Fair Contextual Multi-Armed Bandits: Theory and Experiments

Yifang Chen *, **Alex Cuellar, Haipeng Luo, Jignesh Modi, Heramb Nemlekar, Stefanos Nikolaidis**
Department of Computer Science, University of Southern California
Los Angeles, CA 90089, USA.
{yifang, haipengl, jigneshm, nemlekar, nikolaid}@usc.edu, alexcuel@mit.edu

Abstract

When an AI system interacts with multiple users, it frequently needs to make allocation decisions. For instance, a virtual agent decides whom to pay attention to in a group, or a factory robot selects a worker to deliver a part. Demonstrating *fairness* in decision making is essential for such systems to be broadly accepted. We introduce a Multi-Armed Bandit algorithm with fairness constraints, where fairness is defined as a minimum rate at which a task or resource is assigned to a user. The proposed algorithm uses *contextual* information about the users and the task and makes no assumptions on how the losses capturing the performance of different users are generated. We provide theoretical guarantees of performance and empirical results from simulation and an online user study. The results highlight the benefit of accounting for contexts in fair decision making, especially when users perform better at some contexts and worse at others.

1 INTRODUCTION

We focus on the problem of an AI system assigning tasks or distributing resources to multiple humans, one at a time, while maximizing a given performance metric. For instance, a virtual agent decides whom to pay attention to in a group setting, or a factory robot selects a worker to deliver a part.

If there is clearly a user who outperforms everyone else, the solution to this optimization problem would result in the agent constantly selecting that user. This approach, however, fails to account that this may be perceived as unfair by others, affecting their acceptance of the system.

*all authors contributed equally.

Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI), PMLR volume 124, 2020.

How can we integrate *fairness* in the agent's decisions? The aim of our work is to address this question. Recent works [16, 10, 20] have proposed multi-armed bandit algorithms for *fair* task allocation, where fairness is defined as a constraint on the minimum rate of arm selection. A user study on an online Tetris game, where the computer (player) selects users (arms) based on their score, has shown that users' trust is significantly improved when a fairness constraint is satisfied [10].

These works, however, have assumed that the performance of each user, observed in the form of a loss vector by the agent, follows a fixed distribution that is specific to that particular user. It thus fails to account that people may have different task-related skills. For instance, when making a pin, one worker may be specialized in cutting the wire, while another worker in measuring it. It also fails to account for cases where we cannot make statistical assumptions about the generation of losses, for instance in an adversarial domain.

We generalize this work by proposing a fair multi-armed bandit algorithm that accounts for different *contexts* in task allocation. The algorithm also does not make any assumption on how the loss vector is generated, allowing for applications in non-stationary and even adversarial settings.

We provide theoretical guarantees on performance, as well as empirical results from simulations and a proof-of-concept online user study, where an algorithm assigns knowledge-based questions to participants from different cultural backgrounds. The results show the benefit of the proposed algorithm when allocating tasks fairly to different users, especially when they are better in some contexts and worse in others.

2 PROBLEM DEFINITION

We study the online learning problem of contextual bandits (CB) with fairness constraints.

We assume M possible contexts and K available actions (arms), and use the notation $[M]$ and $[K]$ to denote the set $\{1, \ldots, M\}$ and $\{1, \ldots, K\}$. For each time step $t = 1, ..., T$:

1. The environment first decides the context $j_t \in [M]$ and the loss vector $l_t \in [0, 1]^K$.
2. The learner observes the context $j_t \in [M]$ and selects the action $i_t \in [K]$.
3. The learner suffers the loss $l_t(i_t)$.

We assume that the contexts $j_1, \ldots, j_T$ are i.i.d. samples of a fixed distribution $q \in \Delta_M$ which is known to the learner (see supplemental material for extension to the case when q is unknown). However, we make no assumption on how the loss vectors $l_1, \ldots, l_T$ are generated, and in general l_t could depend on the entire history before round t, which is a key difference compared to previous work [10].

Let Δ_K be the set of distributions over K arms. Given the history up to the beginning of round t and that context j_t is j, we let $p_t^j \in \Delta_K$ be the conditional distributions of the player's selected arm i_t, for $j = 1, \ldots, M$. We require the following *fairness* constraint parameterized by $v \in (0, 1/K)$:

$$\sum_{j=1}^{M} q(j) p_t^j(i) \geq v, \ \forall t, i \tag{1}$$

that is, the marginal probability of each arm being pulled is at least v for each time.

For notational convenience, we denote a collection of M distributions over arms by $P = (p^1, ..., p^M)$ and the feasible set of these collections in terms of the above constraint by:

$$\Omega = \left\{ P = (p^1, ..., p^M) \left| \begin{array}{c} p^1, ..., p^M \in \Delta_K \text{ and} \\ \sum_{j=1}^{M} q(j) p^j(i) \geq v, \forall i \in [K] \end{array} \right. \right\} \tag{2}$$

which is clearly a convex set and is non-empty since the uniform distribution (for all contexts) is always in the set.

The learner's goal is to minimize her regret, defined as the difference between her total loss and the loss of the best fixed distribution satisfying the fairness constraint:

$$\text{Reg} = \max_{P_* \in \Omega} \mathbb{E} \left[\sum_{t=1}^{T} \left\langle p_t^{j_t} - p_*^{j_t}, l_t \right\rangle \right]$$

Achieving sublinear regret $\text{Reg} = o(T)$ thus implies that in the long run the average performance of the learner is arbitrarily close to the best fixed distribution in hindsight.

