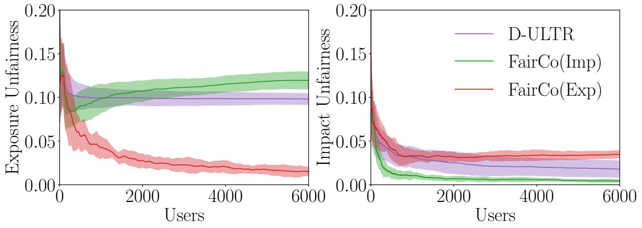

Figure 10: Unfairness of Exposure (left) and Unfairness of Impact (right) for the personalized controller optimized for either Exposure or Impact. (10 trials)

8.2.3 How different are exposure and impact fairness? Figure 10 evaluates how an algorithm that optimizes Exposure Fairness performs in terms of Impact Fairness and vice versa. The plots show that the two criteria achieve different goals and that they are substantially different. In fact, optimizing for fairness in impact can even increase the unfairness in exposure, illustrating that the choice of criterion needs to be grounded in the requirements of the application.

[3] Surprise library (http://surpriselib.com/) for SVD with `biased=False` and `D=50`

9 CONCLUSIONS

We identify how biased feedback and uncontrolled exposure allocation can lead to unfairness and undesirable behavior in dynamic LTR. To address this problem, we propose FairCo, which is able to adaptively enforce amortized merit-based fairness constraints even though their underlying relevances are still being learned. The algorithm is robust to presentation bias and thus does not exhibit rich-get-richer dynamics. Finally, FairCo is easy to implement and computationally efficient, which makes it well suited for practical applications.

ACKNOWLEDGMENTS

This research was supported in part by NSF Awards IIS-1901168 and a gift from Workday. All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

REFERENCES

[1] Himan Abdollahpouri, Gediminas Adomavicius, Robin Burke, Ido Guy, Dietmar Jannach, Toshihiro Kamishima, Jan Krasnodebski, and Luiz Pizzato. 2019. Beyond Personalization: Research Directions in Multistakeholder Recommendation. *arXiv preprint arXiv:1905.01986* (2019).

[2] Himan Abdollahpouri, Robin Burke, and Bamshad Mobasher. 2017. Controlling popularity bias in learning-to-rank recommendation. In *ACM RecSys*.

[3] Lada A Adamic and Bernardo A Huberman. 2000. Power-law distribution of the world wide web. *Science* (2000).

[4] A. Agarwal, K. Takatsu, I. Zaitsev, and T. Joachims. 2019. A General Framework for Counterfactual Learning-to-Rank. In *SIGIR*.

[5] A. Agarwal, I. Zaitsev, Xuanhui Wang, Cheng Li, M. Najork, and T. Joachims. 2019. Estimating Position Bias Without Intrusive Interventions. In *WSDM*.

[6] Qingyao Ai, Keping Bi, Cheng Luo, Jiafeng Guo, and W Bruce Croft. 2018. Unbiased learning to rank with unbiased propensity estimation. In *SIGIR*.

[7] Michael Ekstrand Sebastian Kohlmeier Asia Biega, Fernando Diaz. 2019. TREC Fair Ranking Track. https://fair-trec.github.io/ [Online; accessed 08-14-2019].

[8] Ricardo Baeza-Yates. 2018. Bias on the Web. *Commun. ACM* (2018).

[9] Solon Barocas, Moritz Hardt, and Arvind Narayanan. 2018. Fairness and Machine Learning. (2018).

[10] Solon Barocas and Andrew D Selbst. 2016. Big data's disparate impact. *Calif. L. Rev.* (2016).

[11] Michael A Beam. 2014. Automating the news: How personalized news recommender system design choices impact news reception. *Communication Research* (2014).

[12] B Wayne Bequette. 2003. *Process control: modeling, design, and simulation.* Prentice Hall Professional.

[13] Alex Beutel, Jilin Chen, Tulsee Doshi, Hai Qian, Li Wei, Yi Wu, Lukasz Heldt, Zhe Zhao, Lichan Hong, Ed H. Chi, and Cristos Goodrow. 2019. Fairness in Recommendation Ranking through Pairwise Comparisons. In *ACM SIGKDD*.

