Combinatorial Sleeping Bandits with Contextual Information and Fairness Constraints

***Abstract*—Multi-armed bandit (MAB) is a classic model to make online decisions with the objective of maximizing accumulative rewards, and has been widely applied to practical problems, including ad placement, source routing and computer game playing. Importantly, the MAB model has been extended to address several critical design challenges and considerations, among which the following three are the foci of many recent studies: combinatorial decisions, intermittent arms (i.e., sleeping bandits), and fairness constraints for arm selection. In this paper, we contribute to the existing research by incorporating contextual information — the widely available information that helps make better decisions tailored to the specific condition at runtime — into the extended MAB model and propose a novel online learning algorithm to the under-explored contextual combinatorial sleeping bandit with fairness constraints. Concretely, by extending the LinUCB and Lyapunov optimization framework, our algorithm selects context-specific arms based on the set of available arms at runtime, while satisfying the long-term constraints on the number of times each arm is selected (i.e., fairness constraints). We provide a theoretical performance guarantee on our algorithm and show that, compared to the optimal oracle, the achieved time-average regret is upper bounded by $\frac{N}{2V} + c_1\frac{N}{T} + (c_2 + c_3\log T + c_4\sqrt{\log T})\sqrt{\frac{mN}{T}}$, where N is the total number of arms, m is maximum number of arms that can be played simultaneously in each round, T is the number of total rounds played, and the other parameters are appropriately chosen constants. We also run experiments on both real-world and synthetic datasets to further validate the effectiveness of our algorithm, demonstrating a negligible time-average regret while satisfying the fairness constraints.**

***Index Terms*—Contextual Multi-Armed Bandit, Combinatorial Bandits, Sleeping Bandits, Fairness, Lyapunov Optimization**

I. Introduction

The multi-armed bandit (MAB) problem is a process of sequential decision making with the tradeoff between exploration and exploitation. In each round t, an action is chosen from a pool of candidates called arms, each of which corresponds to an unknown *a priori* reward. The actual reward is not known until the arm is chosen and played. The goal is to maximize the accumulated rewards on a time horizon T or T rounds. With more rounds being played, some arms' rewards become better known to us. The tradeoff is a balance between sticking to the arm with currently known maximal reward (exploitation), or exploring new arms which might give higher rewards in the long run (exploration).

MAB models have been widely applied to practical problems, including as ad placement, source routing, computer game-playing, among others [1]. In a basic MAB setting, the random reward of each arm, often normalized to $[0, 1]$, is subject to independent and identical distributions. Besides the instant reward in each round, the mean reward of each arm is also unknown *a priori* and needs to be learnt over time.

The basic MAB model has been extended in various directions, among which the following three have been the foci of many recent studies. The first one is the combinatorial bandits. While the early work on MAB mainly focuses on the scenario where only a single arm is chosen in each round, pulling a combination of arms (called combinatorial bandits) is also common in practice. Take the online movie platform for example, where multiple movies (each viewed as an arm) can be recommended to a user each time. Second, a practical situation is often that some arms might be unavailable in certain rounds (e.g., due to product unavailability), which is called "sleeping bandits". In this case, those unavailable arms cannot be considered as candidate arms. Third, in practice, specific arms may need to be guaranteed with a minimum number of selection times, which ensures the important notion of *fairness* among arm selection. For instance, due to business contracts, each advertisement (i.e., an arm) needs to be displayed to users for at least certain number of times. Studies that address all these three factors are referred to as combinatorial sleeping bandits with fairness constraints [2].

Although the recent progress in combinatorial sleeping bandits with fairness constraints is appealing, a key limitation of the existing studies is that they are context-*oblivious*: arms are selected while disregarding any information about states of the environment (i.e., context), which can significantly degrade the learning performance of MAB. For example, in an advertisement placement application, regardless of user preferences or advertisement features (contexts), the same advertisement will be displayed, which clearly does not result in the maximum click-through rate. Contextual bandit learning addresses this limitation by making use of the rich side information of the environment (i.e., context). In contextual bandits, there exists a set of arm selection policies, each of which maps a context to an arm [3]. Contextual bandits have been applied to many real-world applications, such as personalized news recommendation application where each news article (arm) is matched to a certain set of users depending on the user preferences and news content (contexts) in order to maximize the click-through rate.