3 BACKGROUND

Adversarial Bandits. In the case when $M = 1$ and $v = 0$ (that is, only one context and no fairness constraint), our problem is exactly the adversarial version of the classic Multi-armed Bandits (MAB) problem, first proposed in [4] and extensively studied since then. It is well-known that the minimax optimal regret is of order $O(\sqrt{TK})$. The most common algorithm with optimal regret is Exp3 [4], which can be regarded as a special case of the Follow-the-Regularized-Leader (FTRL) algorithm when we choose the regularizer to be the negative entropy.

Contextual Bandits (without fairness). When there are multiple contexts but no fairness constraint, with our regret definition there is no connection between the contexts, and the optimal algorithm is to treat each context separately and to run an individual instance of a standard MAB algorithm (such as Exp3) for each context (see Section 4 of [6]).

We assume finite number of contexts and are interested in the case when M is small. There is a different line of research where M could potentially be infinite, in which case a different measure of regret is studied or additional assumptions are made. For example, in [4, 3], the learner is given a fixed set of mappings from contexts to actions, and regret is defined in terms of the difference between the learner's total loss and the loss of the best mapping from the given set. Other works make assumptions on how the losses are connected with the context. Among those, the linear assumption is the most common one, resulting in the so-called contextual linear bandit problem (e.g. [17, 9, 2]). Another common assumption is imposing some Lipschitz conditions [7, 22].

Fair Bandits. Joseph et al. [13, 14] study fairness for bandits and draw inspiration from the idea of fair treatment suggested by Dwork et al. [11] which states that "similar individuals should be treated similarly." The definition of fairness there is quite different from ours, in that a worse arm should not be picked compared to a better arm, despite the uncertainty on payoffs. The authors provide a provably fair algorithm for the linear contextual bandit problem. Liu et al. [18] build upon this work to achieve smooth fairness, which requires arms with similar distributions to be selected with similar probabilities. They further define calibrated fairness, where an arm is selected with a probability equal to the probability of its loss being the lowest. These definitions are quite different from our notion of fairness which is a constraint on the minimum rate at which each arm is selected.

Most relevant to ours is the work by Claure et al. [10], where fairness is defined as a minimum rate on the selection of each arm, satisfied strictly throughout the task. Similarly, Li et al. [16] define fairness as the minimum rate satisfied in expectation at the end of the task. Very recent work by Patil et al. [20] further extends this definition by denoting an unfairness tolerance allowed in the system. The aformentioned works focus on a stochastic MAB setting, where the losses are independent and identically distributed. Instead, we propose an algorithm for the contextual MAB setting and we showcase the benefit of accounting for contexts in an online user study, where the system estimates the performance of players of different backgrounds in knowledge-based questions.

4 ALGORITHM

As mentioned earlier, without the fairness constraint, there is no connection among the contexts and the optimal algorithm is just to run M instances of any standard MAB algorithm separately for each possible context. For example, classic FTRL algorithm would compute for each context $j \in [M]$:

$$p_t^j = \arg\min_{p \in \Delta_K} \sum_{s: j_s = j} \left\langle p, \hat{l}_s \right\rangle + \frac{1}{\eta} \sum_{i=1}^{K} \psi(p(i)) \quad (3)$$

at the beginning of round t, where $\psi : [0,1] \to \mathbb{R}$ is some regularizer, $\eta > 0$ is some learning rate, and $\hat{l}$ is the standard unbiased importance-weighted estimator with:

$$\hat{l}_s(i) = \frac{l_s(i)}{p_s^{j_s}(i)} \mathbf{1}\{i_s = i\}, \ \forall i \in [K]$$

Upon observing the actual context j_t for round t, the algorithm then samples i_t from $p_t^{j_t}$. Standard results [6] show that the j-th instance of FTRL suffers regret $O(\sqrt{|\{t : j_t = j\}|K})$, and thus the total regret is $\sum_{j=1}^{M} O(\sqrt{|\{t : j_t = j\}|K}) = O(\sqrt{TMK})$ via the Cauchy-Schwarz inequality.

With the fairness constraint, however, we can no longer treat each context separately. A natural idea is to optimize jointly over the feasible set Ω defined in Eq. (2), that is, to find $P_t = (p_t^1, \cdots, p_t^M)$ at round t such that:

$$P_t = \arg\min_{P \in \Omega} \sum_{s=1}^{t-1} \left\langle p^{j_s}, \hat{l}_s \right\rangle + \frac{1}{\eta} \sum_{j=1}^{M} \sum_{i=1}^{K} \psi(p^j(i))$$

It is clear that when $v = 0$ (that is, no fairness constraint), the feasible set Ω simply becomes $\Delta_K \times \cdots \times \Delta_K$ and the joint optimization above decomposes over j so that the algorithm degenerates to that described in Eq. (3).

Algorithm 1 Fair CB with Known Context Distribution

Input: learning rate $\eta > 0$, fairness constraint parameter v
Define: $\Psi(P) = \sum_{j=1}^{M} \sum_{i=1}^{K} \psi(p^j(i))$ where $\psi(p) = p \ln p$
for $t = 1, \ldots, T$ **do**
  Compute $P_t = \arg\min_{P \in \Omega} \sum_{s=1}^{t-1} \left\langle p^{j_s}, \hat{l}_s \right\rangle + \frac{1}{\eta}\Psi(P)$
  Observe j_t and play $i_t \sim p_t^{j_t}$
  Construct loss estimator $\hat{l}_t(i) = \frac{l_t(i)}{p_t^{j_t}(i)} \mathbf{1}\{i_t = i\}$, $\forall i \in [K]$
end for

When $v \neq 0$, the algorithm satisfies the fairness constraint automatically and can be seen as an instance of FTRL over a more complicated decision set Ω.