[14] Asia J Biega, Krishna P Gummadi, and Gerhard Weikum. 2018. Equity of Attention: Amortizing Individual Fairness in Rankings. In *SIGIR*.

[15] Garrett Birkhoff. 1940. *Lattice theory.* American Mathematical Soc.

[16] Christopher JC Burges. 2010. From Ranknet to Lambdarank to Lambdamart: An overview. *Learning* (2010).

[17] L Elisa Celis, Damian Straszak, and Nisheeth K Vishnoi. 2017. Ranking with fairness constraints. *arXiv preprint arXiv:1704.06840* (2017).

[18] Nicolò Cesa-Bianchi and Gábor Lugosi. 2006. *Prediction, learning, and games.* Cambridge University Press.

[19] Aleksandr Chuklin, Ilya Markov, and Maarten de Rijke. 2015. Click models for web search. *Synthesis Lectures on Information Concepts, Retrieval, and Services* (2015).

[20] Giovanni Luca Ciampaglia, Azadeh Nematzadeh, Filippo Menczer, and Alessandro Flammini. 2018. How algorithmic popularity bias hinders or promotes quality. *Scientific reports* (2018).

[21] Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. 2008. An experimental comparison of click position-bias models. In *WSDM*.

[22] Danielle Ensign, Sorelle A Friedler, Scott Neville, Carlos Scheidegger, and Suresh Venkatasubramanian. 2018. Runaway Feedback Loops in Predictive Policing. In *Conference of Fairness, Accountability, and Transparency*.

[23] Zhichong Fang, A. Agarwal, and T. Joachims. 2019. Intervention Harvesting for Context-Dependent Examination-Bias Estimation. In *SIGIR*.

[24] Fabrizio Germano, Vicenç Gómez, and Gaël Le Mens. 2019. The few-get-richer: a surprising consequence of popularity-based rankings. *arXiv preprint arXiv:1902.02580* (2019).

[25] F Maxwell Harper and Joseph A Konstan. 2015. The movielens datasets: History and context. *ACM TIIS* (2015).

[26] Herbrich, Graepel, and Obermayer. 2000. Large Margin Ranking Boundaries for Ordinal Regression. In *Advances in Large Margin Classifiers*.

[27] Katja Hofmann, Shimon Whiteson, and Maarten de Rijke. 2013. Balancing exploration and exploitation in listwise and pairwise online learning to rank for information retrieval. *Information Retrieval* (2013).

[28] Daniel G Horvitz and Donovan J Thompson. 1952. A generalization of sampling without replacement from a finite universe. *Journal of the American statistical Association* (1952).

[29] Guido W Imbens and Donald B Rubin. 2015. *Causal inference in statistics, social, and biomedical sciences.* Cambridge University Press.

[30] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *TOIS* (2002).

[31] T. Joachims. 2002. Optimizing Search Engines Using Clickthrough Data. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*. 133–142.

[32] T. Joachims, L. Granka, Bing Pan, H. Hembrooke, F. Radlinski, and G. Gay. 2007. Evaluating the Accuracy of Implicit Feedback from Clicks and Query Reformulations in Web Search. *ACM TOIS* (2007).

[33] T. Joachims, A. Swaminathan, and T. Schnabel. 2017. Unbiased Learning-to-Rank with Biased Feedback. In *WSDM*.

[34] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[35] Shuai Li, Tor Lattimore, and Csaba Szepesvári. 2018. Online Learning to Rank with Features. *arXiv preprint arXiv:1810.02567* (2018).

[36] Lydia Liu, Sarah Dean, Esther Rolf, Max Simchowitz, and Moritz Hardt. 2018. Delayed Impact of Fair Machine Learning. In *ICML*.

[37] F. Radlinski and T. Joachims. 2006. Minimally Invasive Randomization for Collecting Unbiased Preferences from Clickthrough Logs. In *AAAI*. 1406–1412.

[38] F. Radlinski, R. Kleinberg, and T. Joachims. 2008. Learning Diverse Rankings with Multi-Armed Bandits. In *ICML*.

[39] Stephen E Robertson. 1977. The probability ranking principle in IR. *Journal of documentation* (1977).

[40] M. J. Salganik, P. Sheridan Dodds, and D. J. Watts. 2006. Experimental study of inequality and unpredictability in an artificial cultural market. *Science* (2006).