In this paper, we contribute to the literature by incorporating contextual information into combinatorial sleeping bandits with fairness constraints. We first formulate the contextualized version of combinatorial sleeping bandits with fairness constraints, in which side information (i.e., context) is provided to the learner prior to arm selection and arms can only be selected

out of an intermittently available set at each round. As in the literature [2], fairness is defined as the minimum number of times an arm needs to be chosen over the entire time horizon. Naturally, the fairness constraint couples all the arm selections over time, while only current contextual information and set of currently available arms are known at each round. Thus, the lack of complete offline information makes the problem of arm selection subject to fairness constraints very challenging, let alone the unknown reward functions corresponding to each arm and context that need to be learnt over time.

We propose a novel provably-efficient algorithm, called Contexual bandit Learning with Fairness Constraints (CLFC), which learns online to select the optimal arm set given each context while meeting the long-term fairness constraints. CLFC builds on the general yet computationally efficient contextual bandit algorithm LinUCB (linear upper confidence bound) [4] as well as the classic Lyapunov optimization framework [5]. Concretely, at each round, CLFC selects arms based on not only the upper confidence bound associated with each given context, but also the virtual fairness *deficit* queues which indicate how far each arm selection up to the current round are away from the fairness constraints. A longer queue means that the corresponding arm has not been selected adequately and needs to be prioritized for selection in order to meet the long-term fairness constraint.

To validate the efficiency of our proposed CLFC algorithm, we provide a theoretical performance guarantee, showing that the fairness constraints are approximately satisfied and that meanwhile the achieved time-average regret compared to the optimal oracle is upper bounded by $\frac{N}{2V}+c_1\frac{N}{T}+(c_2+c_3\log T+c_4\sqrt{\log T})\sqrt{\frac{mN}{T}}$, where N is the total number of arms, m is maximum number of arms that can be played simultaneously in each round, T is the total rounds played, c_1, c_2, c_3 and c_4 are appropriately chosen constants, and V is the parameter governing the regret and fairness constraint satisfaction tradeoff. The time-average regret bound includes two parts: the constant part (due to the lack of offline information) independent of the rounds T and the time-varying part (due to the unknown parameters in reward functions which need to be learnt) that shrinks with T. Next, we run simulations on both real-world and synthetic datasets to further demonstrate the effectiveness of our proposed algorithm. Our results show that by selecting the parameter V, the balance between fairness constraints and the estimated reward can be adjusted and, importantly, a negligible time-average regret can be achieved.

The rest of the paper is organized as follows. Related work is discussed in Section II. Then, we formulate the contextual combinatorial sleeping bandits problem in Section III. Next, we introduce the solution approach on which our proposed algorithm is based, and details of our CLFC algorithm in Section IV. Our theoretical and experimental results are presented in Sections V and VI correspondingly. Finally, we conclude our work in Section VII.

II. Related Work

In the basic MAB setting without contextual information, arm selection is simply performed by trial and exploitation. This basic setting originates from Robbins' seminal work in [6] that developed an approach for constructing asymptotically efficient rules to achieve the greatest possible expected value of the total reward. Then, [7] proposed to use confidence bounds to handle the tradeoff between exploitation and exploration, which is also the first UCB-style algorithm. The subsequent study [8] showed that the optimal logarithmic regret can be achieved uniformly over time.

The most notable work on contextual MAB is perhaps [7], which proposed the first $\widetilde{O}(\sqrt{Td})$ algorithm called LinRel. Another important algorithm LinUCB was proposed in [4], based on UCB [8] and KWIK [9] algorithms to devise them for contextual bandits. However, [4] only presented experiments in real system, but no theoretical analysis of the regret was provided. Then, [10] theoretically analyzed a variant of LinUCB, by decomposing it into BaseLinUCB and SupLinUCB, proving a regret upper bound of $O(\sqrt{Td\ln^3(KT\ln T)/\delta})$.

Several studies have also extended the basic MAB setting by incorporating practical factors and consideration, such as combinatorial decisions ([11] - [12]) and intermittently available arms ([13] - [14]). Notably, [11] is the first to study sampling m processes simultaneously out of N i.i.d. processes. Then [15] defined an effective and efficient general framework for combinatorial bandits to select a super arm. Later in the following work [16], this framework was further extended to handle general nonlinear reward functions. Research on sleeping bandits dates back to [13], where an online learning problem with time-varying sets of available actions was studied. Despite the abundant research on extending the MAB model to address the combinatorial decisions and intermittently available arms, most work studied these two factors separately, not considering them simultaneously. In contrast, we add both combinatorial and sleeping bandits as constraints to arm selection at runtime.