We deploy the standard entropy regularizer $\psi(p) = p \ln p$, used in the classic Exp3 algorithm [4] for MAB. See Algorithm 1 for the complete pseudocode. We remark that even though unlike Exp3, there is no closed form for computing P_t, one can apply any standard convex optimization toolbox to find P_t when implementing the algorithm.

We provide the following regret guarantee of our algorithm, which is essentially the same as the aforementioned bound for $v = 0$. We note that the regret bound is tight with respect to all parameters, since it matches the known lower bound even in the case when there is no fairness constraint. On the other hand, also note that when v increases, the feasible set Ω becomes smaller and so does its range D, meaning that our algorithm suffers smaller regret when the fairness constraint is more stringent. The proof is included in the supplemental material.

Theorem 1. *With learning rate* $\eta = \sqrt{\frac{D}{TK}}$, *where* $D = \max_{P \in \Omega} \Psi(P) - \min_{P \in \Omega} \Psi(P) \leq M \ln K$, *Algorithm 1 achieves Reg* $= O\left(\sqrt{TKD}\right) = O\left(\sqrt{TMK \ln K}\right)$.

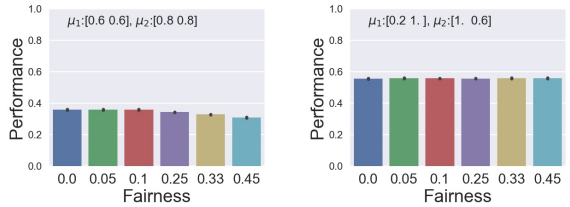
5 EXPERIMENTS

This section illustrates different behaviors of the Fair CB algorithm, highlighting the interplay between choice of loss distributions, fairness and context.

For each experiment we define the empirical performance of the algorithm in each experiment trial as one minus the average loss.

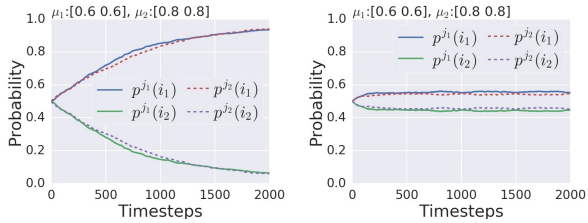
$$\text{Performance} = 1 - \frac{\sum_{t=1}^{T} l_t(i_t)}{T}.$$

In all experiments we set the learning rate as: $\eta =$



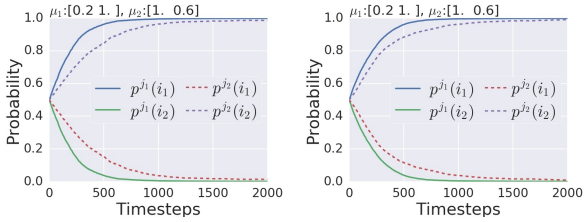
(a) i_1 better at both contexts (b) i_1 better only at j_1

Figure 1: Performance of algorithm for different levels of fairness for $T = 2000$. The performance is averaged over 2000 timesteps and 100 simulations.



(a) $v = 0$ (b) $v = 0.45$

Figure 2: Probabilities of pulling an arm over time averaged over 100 simulations when one arm is better in both contexts.



(a) $v = 0$ (b) $v = 0.45$

Figure 3: Probabilities of pulling an arm over time averaged over 100 simulations when one arm is better in one context and worse in another.

$\sqrt{M \ln K/TK}$, following the theoretical result of section 4.

We are motivated by settings where a system assigns resources to human users (arms) based on whether they succeed in a task or they exhibit a desired behavior. In Sections 5.1 and 5.2 we thus focus on the case where the loss induced by an arm i under context j follows a Bernoulli distribution parametrized by $\mu_{i,j} \in [0, 1]$, so that $l_t(i)$ is 1 with probability $\mu_{i,j}$ and 0 with probability $1 - \mu_{i,j}$, when the context is j. To showcase the advantage of our adversarial algorithm, in Section 5.3 we also consider time-varying Bernoulli distributions. The fairness level v specifies the minimum rate that an arm is selected as defined in Eq. (1).

5.1 FAIRNESS AFFECTS PERFORMANCE

With the presence of contexts, having a fairness constraint does not always lead to worse performance. For instance, if for each arm, the probability of seeing the contexts in which this arm is the best is larger than v, then the fairness constraint can be satisfied trivially by picking the best arm for each context and the performance is also the best. However, in the case where the fairness constraint forces the algorithm to select suboptimal arms, larger value of v unavoidably leads to worse performance. Below we demonstrate this phenomenon empirically with our Fair CB algorithm.
We start with the simplest case of two arms (i_1 and i_2) and two contexts (j_1 and j_2), and then we show how our insights generalize to more contexts and arms.

Even distribution of both contexts: We first let the contexts be distributed evenly, that is, $q(j_1) = q(j_2) = 0.5$.

If each arm is better than the other in one of the contexts, we expect that fairness does not affect the performance of an optimal algorithm, since the probability of the context occuring – and thus that arm being selected – is $q(j) = 0.5$ which is always greater than a fairness constraint $v \in \left(0, \frac{1}{K}\right)$. On the other hand, if one arm is better than the other in both contexts, we expect the algorithm to enforce the fairness constraint and choose the weakest arm with the minimum rate in at least one of the contexts.

Arm 1 is better in both contexts. For instance, we let $\mu_1 = (\mu_{i_1,j_1}, \mu_{i_1,j_2}) = (0.6, 0.6)$ be the expected values of the loss distributions for contexts 1 and 2 for arm 1, and $\mu_2 = (\mu_{i_2,j_1}, \mu_{i_2,j_2}) = (0.8, 0.8)$ for arm 2. We run the algorithm in simulation for varying levels of fairness. We expect that increasing fairness results in selecting the suboptimal arm (i_2) with increasing frequency, which subsequently increases the total loss.