[41] Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as Treatments: Debiasing Learning and Evaluation. In *ICML*.

[42] Ashudeep Singh and Thorsten Joachims. 2018. Fairness of Exposure in Rankings. In *ACM SIGKDD*.

[43] Ashudeep Singh and Thorsten Joachims. 2019. Policy Learning for Fairness in Ranking. In *NeurIPS*.

[44] Behzad Tabibian, Vicenç Gómez, Abir De, Bernhard Schölkopf, and Manuel Gomez Rodriguez. 2019. Consequential ranking algorithms and long-term welfare. *arXiv preprint arXiv:1905.05305* (2019).

[45] Michael Taylor, John Guiver, Stephen Robertson, and Tom Minka. 2008. Softrank: optimizing non-smooth rank metrics. In *WSDM*. ACM.

[46] Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018. The spread of true and false news online. *Science* (2018).

[47] Xuanhui Wang, Nadav Golbandi, Michael Bendersky, Donald Metzler, and Marc Najork. 2018. Position bias estimation for unbiased learning to rank in personal search. In *WSDM*. ACM.

[48] Himank Yadav, Zhengxiao Du, and Thorsten Joachims. 2019. Fair Learning-to-Rank from Implicit Feedback. arXiv:cs.LG/1911.08054

[49] Hongzhi Yin, Bin Cui, Jing Li, Junjie Yao, and Chen Chen. 2012. Challenging the long tail recommendation. *VLDB* (2012).

[50] Masrour Zoghi, Tomas Tunys, Mohammad Ghavamzadeh, Branislav Kveton, Csaba Szepesvari, and Zheng Wen. 2017. Online learning to rank in stochastic click models. In *ICML*.

A LINEAR PROGRAMMING BASELINE

Here, we present a version of the fairness constraint defined in Singh and Joachims [42] that explicitly computes an optimal ranking to present in each time step τ by solving a linear program (LP). In particular, we formulate an LP that explicitly maximizes the estimated DCG of the ranking $\boldsymbol{\sigma}_\tau$ while minimizing the estimated cumulative fairness disparity D^I_τ formulated in Equation (11). This is used as a baseline to compare the P-Controller with.

To avoid an expensive search over the exponentially-sized space of rankings as in [14], we exploit an alternative representation [42] as a doubly-stochastic matrix $\mathbb{P}$ that is sufficient for representing $\boldsymbol{\sigma}_t$. In this matrix, the entry $\mathbb{P}_{y,j}$ denotes the probability of placing item y at position j. Both DCG as well as $Imp_\tau(G_i)$ are linear functions of $\mathbb{P}$, which means that the optimum can be computed as the following linear program.

$$\begin{aligned}
\mathbb{P}^* = \underset{\mathbb{P},\,\xi\geq 0}{\operatorname{argmax}} \; & \underbrace{\sum_{y} \hat{R}(y|\boldsymbol{x}) \sum_{j=1}^{n} \frac{\mathbb{P}_{y,j}}{\log(1+j)}}_{\text{Utility}} - \lambda \sum_{i,j} \xi_{ij} \\
\text{s.t. } \; \forall y, j : & \sum_{i=1}^{n} \mathbb{P}_{y,i} = 1, \quad \sum_{y'} \mathbb{P}_{y',j} = 1, \quad 0 \leq \mathbb{P}_{y,j} \leq 1 \\
\forall\, G_i, G_j : & \left(\frac{\hat{Imp}_\tau(G_i|\mathbb{P}_\tau)}{\hat{Merit}(G_i)} - \frac{\hat{Imp}_\tau(G_j|\mathbb{P}_\tau)}{\hat{Merit}(G_j)} \right) + D_{\tau-1}(G_i, G_j) \leq \xi_{ij}
\end{aligned} \tag{21}$$

The parameter λ controls trade-off between DCG of $\boldsymbol{\sigma}_t$ and fairness. We explore this parameter empirically in Section 8.1.