Short-term constraints have also been considered in the literature on MAB ([17], [18]). For example, [17] investigated the MAB problem with budget models for advertisement display. Different types of budgets were considered, including conditions where the advertiser has a fixed budget over a time horizon and the amount of available money is incremented in each time slot. A regret upper bound of $O(\log T)$ was given. However, these types of constraints are significantly different from the long-term constraints we are considering in this work.

The long-term fairness constraints on arm selection we consider in this work are ones that have only been recently considered. Specifically, the total number of times each arm is selected over the entire time horizon must exceed a certain threshold under the fairness constraint, and hence it is very different from the constraints in many prior studies. A small but growing set of studies have considered other types of long-term constraints on arm selection ([19] - [20]). For example, [19] studied a setting where pulling an arm incurs a random

cost, while the total cost needs to meet a budget constraint. The scenario in [21] is that each arm is associated with 2-level rewards, and the objective is to maximize the compound rewards while satisfying a minimum reward for the total level-1 rewards. Nevertheless, besides the totally different long-term constraint types considered, another major difference between our work is that their requirements (budget constraint or total reward) are for all the arms together, instead of for each individual arm.

The notion of fairness has been studied as a constraint in the prior MAB research (e.g., [22], [23]), but the fairness definition in these works is often from the resource allocation perspective and hence is orthogonal to the one we consider. The most relevant work to ours is [2] which studies combinatorial sleeping bandits with long-term fairness constraints. Nonetheless, a key difference that separates our study apart from [2] is that we consider contextual information, which results in a different and refined arm selection policy subject to incoming contexts. Further, unlike [2], our algorithm results in a different regret bound that depends on the dimensionality of contextual information vectors (Section V).

III. Problem Formulation

In this section, we formulate the problem of contextual combinatorial sleeping bandits with fairness constraints.

A. Model and Fairness Constraints

Let N denote the number of arms and T be the total rounds played. For each arm a played in round t, the reward is $r_{t,a}$. We make the standard assumption that $r_{t,a} \in [0,1]$ is an unknown *a priori* random variable. Since some arms might be unavailable or sleeping in a round, the assumption is that the set of available arms M_t is revealed to the player/learner at the beginning of round t. In each round, the player can choose multiple but no more than m arms, which constitute a subset of M_t. Suppose that the feasible arm set selected in round t is U_t ($|U_t| \leq m$). Then, the compound reward R_t received instantly after round t is a weighted sum of reward of every single arm in U_t, defined as $R_t \triangleq \sum_{a \in U_t} \omega_a r_{t,a}$, where ω_a is the weight of arm a (fixed and known). The arm weight vector is $\boldsymbol{\omega} = (\omega_1, \omega_2, ..., \omega_N)$. Thus, the optimization objective is:

$$\max_{U_t \in \mathcal{P}(M_t)} \sum_{t=0}^{T-1} \sum_{a \in U_t} \omega_a r_{t,a} \tag{1}$$

where $\mathcal{P}(M_t)$ denotes the power set of M_t, namely, all the feasible combination of available arms at time t.

While there are different notions of fairness, we follow the fairness definition studied in [2], which captures various practical scenarios such as business agreement that requires an item (e.g., news, movie) be displayed to users for at least a certain number of times. Concretely, to incorporate fairness constraints, we define a binary variable $y_{t,a}$ for arm a to indicate if it is selected in round t or not. If selected, then $y_{t,a} = 1$, otherwise $y_{t,a} = 0$. Obviously for each $a \in U_t$, $y_{t,a} = 1$. Note that $\sum_{a=1}^{N} y_{t,a} \leq m$ should be satisfied due to the maximum number of combinatorial arms that can be selected in each round t. Supposing that the minimum selection rate requirements of arm a is $f_a \in [0,1]$ (fixed and known), the fairness constraints can also be equivalently modeled as

$$\limsup_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[y_{t,a}] \geq f_a. \tag{2}$$

We define a fairness constraint vector for all the arms as $\boldsymbol{f} = (f_1, f_2, ..., f_N)$. Such a $\boldsymbol{f}$ is said to be feasible if there exists at least one policy or algorithm that outputs a feasible arm set U_t in round $t = 0, 1, \cdots, T-1$. Then, the set of all feasible $\boldsymbol{f}$ is called *maximal feasibility region*, denoted by $\mathbb{C}$.