Fig. 1(a) shows the performance of our algorithm for six different values of v over $T = 2000$ rounds (and averaged over 100 simulations). As expected, the performance degrades as v gets larger. Note that performance was similar across the first three fairness levels. We attribute this to the inherent exploration of the FTRL algorithm from the regularization term and the relatively small difference between the expected losses μ_1 and μ_2 of the two players. A linear regression established that the fairness level significantly predicted performance, with $F(1, 598) = 1168.5, p < .0001$ and fairness accounted for 66.1% of the explained variability in performance. The regression equation was: predicted performance $= 0.37 - 0.11v$.

Fig.2 shows the assigned probabilities by the algorithm for every timestep, averaged over 100 simulations. Since i_1 is better than i_2 in both contexts, it eventually gets selected with probability close to 1 in both contexts when

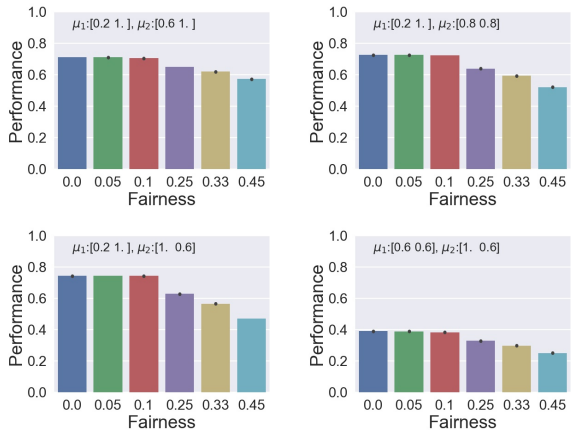


Figure 4: Performance for 2-arm 2-context with $q(j_1) = 0.9, q(j_2) = 0.1, T = 10000$, averaged over 100 simulations.

fairness $v = 0$ and with probability 0.55 when fairness $v = 0.45$.

There is no arm that is better in both contexts. In this case fairness level does not affect the performance of our algorithm, as shown in Fig.1(b), where $\mu_1 = (0.2, 1.0), \mu_2 = (1.0, 0.6)$.

Fig.3 shows the assigned probabilities over time. Regardless of the fairness parameter, since i_1 is better than i_2 in j_1 but worse in j_2, i_1 will be selected with probability close to 1 for j_1 and i_2 with probability close to 1 for j_2. Since j_1 and j_2 are distributed with probability 0.5, the fairness constraint is naturally satisfied.

Uneven distribution of contexts: We then examine the general case where contexts are distributed with different probabilities. We expect that increasing fairness will result in worse performance when one arm is better than the other arm in both contexts, or when one arm i_1 is better than the other arm i_2 in only one context j_1 and $v > q(j_2)$. We let $q(j_1) = 0.9, q(j_2) = 0.1$ be the distribution of the two contexts.

The case for one arm being better in both contexts follows the same reasoning as before. On the other hand, if one arm i_1 is better than the other arm in one of the contexts j_1 with probability $q(j_1)$, we expect increasing fairness to reduce performance for $v > q(j_2) = 0.1$.

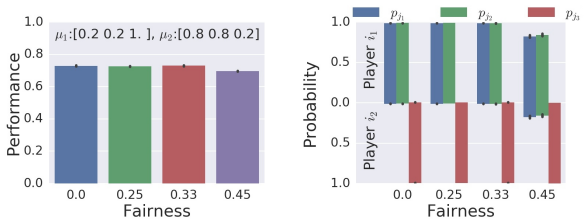
Indeed, for different combinations of $\mu_{i_1,j_1}, \mu_{i_1,j_2} \in \{0.2, 0.6, 1\}$, $\mu_{i_2,j_1}, \mu_{i_2,j_2} \in \{0.6, 0.8, 1\}$, a multiple regression model statistically significantly predicted performance, with $F(2, 2397) = 7294, p < .0001$, adj. $R^2 = 0.86$ and fairness being a significant predictor ($p < 0.001$). Fig. 4 shows the performance for different configurations. We see that indeed fairness starts decreasing the performance once $v > 0.1$.

Multiple contexts and arms: We evaluate the performance of the Fair CB algorithm when there are more than two contexts and arms for completeness.

More arms than contexts. We first show the performance of the Fair CB algorithm if we have more arms than contexts. In that case, there will be one remaining arm that is sub-optimal in all contexts. Since the fairness constraint specifies a minimum rate for all arms, increasing fairness will decrease performance, as we observed in the previous section.

This, however, will not hold if there are multiple optimal arms with identical loss distributions for a given context. We let the example where $\mu_{i_1,j_1}, \mu_{i_1,j_2} = (0.2, 1.0)$, $(\mu_{i_2,j_1}, \mu_{i_2,j_2}) = (0.6, 1.0)$, $(\mu_{i_3,j_1}, \mu_{i_3,j_2}) = (0.8, 1.0)$, and $q(j_1) = q(j_2) = 0.5$. If there is no fairness ($v = 0$), the algorithm will pick i_1 with probability 1 in context j_1, while in context j_2 all arms are selected uniformly, since they have identical losses. For $v = 0.25$, the algorithm will keep picking i_1 in j_1, but it will pick evenly i_2 and i_3 in j_2. The constraint is satisfied since each context appears with probability $q(j_1) = q(j_2) = 0.5$. However, larger value of fairness will instead result in one of the two arms that are sub-optimal in j_1 to get selected in that context as well, resulting in a decrease in performance.