It remains to define $\hat{Imp}_\tau(G_j|\mathbb{P}_\tau)$. Assuming the PBM click model with $q(j)$ denoting the examination propensity of item d at position j, the estimated probability of a click is $\hat{R}(d) \cdot q(j)$. So we can estimate the impact on the items in group G for the rankings defined by $\mathbb{P}$ as

$$\hat{Imp}_\tau(G|\mathbb{P}) = \frac{1}{|G|} \sum_{d \in G} \hat{R}(d|\boldsymbol{x}) \left(\sum_{j=1}^{n} \mathbb{P}_{y,j}\; q(j) \right)$$

We use the `scipy.optimize.linprog` LP solver to solve for the optimal $\mathbb{P}^*$, and then use a Birkhoff von-Neumann decomposition [15, 42] to sample a deterministic ranking $\boldsymbol{\sigma}_\tau$ to present to the user. This ranking is guaranteed to achieve the DCG and fairness optimized by $\mathbb{P}^*$ in expectation.

Note that the number of variables in the LP is $O(n^2 + |\mathcal{G}|^2)$, and even a polynomial-time LP solver incurs substantial computation cost when working with a large number of items in a practical dynamic ranking application.

B CONVERGENCE OF FAIRCO-CONTROLLER

In this section we will prove the convergence theorem of FairCo for exposure fairness. We conjecture that analogous proofs apply to other fairness criteria as well. To prove the main theorem, we will first set up the following lemmas.

Lemma B.1. *Under the conditions of the main theorem, for any value of λ and any $\tau > \tau_0$: if $D^E_{\tau-1}(G_i, G_j) > \frac{1}{(\tau-1)\lambda}$, then*

$$\tau D^E_\tau(G_i, G_j) \le (\tau-1) D^E_{\tau-1}(G_i, G_j).$$

Proof. From the definition of D^E_τ in Eq. (10) we know that for $\tau > \tau_0$,

$$\tau D^E_\tau(G_i, G_j) = (\tau-1) D^E_{\tau-1}(G_i, G_j) + \left(\frac{Exp_\tau(G_i)}{\widehat{Merit}(G_i)} - \frac{Exp_\tau(G_j)}{\widehat{Merit}(G_j)}\right).$$

Since $D^E_{\tau-1}(G_i, G_j) > \frac{1}{(\tau-1)\lambda}$, we know that for all items in G_j it holds that $\mathbf{err}_\tau(d) > \frac{1}{\lambda}$. Hence, FairCo adds a correction term $\lambda \mathbf{err}_\tau(d)$ to the $\hat{R}(d)$ of all $d \in G_j$ that is greater than $\lambda \frac{1}{\lambda} = 1$. Since $0 \le \hat{R}(d) \le 1$, the ranking is dominated by the correction term $\lambda \mathbf{err}_\tau(d)$. This means that all $d \in G_j$ are ranked above all $d \in G_i$. Under the feasibility condition from Eq.(18), this implies that $\left(\frac{Exp_\tau(G_i)}{\widehat{Merit}(G_i)} \le \frac{Exp_\tau(G_j)}{\widehat{Merit}(G_j)}\right)$ and thus $\tau D^E_\tau(G_i, G_j) \le (\tau - 1) D^E_{\tau-1}(G_i, G_j)$. □

Lemma B.2. *Under the conditions of the main theorem, for any value of $\lambda > 0$ there exists $\Delta \ge 0$ such that for any G_i, G_j and $\tau > \tau_0$: if $D^E_{\tau-1}(G_i, G_j) \le \frac{1}{(\tau-1)\lambda}$, then $\tau D^E_\tau(G_i, G_j) \le \frac{1}{\lambda} + \Delta$.*

Proof. Using the definition the definition of D^E_τ in Eq. (10), we know that

$$\begin{aligned}
\tau D^E_\tau(G_i, G_j) &= (\tau-1) D^E_{\tau-1}(G_i, G_j) + \left(\frac{Exp_\tau(G_i)}{\widehat{Merit}(G_i)} - \frac{Exp_\tau(G_j)}{\widehat{Merit}(G_j)}\right) \\
&\le \frac{1}{\lambda} + \left(\frac{Exp_\tau(G_i)}{\widehat{Merit}(G_i)} - \frac{Exp_\tau(G_j)}{\widehat{Merit}(G_j)}\right) \\
&\le \frac{1}{\lambda} + \Delta
\end{aligned}$$

where $\Delta = \max_\sigma \max_{\substack{G, G' \\ G \neq G'}} \left(\frac{Exp_\sigma(G)}{\widehat{Merit}(G)} - \frac{Exp_\sigma(G')}{\widehat{Merit}(G')}\right)$. Note that Δ is a constant independent of τ and refers to the ranking σ for which two groups G, G' have the maximum exposure difference (e.g. one is placed at the top of the ranking, and the other is placed at the bottom). □