B. Contextual Information

We now model the contextual information following the literature [4]. In each round t, for each available arm a, the player observes a d-dimensional feature vector $x_{t,a} \in \mathbb{R}^d$, called context, which contains information of (possibly) both the player herself and the arm a. It satisfies that $||x_{t,a}|| \leq 1$, where $||\cdot||$ is the l_2-norm without loss of generality. After observing the feature vectors of all the available arms, the player selects an arm set U_t and receives an reward $\sum_{a \in U_t} \omega_a r_{t,a}$. Based on [4], we consider a linear reward function, which means that there exists an unknown weight vector $\theta_a \in \mathbb{R}^d$ with $||\theta_a|| \leq 1$ such that

$$\mathbb{E}[r_{t,a}|x_{t,a}] = x_{t,a}^T \theta_a \tag{3}$$

for each round t and arm a. The player needs to learn $\theta_a \in \mathbb{R}^d$ over the course of learning. While the usefulness of linear reward functions has been widely demonstrated in practice (e.g., modeling user preferences towards an news article) [4], we also note that nonlinear reward functions can be first projected into a reproducing kernel Hilbert space and then handled similarly as linear reward functions, which is beyond the scope of our current work.

C. Regret

Regret is a standard metric to measure the loss incurred by an arm selection algorithm A, namely the reward difference between the expected optimal reward achieved by an oracle and the actual expected one obtained by the player. Suppose that the maximum expected total reward achieved by the oracle over the time horizon T is R^*. Then, the time-average regret can be defined as follows

$$R_A(T) \triangleq \frac{1}{T} R^* - \mathbb{E}[\frac{1}{T} \sum_{t=0}^{T-1} \sum_{a \in U_t} \omega_a r_{t,a}] \tag{4}$$

Note that minimzing the regret $R_A(T)$ is equivalent to the reward maximization objective in (1), as the benchmark $\frac{1}{T} R^*$ is independent of the learning algorithm.

TABLE I: Notations

Notation	Description
N	Total number of arms
$\mathcal{A}$	Set of all arms
T	Totoal number of rounds played
M_t	Set of available arms in round t
$\mathcal{P}(M_t)$	Power set of M_t
m	Maximum number of simultaneously played arms
U_t	The feasible arm set selected in round t
$\mathcal{P}(U_t)$	Power set of U_t
ω_a	Weight of arm a
ω_{max}	Maximum weight among all the arms
$\boldsymbol{\omega}$	Weight vector for all the arms
f_a	Required minimum selection fraction for arm a
$\boldsymbol{f}$	Fairness constraint vector for all the arms
$r_{t,a}$	Reward of arm a in round t
R_t	Compound reward of selected feasible arm set U_t in round t
$y_{t,a}$	Binary variable to indicate whether arm a is selected in round t or not; $y_{t,a}=1$ if selected
$R_A(T)$	Time-average regret of algorithm A on time horizon T
R^*	Maximum expected total compound reward on time horizon T
$x_{t,a}$	d-dimensional feature vector of arm a observed in round t
$p_{t,a}$	LinUCB estimate of arm a in round t
$A_{t,a}$	Design matrix of arm a in round t
$b_{t,a}$	Response vector of arm a in round t
$Q_{t,a}$	Virtual queue length of arm a in round t
p_a	Probability that arm a is available in each round
$\boldsymbol{p}$	Probability of availability vector of all the arms

IV. CLFC: Online Learning Algorithm

Maximizing the accumulative rewards (1) while guaranteeing the long-term fairness constraints for each arm (2) is challenging. Concretely, the long-term fairness constraint naturally couples all the arm selections over time, adding significant challenges to the online learning problem. In fact, even when the unknown parameters $\theta_a \in \mathbb{R}^d$ are perfectly known for each arm a, the problem of online arm selection is difficult. This is because solving the arm selection problem to meet the long-term fairness constraints requires the complete offline information, whereas only current contextual information and set of currently available arms are known at each round.

In this section, we address these challenges and develop a novel online learning algorithm, called CLFC, based on two main techniques: LinUCB and Lyapunov optimization. Specifically, we extend LinUCB algorithm [4] to provide an estimate of each individual arm's expected reward using contextual information and the virtual fairness queue lengths that represent how far each arm selection up to the current round are away from the fairness constraints. Next, we show the details of our algorithm CLFC.