More contexts than arms. We then explore the case of $K = 2$ arms and $M = 3$ contexts. Clearly, if one arm is better than the second arm in all the contexts, increasing fairness will decrease performance. We focus on the case where i_1 is better than i_2 in two of the contexts, j_1 and j_2 and worse in the third context j_3. We assume again equal probability distribution of contexts $q(j) = \frac{1}{M}, j \in [M]$.



(a) Overall performance (b) Final probabilities

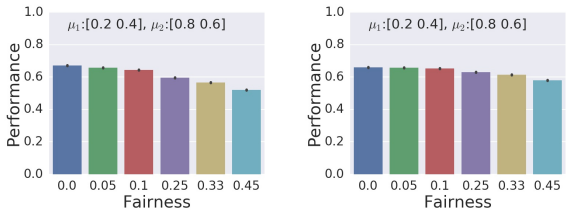
Figure 5: 2-arm 3-context problem where i_1 is better in j_1 and $j_2, T = 3000$.

We let $\mu_1 = (\mu_{i_1,j_1}, \mu_{i_1,j_2}, \mu_{i_1,j_3}) = (0.2, 0.2, 1.0)$ and $\mu_2 = (\mu_{i_2,j_1}, \mu_{i_2,j_2}, \mu_{i_2,j_3}) = (0.8, 0.8, 0.2)$. Fig.5 shows that the algorithm ends up picking i_1 with probability 1.0 in context j_1 and j_2, and i_2 with probability 1.0 in context j_3 (see Fig.5(b)). However, when $v > 0.33$, the algorithm needs to pull arm i_2 in j_1 and j_2 leading to decrease in performance. This result is supported by an one-way ANOVA ($F(1, 198) = 273.8, p < 0.0001$.

Overall, our analysis shows that, for any number of con-

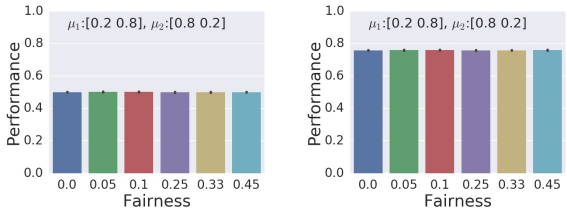
texts and arms, fairness matters if the fairness constraint enforces an arm to be pulled in a context that is not optimal, which occurs either when there is no context where the arm is optimal, or when the probability of the context(s) that the arm is optimal is smaller than the probability imposed by the fairness constraint.

5.2 IMPORTANCE OF CONTEXTS



(a) Non-contextual FTRL (b) Fair CB

Figure 6: Performance when i_1 is better in both contexts, $q(j_1) = q(j_2) = 0.5$ and $T = 2000$, averaged over 100 simulations.



(a) Non-contextual FTRL (b) Fair CB

Figure 7: Performance when i_1 is better in one of the contexts only, $q(j_1) = q(j_2) = 0.5$ and $T = 2000$, averaged over 100 simulations.

To illustrate the importance of contexts, we compare to an FTRL algorithm that ignores the context (equivalently, our algorithm with $M = 1$). We consider two arms and an even distribution among two contexts.

First, we examine the case where one arm is better than the other in both contexts: $((\mu_{i_1,j_1}, \mu_{i_1,j_2}) = (0.2, 0.4), (\mu_{i_2,j_1}, \mu_{i_2,j_2}) = (0.8, 0.6))$. Fig. 6 shows the result for increasing values of fairness. While for 0 fairness there is no noticeable difference, as fairness increases, we observe that our Fair CB performs better. A one-way ANOVA for $v = 0.45$ showed a significant effect of the choice of algorithm on performance ($F(1, 198) = 1197.43, p < 0.0001$). Despite arm i_1 being better than arm i_2 in both contexts, we see a difference in performance, since the difference between the two arms' loss is much higher for the first context than the second. The contextual algorithm recognizes this disparity and selects to impose the fairness constraint in the second context rather than in both contexts.

Fig. 7 shows another result when one player is better in one context and worse in the other $((\mu_{i_1,j_1}, \mu_{i_1,j_2}) = (0.2, 0.8), (\mu_{i_2,j_1}, \mu_{i_2,j_2}) = (0.8, 0.2))$. We observe that the contextual algorithm outperforms the baseline in all fairness levels, since it distributes the arms to different contexts while satisfying the fairness constraint.

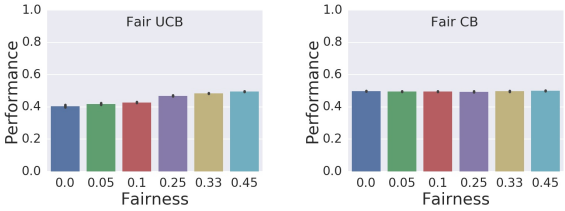


Figure 8: Performance of Fair UCB and Fair CB algorithms for 2-arm 1-context problem with adversarial losses, with $T = 1500$ averaged over 100 simulations.

5.3 ADVERSARIAL LOSSES

An advantage of the Fair CB algorithm is that it makes no assumptions on how losses are generated. This contrasts previous work on fair task allocation [16, 10, 20], which assume a fixed distribution.

To showcase this advantage, we compare our algorithm with Fair UCB, which assumes a stochastic setting and implements the standard UCB algorithm with a minimum pulling rate constraint (fairness) for each arm. While different implementations of Fair UCB were proposed independently by Claure et al. [10] and Patil et al. [20], we use the former stochastic-rate constrained UCB implementation. Since we wish to focus on the effect of adversarial losses on performance, we used only one context ($M = 1$) in both algorithms.