Using these two lemmas, we conclude the following theorem:

Theorem B.3. *For any set of disjoint groups $\mathcal{G} = \{G_1, \dots, G_m\}$ with any fixed target merits $\widehat{Merit}(G_i) > 0$ that fulfill (18), any relevance model $\hat{R}(d|\mathbf{x}) \in [0, 1]$, any exposure model $\mathbf{p}_t(d)$ with $0 \le \mathbf{p}_t(d) \le \mathbf{p}_{\max}$, and any value $\lambda > 0$, running FairCo(Exp) from time τ_0 will always ensure that the overall disparity $\overline{D}^E_\tau$ with respect to the target merits converges to zero at a rate of $O\left(\frac{1}{\tau}\right)$, no matter how unfair the exposures $\frac{1}{\tau_0}\sum_{t=1}^{\tau_0} Exp_t(G_j)$ up to τ_0 have been.*

Proof. To prove that $\overline{D}^E_\tau$ converges to zero at a rate of $O\left(\frac{1}{\tau}\right)$, we will show that for all $\tau \ge \tau_0$, the following holds:

$$\overline{D}^E_\tau \le \frac{1}{\tau} \frac{2}{m(m-1)} \sum_{i=1}^{m} \sum_{j=i+1}^{m} \max\left(\tau_0 \left|D^E_{\tau_0}(G_i, G_j)\right|, \frac{1}{\lambda} + \Delta\right)$$

The two terms in the max provide an upper bound on the disparity at time τ for any G_i and G_j. To show this, we prove by induction that $\tau D^E_\tau(G_i, G_j) \le \max\left(\tau_0 \left|D^E_{\tau_0}(G_i, G_j)\right|, \frac{1}{\lambda} + \Delta\right)$ for all $\tau \ge \tau_0$. At the start of the induction at $\tau = \tau_0$, the max directly upper bounds $\tau_0 D^E_{\tau_0}(G_i, G_j)$. In the induction step from $\tau - 1$ to τ, if $(\tau - 1) D^E_{\tau-1}(G_i, G_j) > \frac{1}{\lambda}$, then Lemma B.1 implies that $\tau D^E_\tau(G_i, G_j) \le (\tau - 1) D^E_{\tau-1}(G_i, G_j) \le \max\left(\tau_0 \left|D^E_{\tau_0}(G_i, G_j)\right|, \frac{1}{\lambda} + \Delta\right)$.
If $(\tau-1) D^E_{\tau-1}(G_i, G_j) \le \frac{1}{\lambda}$, then Lemma B.2 implies that $\tau D^E_\tau(G_i, G_j) \le \frac{1}{\lambda} + \Delta \le \max\left(\tau_0 \left|D^E_{\tau_0}(G_i, G_j)\right|, \frac{1}{\lambda} + \Delta\right)$ as well. This completes the induction, and we conclude that

$$D^E_\tau(G_i, G_j) \le \frac{1}{\tau} \max\left(\tau_0 |D^E_{\tau_0}(G_i, G_j)|, \frac{1}{\lambda} + \Delta\right).$$

Putting everything together, we get

$$\begin{aligned}
\overline{D}^E_\tau &= \frac{2}{m(m-1)} \sum_{i=0}^{m} \sum_{j=i+1}^{m} \left|D^E_\tau(G_i, G_j)\right| \\
&\le \frac{2}{m(m-1)} \sum_{i=0}^{m} \sum_{j=i+1}^{m} \left|\frac{1}{\tau} \max\left(\tau_0 |D^E_{\tau_0}(G_i, G_j)|, \frac{1}{\lambda} + \Delta\right)\right| \\
&\le \frac{1}{\tau} \frac{2}{m(m-1)} \sum_{i=0}^{m} \sum_{j=i+1}^{m} \max\left(\tau_0 \left|D^E_{\tau_0}(G_i, G_j)\right|, \frac{1}{\lambda} + \Delta\right)
\end{aligned}$$

(since $\lambda, \Delta, \tau > 0$)

□