Achieving a balance between exploration and exploitation is the key of maximizing cumulative rewards for bandit problems. Without loss of generality, we employ the classic LinUCB algorithm with disjoint linear models [4] to compute an upper confidence bound of the estimated payoff for each arm. Specifically, given the model in Section III, the payoff $p_{t,a}$ estimated by LinUCB for arm a in round t is given as

$$p_{t,a} \triangleq \hat{\theta}_{t,a}^T x_{t,a} + \alpha\sqrt{x_{t,a}^T A_{t,a}^{-1} x_{t,a}} \tag{5}$$

where $\hat{\theta}_{t,a} = A_{t,a}^{-1} b_{t,a}$ is an estimate of the unknown coefficient θ_a in (3), $A_{t,a}$ is a certain design matrix of arm a at time t (to be specified in Algorithm 1), $b_{t,a}$ is the corresponding response vector of arm a, and α is a constant which can be computed as $\alpha = 1 + \sqrt{ln(2/\delta)/2}$ in which δ is defined such that with a probability of at least $1-\delta$, $|\hat{\theta}_{t,a}^T x_{t,a} - \mathbb{E}[r_{t,a}|x_{t,a}]| \leq \alpha\sqrt{x_{t,a}^T A_{t,a}^{-1} x_{t,a}}$ stands. The first term of (5) corresponds to exploitation, while the second term represents exploration to avoid being stuck in a local optimum.

In the standard contextual bandit setting, the LinUCB algorithm simply chooses the arm with largest estimated payoff $p_{t,a}$, namely $a_t = \arg\max_{a\in\mathcal{A}} p_{t,a}$. Nonetheless, our problem has three additional key constraints: combinatorial bandits, sleeping bandits, and fairness constraints. Thus, directly following the LinUCB algorithm is far from being enough. Specifically, combinatorial and sleeping bandits add more restrictions to the reward optimization in the sense that some arms might be unavailable and a combination of arms can be selected, thus making the reward in each round a weighted sum of rewards obtained by each selected single arm. More importantly, fairness constraints bring another challenge, since arms need to be guaranteed with at least a certain number of selections besides the exploration-exploitation balance. These factors altogether add challenges to the contextual combinatorial sleeping bandits problem, invalidating the standard LinUCB algorithm in a basic contextual MAB setting.

Queueing dynamics for fairness constraints. While the combinatorial and sleeping bandits are per-round and can be solved by adding constraints on arm selection at each round, fairness is a long-term constraint that couples the arm selections all over the entire time horizon.

To relax the long-term fairness constraints, we employ the Lyapunov optimization framework [5], in which the state of a time-slotted system at a particular time can be represented by a multi-dimensional vector of queues. Specifically, any entity satisfying the "coming and serving" fashion (e.g., network packets) can be modeled as a real or virtual queue. In the context of fairness constraints, an arm being selected can be treated as "serving", while the selection ratio requirement as "coming". Then, we can create a virtual queue for each arm a to capture the fairness constraint, and use $Q_{t,a}$ to denote the virtual queue length of arm a at time t. Therefore, the queue dynamic can be written as

$$Q_{t+1,a} = \max\left[Q_{t,a} - y_{t,a} + f_a, 0\right] \tag{6}$$

for $t = 0, 1, ..., T-2$, with $Q_{0,a} = 0$ for all arms a. In this queue dynamic, at time $t+1$, there is accumulated amount of $Q_{t,a}$ from the past history, an incoming amount of f_a for arm a at time t that needs to be processed, and $y_{t,a}$ is the served amount depending on the arm selection algorithm. The length of virtual queue should be non-negative, thus adding the $\max\{\ ,\ \}$ operator. In essence, if the fairness

queue length is zero or sufficiently small, then the long-term fairness constraint is also (approximately) satisfied for the corresponding arm. Thus, instead of directly considering the long-term fairness constraint, we instead seek to keep the fairness queues short for all the arms when designing our online learning algorithm.

Arm selection. The virtual fairness queues indicate the fairness deficits — how far each arm selection up to the current round are away from the fairness constraint. In other words, a longer queue means that the corresponding arm has not been selected adequately and hence needs to be prioritized for selection in order to meet the long-term fairness constraint.

Based on this intuition, we integrate the fairness queue lengths into our arm selection procedure and modify the LinUCB algorithm for arm selectionas follows:

$$U_t = \arg \max_{U \in \mathcal{P}(M_t)} \sum_{a \in U} (Q_{t,a} + V \cdot \omega_a p_{t,a}) \tag{7}$$

where V is the control parameter that strikes a balance between maximizing the total reward and meeting the long-term fairness constraints. The first component of (7) corresponds to the virtual queue length that represents the runtime importance of selecting an arm due to its fairness constraint, and the second component is the weighted accumulative payoff estimated by the LinUCB algorithm for selected arms, and $U \in \mathcal{P}(M_t)$ and summation symbol $\sum_{a \in U}$ reflect the combinatorial and available arm set constraints, respectively.