To simulate an adversarial setting, we generate the loss vector as follows: every time the learner incurs a loss of 0, the loss distribution switches between $(\mu_{i_1}, \mu_{i_2}) = (0.1, 0.9)$ and $(\mu_{i_1}, \mu_{i_2}) = (0.9, 0.1)$ (note that the index for j is omitted here since $M = 1$).

We evaluate the performance of our algorithm and Fair UCB for different levels of fairness. A two-way ANOVA comparing the main effects of algorithm selection (Fair UCB and Fair CB) and fairness level (v) on performance shows a significant difference for both algorithms ($F(1, 1188) = 1926.4$, $p < 0.001$, Fair UCB M = 0.45, SE = 0.0017, Fair CB M = 0.496, SE = $5.46e-4$) and fairness ($F(5, 1188) = 220.41$, $p < 0.001$). There was a significant interaction between the effects of algorithm selection and fairness ($F(5, 1188) = 211.46$, $p < 0.001$).

Fig. 8 shows the performance of the two algorithms. We observe that fairness does not affect performance for Fair CB, since the switching loss vector makes the algorithm

already quite conservative in the arm selection. On the contrary, Fair UCB has poor performance when fairness is small, while performance improves for increasing levels of fairness. This is because large fairness level makes the algorithm rely less on the UCB bound which is exploited by the adversary in this setting.

5.4 MOVIELENS DATASET

We also test the performance of our algorithm for recommending movie genres using the MovieLens 25m dataset [12]. We formulate the bandit problem as selecting a genre (arm) for a particular user (context) based on the ratings provided by the user. Fairness is enforced over genres to ensure that all genres are recommended to the users at a rate of at least v. We consider the first two users in the MovieLens dataset as the two contexts and the five genres: `Drama`, `Action`, `Comedy`, `Adventure`, and `Crime` (that are common between the users), as the arms. We see that the performance of Fair CB decreases as the fairness level is increased. For fairness values $v = \{0, 0.05, 0.1, 0.199\}$, a linear regression established that the fairness level significantly predicted performance, with $F(1, 399) = 564$ and $p < 0.0001$. The regression equation was: predicted performance $= 0.799 - 0.125v$. When compared to a non-contextual FTRL, a one-way ANOVA for $v = 0.199$ showed a significant effect of the choice of algorithm on performance ($F(1, 198) = 26130$, $p < 0.0001$), thus highlighting the importance of contexts.

6 USER STUDY

We wish to assess whether accounting for contexts when distributing the resources fairly, leads to a better performance. Results from section 5.2 show that Fair CB is particularly beneficial when the arms are better in one context and worse in another. Therefore, we design a proof-of-concept online user study, where we expect participants to perform better in different contexts.

Previous studies on contextual bandits have mostly utilized offline datasets of recommendation systems like Yahoo! Today Module [17, 24, 23] and MovieLens [5], without any fairness restrictions. A recent study on contextual bandits [19] considers fair distribution of information in tutoring systems, however it assumes that the context remains same in each round. On the contrary, in our study, the system has to assign knowledge-based questions from two different topics (contexts) to one of two users (arms) in an online quiz. The system attempts to assign questions such that the number of correct answers are maximized while ensuring that each user receives at least a minimum number of questions.

6.1 EXPERIMENTAL SETUP

Methodology: We created an online quiz where users have to identify states and famous people from either USA or India, which are the 2 contexts. We paired two users to simultaneously take the quiz by matching users indicating India as their country of origin with users indicating the United States. We did this with the expectation that users from India would be better in questions related to their country than users from USA and vice versa.

We had two quizzes, each assigned to one of the algorithms: Fair CB and a non-contextual FTRL, which is our *Baseline*. Each quiz had a fixed set of 44 questions evenly distributed between the two topics (20 questions about India, 20 about USA in alternating order). The first four questions of each quiz were equally divided among the two players for initialization. For each question, users had 10 seconds to select one out of four candidate answers.

We adopted a within-subjects design, where the same pair of users took both quizzes, one running the Fair CB algorithm and other running the Baseline algorithm. We counter-balanced the assignment of quizzes to algorithms. While we did not expect any learning effects, since the quizzes included knowledge-based questions, we had a training section where subjects answered example questions and we also counter-balanced the order of the two algorithms.

Algorithm: In this experiment we had two contexts $M = \{1, 2\}$ and two human participants $K = \{1, 2\}$. We set the fairness parameter v to 0.33. We tuned the learning rate for both algorithms to $\eta = 0.25$.

To reduce variance from sampling, we implemented the Fair CB algorithm with deterministic schedules by setting a "window" of 10 questions, 5 for each context in alternating order, and we assigned participants to questions deterministically, based on the output of each algorithm. For instance, if $p^{j_1}(i_1) = 0.6$ for context 1 and $p^{j_2}(i_1) = 1.0$ for context 2, we assigned 3 of the 5 questions of context 1 to participant 1, all 5 questions of context 2 to participant 1, and the remaining questions to participant 2.

At the end of that window the system received the loss values for each question corresponding to the context and participant, and updated the participant probabilities. Since we had a total of 44 questions, the algorithm performed 4 updates.

Hypotheses: We make the following hypothesis:

H1. *Fair CB algorithm will perform better than the Baseline algorithm.* Since we expect users to be more knowledgeable in one of the contexts and less knowl-

edgeable in the other context, we expected that Fair CB would result in better performance, compared to an algorithm that assesses users based on their performance in both contexts together. We base this on the results from the simulations in section 5.2.

H2. *Participants' subjective responses will not be worse in the Fair CB algorithm, compared to the baseline.* Since both algorithms account for fairness, we expected users' responses for the Fair CB to be at least as good as in the baseline case.