Remarks. The new arm selection rule modified based on the LinUCB algorithm prioritizes arms that have longer fairness deficits and arms with large estimated payoffs, with V being the control parameter adjusting their relative weights. With a larger V, the algorithm resembles the standard LinUCB more while paying less attention to the fairness constraints, and vice versa. In Section V, we provide a formal analysis on the role of V in reward maximization and fairness constraint satisfaction.

Another point worth mentioning is the complexity of our arm selection algorithm. In general, with a budget of at most m arms that can be selected simultaneously (i.e., $|U_t| \leq m$), the size of power set $\mathcal{P}(U_t)$ is 2^m, thus resulting in an exponential complexity. Nonetheless, in (7), arms are selected based on a weighted linear sum of functions in terms of individual selected arm. Therefore, the arm selection complexity reduces exponential to linear complexity in our algorithm. Concretely, we can choose up to m top arms with the largest "$Q_{t,a} + V \cdot \omega_a p_{t,a}$" value subject to the available arm set constraint.

Finally, the details of CLFC are shown in Algorithm 1, and the mathematical notations are listed in Table I. In the CLFC algorithm, we firstly initialize the virtual queue length $Q_{t,a}$ of each arm to 0. At each round t, we go through the standard LinUCB procedure: updating the estimated unknown parameter $\hat{\theta}_{t,a}$, observing the time-varying context and computing the LinUCB estimate $p_{t,a}$ for each arm a. Then, we update the queue length $Q_{t,a}$ for each arm according to (6). Next, we observe the available arm set M_t, compute the "$Q_{t,a} + V \cdot \omega_a p_{t,a}$" value for each arm in M_t, and then choose up to m top arms with the largest "$Q_{t,a} + V \cdot \omega_a p_{t,a}$" to form arm selection set U_t. Finally, we play the arms in U_t, observe the rewards and update $y_{t,a}$, $A_{t,a}$, $b_{t,a}$ for arms in U_t.

Algorithm 1 Contextual Combinatorial Sleeping Bandits with Fairness Constraints (CLFC)

for $a \in \mathcal{A}$ **do**
  Initialize $Q_{t,a} = 0$;
end for
for $t = 0, 1, 2, ..., T-1$ **do**
  for $a \in \mathcal{A}$ **do**
    if a is new **then**
      $A_{t,a} \leftarrow I_{d \times d}$ (d-dimensional identity matrix);
      $b_{t,a} \leftarrow 0_{d \times 1}$ (d-dimensional zero vector);
    end if
    Observe features of arm $a \in \mathcal{A}$: $x_{t,a} \in \mathbb{R}^d$;
    $\hat{\theta}_{t,a} \leftarrow A_{t,a}^{-1} b_{t,a}$;
    $p_{t,a} \leftarrow \hat{\theta}_{t,a}^T x_{t,a} + \alpha \sqrt{x_{t,a}^T A_{t,a}^{-1} x_{t,a}}$;
    $Q_{t+1,a} \leftarrow \max[Q_{t,a} - y_{t,a} + f_a, 0]$;
  end for
  Observe the set of available arms M_t;
  Select feasible arm set U_t according to (7);
  Play arms in U_t;
  for $a \in U_t$ **do**
    Observe the real-valued reward $r_{t,a}$;
    Set $y_{t,a} = 1$ (for $a \notin U_t$, set $y_{t,a} = 0$);
    $A_{t,a} \leftarrow A_{t,a} + x_{t,a} x_{t,a}^T$;
    $b_{t,a} \leftarrow b_{t,a} + r_{t,a} x_{t,a}$;
  end for
end for

V. Theoretical Results

In this section, we present our main theoretical results and show the sketch of proof, whose details can be found in the [24]. First, since we address the fairness constraints via virtual queue techniques, it is necessary to guarantee the virtual queue stability defined in (6), translating into fairness constraint satisfaction. More precisely, it needs to be proved that the virtual queueing system is mean rate stable for any arrival rate vector $\boldsymbol{f}$ inside the capacity region (i.e., feasible fairness constraints). Then, we derive the time-average regret upper bound of our CLFC algorithm to show the learning efficiency.

A. Fairnes Constraint Satisfaction

It is established from [5] that if the virtual queue defined in (6) is *mean rate stable*, then we have $\limsup_{T \to \infty} \frac{\mathbb{E}[\sum_{a=1}^{N} Q_{t,a}]}{T} \leq 0$. This inequality means that the served amount would be larger than incoming amount in the long run, thus guaranteeing the long-term fairness constraints for arm selection. Therefore, in order to show that CLFC meets the fairness constraints, it suffices to prove the virtual queue stability for any minimum selection fraction vector $\boldsymbol{f}$ strictly inside the maximal feasibility region $\mathbb{C}$. Towards this end, we consider a stronger

stability notion called *strong stability*, which also implies *mean rate stability* and hence fairness constraint satisfaction.