We note that we did not compare against different fairness levels, since simulations in section 5.1 show that fairness matters only when one arm is better at both contexts, which we expect to happen infrequently in this study. We refer the reader to previous studies [10] which highlight the effects of fairness on users' perceived fairness and trust in the system.

Measures: We recorded the participants' performance, the number of the questions assigned, the loss values corresponding to the participant responses, and the probabilities estimated at each time step. We additionally asked participants questions related to their perceived fairness and trust in the system, using survey questions (Table 1), where each response was measured on a seven-point Likert scale.

Procedures: We recruited participants using Amazon Mechanical Turk (AMT) and used Qualtrics to host the survey. The AMT participants were instructed that they would be paired with another person to take the quiz together and the computer would decide who gets to answer a particular question. After the quiz the participants were redirected to the survey, where they answered questions about their experience. The study was approved by the Institutional Review Board of our University.

Participants: We recruited 80 participants (40 pairs) from AMT. We removed the data for 3 pairs because they did not complete the quiz. The final dataset has 37 pairs of participants and is publicly available [1].

6.2 RESULTS

6.2.1 Performance

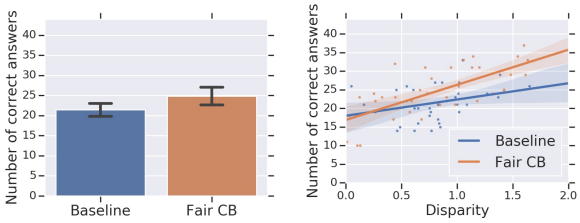


Figure 9: Performance of Fair CB compared to Baseline.

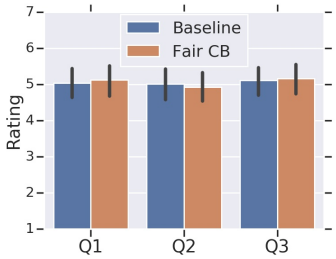


Figure 10: Responses to the subjective questions in Table 1 by each player for the Baseline and Fair CB algorithms

We measure the performance of each algorithm by the total number of questions answered correctly for each quiz. A paired t-test showed a statistical difference ($t(36) = -3.308, p = 0.002$) in the performance of the users for Baseline (M = 21.472, SE = 0.777) and Fair CB (M = 24.833, SE = 1.137) conditions. On average, the users answered 48.8% questions correctly in the Baseline and 56.44% questions correctly in the Fair CB conditions. We found no significant effect of the set of questions on performance. This result supports hypothesis H1.

A post-hoc analysis of the data shows that the difference in performance was larger when one participant was much better than the other in one of the contexts. We show this by defining the *disparity* between the participants as the average difference in participant performances for each context. Higher disparity means that one participant was much better in one of the contexts and worse in the other context:

$$\delta = |(\hat{\mu}_{i_1,j_1} - \hat{\mu}_{i_2,j_1}) - (\hat{\mu}_{i_1,j_2} - \hat{\mu}_{i_2,j_2})|$$

where $\hat{\mu}_{i_1,j_1}$ (and similarly for others) is the measured performance per question of participant i_1 in context j_1 at the end of the experiment.

A linear regression on the performance of the Baseline established that disparity (δ) did not show a significant effect, $F(1, 34) = 3.65, p = 0.0645$ and accordingly disparity accounted for only 7.04% of the explained variability. Whereas, a linear regression on the performance of Fair CB established that disparity significantly predicted its performance, $F(1, 34) = 32.9, p < 0.0001$ and the model explained 47.7% of the variability in performance. The regression equation was: predicted performance $= 16.871 + 9.448\delta$. Fig. 9 shows the positive effect of disparity on the performance of Fair CB.

6.2.2 Subjective Responses

Out of the 37 pairs of participants that completed the quiz, 27 pairs (54 participants) answered all the subjective responses. We compare the responses of participants for the subjective questions given in Table 1 across the Baseline and Fair CB algorithms (Fig. 10).

Index	Question
Q1.	How FAIR or UNFAIR was it for YOU that the computer gave you the designated number of questions?
Q2.	How FAIR or UNFAIR was it for your PARTNER that the computer gave them the designated number of questions?
Q3.	How much do you trust the computer to make a good decision about the distribution of questions?

Table 1: Survey questions answered for both algorithms after the quiz.

Example Quote
"Maximum from US based question to me while other India based"
"I feel my partner had more in section B (Baseline)."
"There seemed to be fewer questions in a row for each of us in Set B (Fair CB)."
"Part 1 (Fair CB) seemed to do a much better job of giving questions about the US to me, and questions about India to my partner."
"The first (Fair CB) was more even, in the 2nd (Baseline) the other player got a lot more questions"

Table 2: Participants response to the question "Did you notice a difference in how each part distributed questions?"

To test our hypothesis that the perceived fairness of the Fair CB algorithm is not worse than the Baseline,[1] a one-tailed paired t-test for a non-inferiority margin $\Delta = 0.5$ and a level of statistical significance $\alpha = 0.025$ showed that participants perceived the fairness of the Fair CB algorithm not worse than the Baseline for all questions ($p < 0.0001$).

We also asked participants to describe any difference they noticed in the way the questions were distributed between the two quizzes corresponding to the two algorithms. Users that did not have a clear disparity in their performance in the two contexts did not see a difference in the behaviour of the two algorithms. Users with greater disparity noticed a difference between the two algorithms, with some users even recognizing that Fair CB did better at assigning them questions that were related to their country. Table 2 shows some example responses.

7 DISCUSSION

We view our findings as valuable considerations regarding AI systems that make fair allocation decisions to multiple users. Theoretically, we show how the classic FTRL framework can be naturally generalized to ensure fairness and we rigorously analyze the performance of our proposed algorithm in terms of regret guarantees. In the supplemental material, we provide an algorithm for the case of unknown context distribution and we analyze its performance in terms of fairness violation.