Theorem 1. *For any minimum selection fraction vector $\boldsymbol{f}$ strictly inside the maximal feasibility region $\mathbb{C}$, the virtual queue defined in (6) is strongly stable. Namely:*

$$\limsup_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1}\sum_{a=1}^{N}\mathbb{E}[Q_{t,a}] \leq \frac{B}{\phi} < \infty \tag{8}$$

where B is a constant defined as $B \triangleq \frac{N}{2} + Vm\omega_{max}$, and ω_{max} is the maximum weight among all the arms.

Proof. The proof can be established utilizing the Lyapunov Drift Theorem. Denote the virtual queue vector by $\boldsymbol{Q_t} = (Q_{1,t}, Q_{2,t}, ..., Q_{N,t})$ for all the N queues at time t. Then, we define the Lyapunov function as $L(\boldsymbol{Q_t}) \triangleq \frac{1}{2}\sum_{a=1}^{N} Q_{t,a}^2$. Then Lyapunov function drift is:

$$L(\boldsymbol{Q_{t+1}}) - L(\boldsymbol{Q_t}) \leq \frac{N}{2} + \sum_{a=1}^{N} f_a Q_{t,a} - \sum_{a=1}^{N} y_{t,a} Q_{t,a} \tag{9}$$

Taking conditional expectation of both sides gives the *conditional Lyapunov drift* for slot t:

$$\begin{aligned}
\triangle(\boldsymbol{Q_t}) &\triangleq \mathbb{E}[L(\boldsymbol{Q_{t+1}}) - L(\boldsymbol{Q_t})|\boldsymbol{Q_t}] \\
&\leq \frac{N}{2} + \sum_{a=1}^{N} f_a Q_{t,a} - \mathbb{E}[\sum_{a=1}^{N} y_{t,a} Q_{t,a}|\boldsymbol{Q_t}] \\
&\leq B + \sum_{a=1}^{N} f_a Q_{t,a} - \mathbb{E}[\sum_{a\in U_t} (Q_{t,a} + V\omega_a p_{t,a})|\boldsymbol{Q_t}] \\
&\leq B - \phi \sum_{a=1}^{N} Q_{t,a}
\end{aligned}$$

where ϕ is defined as follows: if $\boldsymbol{f}$ is strictly inside the maximal feasibility region $\mathbb{C}$, then there exists such ϕ that $\boldsymbol{f} + \phi\mathbf{1}$ is also strictly inside $\mathbb{C}$.

Finally, we know from the Lyapunov Drift Theorem in [5] that if the inequality above is satisfied, then the queue $\boldsymbol{Q_t}$ is strongly stable, namely $\limsup_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1}\sum_{a=1}^{N}\mathbb{E}[Q_{t,a}] \leq \frac{B}{\phi} < \infty$. □

Remark. Theorem 1 states that as long as the long-term fairness (or equivalently, minimum selection) constraints $\boldsymbol{f}$ are feasible, then CLFC ensures that they are satisfied in the long term with a bounded queue backlog (i.e., bounded temporary fairness deficit/violation), even though the complete offline information is not available to CLFC.

B. Regret Upper Bound on Reward

In addition to fairness constraint satisfaction, CLFC has a bounded regret compared to the oracle, as shown below.

Theorem 2. *For the time-average regret of* CLFC *defined in (4), the upper bound is:*

$$\begin{aligned}
R_A(T) \leq \frac{N}{2V} &+ \omega_{max} N \frac{\delta}{T} \\
&+ \omega_{max}\sqrt{\frac{Nm}{T}}(4\sqrt{d\log(\lambda + \frac{TL}{d})}) \\
&(\lambda^{\frac{1}{2}}S + R\sqrt{2\log(\frac{1}{\delta}) + d\log(1 + \frac{TL}{\lambda d})}) \\
&+ \alpha\sqrt{2d\log(\lambda + \frac{TL}{d})})
\end{aligned} \tag{10}$$

where S is the upper bound of l_2-norm of θ_a ($||\theta_a|| \leq S$), L is the upper bound of $||x_{t,a}||$, R is from the assumption that the noise sequence is subject to R-sub-Gaussian distribution, d is the dimension of $x_{t,a}$, λ is from the definition of design matrix $A_{t,a} \triangleq D_{t,a}^T D_{t,a} + \lambda I_d$, and δ is defined as follows: with probability at least $1-\delta$, $\sum_{t=0}^{T-1} x_{t,a}^T(\hat{\theta}_{t,a} - \theta_a) \leq 4\sqrt{Td\log(\lambda + \frac{TL}{d})}(\lambda^{\frac{1}{2}}S + R\sqrt{2\log(\frac{1}{\delta}) + d\log(1 + \frac{TL}{\lambda d})})$ is satisfied.