Empirically, our first finding is that increasing fairness results in worse performance, when there is one user who is outperformed in all contexts. On the other hand, if there exists a context where a user outperforms all others, whether fairness will affect performance depends on the distribution of contexts. If that context appears frequently enough for the desired fairness constraint to be satisfied, performance will not be affected.

We also found that having a fair algorithm with no statistical assumptions about the process generating the losses is particularly beneficial in adversarial domains. Interestingly, increasing fairness in our adversarial setting was beneficial to the Fair UCB algorithm, since fairness reduced the reliance on the optimistic bounds that was exploited by the adversary.

Finally, the benefit of the context-based algorithm depends on the disparity between users, that is how much they differ in their performance on each context. In our user study, Fair CB performed best for pairs of participants where each participant was better on one context and worse on another.

Future Directions. We are excited to further investigate how our findings can generalize beyond online game settings, in domains where multiple users interact with a physically embodied robot : for instance, a robot receptionist greeting customers, an assistive robot in a stroke care facility helping patients eating a meal, or a factory robot delivering parts to workers. We are also excited about applications of this work in banking and advertising, where arms are partitioned into protected groups and a decision maker needs to allocate resources across these groups [21].

Conclusion. Overall, we are excited to have brought about a better understanding of the interplay between contexts, fairness and performance in task allocation settings, in addition to the theoretical analysis [8]. Designing AI systems that ensure and demonstrate fairness when interacting with people is critical to their acceptance, and deriving theoretical and experimental foundations for these systems is yet an under-served aspect in Human-AI Interaction.

[1] We define "not worse than" using the concept of "non-inferiority" [15].

References

[1] User study dataset. https://github.com/icaros-usc/fairCB-dataset.

[2] Y. Abbasi-Yadkori, D. Pál, and C. Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, pages 2312–2320, 2011.

[3] A. Agarwal, D. Hsu, S. Kale, J. Langford, L. Li, and R. Schapire. Taming the monster: A fast and simple algorithm for contextual bandits. In *International Conference on Machine Learning*, pages 1638–1646, 2014.

[4] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002.

[5] A. Balakrishnan, D. Bouneffouf, N. Mattei, and F. Rossi. Incorporating behavioral constraints in online AI systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3–11, 2019.

[6] S. Bubeck, N. Cesa-Bianchi, et al. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1):1–122, 2012.

[7] S. Bubeck, G. Stoltz, and J. Y. Yu. Lipschitz bandits without the Lipschitz constant. In *International Conference on Algorithmic Learning Theory*, pages 144–158. Springer, 2011.

[8] Y. Chen, M. Cuellar, Alex, H. Luo, J. Modi, H. Nemlekar, and S. Nikolaidis. The fair contextual multi-armed bandit (extended abstract). In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, 2020.

[9] W. Chu, L. Li, L. Reyzin, and R. Schapire. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 208–214, 2011.

[10] H. Claure, Y. Chen, J. Modi, M. Jung, and S. Nikolaidis. Multi-armed bandits with fairness constraints for distributing resources to human teammates. In *Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*, pages 299–308, 2020.

[11] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, pages 214–226. ACM, 2012.

[12] F. M. Harper and J. A. Konstan. The movielens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems (TIIS)*, 5(4):1–19, 2015.

[13] M. Joseph, M. Kearns, J. Morgenstern, S. Neel, and A. Roth. Fair algorithms for infinite and contextual bandits. *arXiv preprint arXiv:1610.09559*, 2016.

[14] M. Joseph, M. Kearns, J. H. Morgenstern, and A. Roth. Fairness in learning: Classic and contextual bandits. In *Advances in Neural Information Processing Systems*, pages 325–333, 2016.

[15] E. Lesaffre. Superiority, equivalence, and non-inferiority trials. *Bulletin of the NYU hospital for joint diseases*, 66(2), 2008.

[16] F. Li, J. Liu, and B. Ji. Combinatorial sleeping bandits with fairness constraints. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pages 1702–1710. IEEE, 2019.

[17] L. Li, W. Chu, J. Langford, and R. E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th International Conference on World Wide Web*, pages 661–670. ACM, 2010.

[18] Y. Liu, G. Radanovic, C. Dimitrakakis, D. Mandal, and D. C. Parkes. Calibrated fairness in bandits. *arXiv preprint arXiv:1707.01875*, 2017.

[19] B. Metevier, S. Giguere, S. Brockman, A. Kobren, Y. Brun, E. Brunskill, and P. S. Thomas. Offline contextual bandits with high probability fairness guarantees. In *Advances in Neural Information Processing Systems*, pages 14893–14904, 2019.

[20] V. Patil, G. Ghalme, V. Nair, and Y. Narahari. Achieving fairness in the stochastic multi-armed bandit problem. *arXiv preprint arXiv:1907.10516*, 2019.

[21] C. Schumann, Z. Lang, N. Mattei, and J. P. Dickerson. Group fairness in bandit arm selection. *arXiv preprint arXiv:1912.03802*, 2019.

[22] A. Slivkins. Contextual bandits with similarity information. *The Journal of Machine Learning Research*, 15(1):2533–2568, 2014.

[23] L. Tang, Y. Jiang, L. Li, and T. Li. Ensemble contextual bandits for personalized recommendation. In *Proceedings of the 8th ACM Conference on Recommender Systems*, pages 73–80. ACM, 2014.

[24] Q. Wu, H. Wang, Q. Gu, and H. Wang. Contextual bandits in a collaborative environment. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 529–538. ACM, 2016.