Proof. Based on the Lyapunov optimization framework, we can include other weighted objectives into the Lyapunov drift to give a drift-plus-penalty, which is a modified optimization objective. In our problem setting, minimizing the time-average regret defined in (4) is another objective besides the virtual queue stability (fairness constraint satisfaction) and can be written as:

$$R_A(T) = \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[\sum_{a\in U_t^*}\omega_a r_{t,a} - \sum_{a\in U_t}\omega_a r_{t,a}] \tag{11}$$

where U_t^* is the optimal arm set selection in round t. Define $P(t) \triangleq \sum_{a\in U_t^*}\omega_a r_{t,a} - \sum_{a\in U_t}\omega_a r_{t,a}$ as the penalty term to be added to Lyapunov drift (9), and the resulting drift-plus-regret is given by:

$$\begin{aligned}
&L(\boldsymbol{Q_{t+1}}) - L(\boldsymbol{Q_t}) + VP(t) \\
\leq &\frac{N}{2} + \sum_{a=1}^{N}(Q_{t,a} + V\omega_a r_{t,a})(y_{t,a}^* - y_{t,a}) \\
&+ \sum_{a=1}^{N} Q_{t,a}(f_a - y_{t,a}^*)
\end{aligned} \tag{12}$$

where $y_{t,a}^*$ indicates whether arm a is included in U_t^* or not. The expectation of (12) is bounded by:

$$\mathbb{E}[L(\boldsymbol{Q_{t+1}}) - L(\boldsymbol{Q_t}) + VP(t)] \leq \frac{N}{2} + \mathbb{E}[K_1(t)] \tag{13}$$

where $K_1(t) \triangleq \sum_{i=1}^{N}(Q_{t,a} + V\omega_a r_{t,a})(y_{t,a}^* - y_{t,a})$. Then, using the telescope sum, we can show the time-average regret is bounded by:

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[P(t)] \leq \frac{N}{2V} + \frac{1}{VT}\sum_{t=0}^{T-1}\mathbb{E}[K_1(t)] \tag{14}$$

Next, based on the regret analysis in [25] and [10], we prove the final regret upper bound is:

$$R_A(T) \leq \frac{N}{2V} + \omega_{max} N \frac{\delta}{T} + \omega_{max}\sqrt{\frac{Nm}{T}}(4\sqrt{d\log(\lambda + \frac{TL}{d})} (\lambda^{\frac{1}{2}} S + R\sqrt{2\log(\frac{1}{\delta}) + d\log(1+\frac{TL}{\lambda d})}) + \alpha\sqrt{2d\log(\lambda+\frac{TL}{d})})$$

This concludes the proof (details are available in [24]). □

Remark: The regret bound can be separated into two parts. The first term $\frac{N}{2V}$ of our regret bound is caused by the lack of complete offline information to ensure long-term fairness constraints, which is inversely proportionally to V. Intuitively, a larger V means that more priority is given to reward maximization instead of fairness constraints, resulting in a smaller regret (at the expense of a potentially larger fairness queue backlog as shown in Theorem 1). In contrast, if V is small, CLFC tends to favor arms with a longer queue to reduce the fairness queue backlog and ensure fairness constraints. The other terms in our regret bound correspond to the gap between optimal and estimated rewards due to the unknown parameters in reward functions that need to be learnt over time. Importantly, for this time-average learning-induced regret which shrinks over time (equivalently, the cummulative regret grows sublinearly with time T), its form aligns with the regret analysis in [25].

VI. Performance Evaluation

In this section, we first introduce some real-world applications to which our contextual combinatorial sleeping bandits with fairness constraints apply. Then, we run simulations on two selected applications — movie recommendation and weight matrix matching — for empirical evaluations.

A. Applications

1) Product Recommendation: When a user visits an online shopping website, the website administrator dynamically chooses multiple products from the product pool to display on the page. Assume there are N products in the pool, and the display limit is m products simultaneously. The context in this setting includes users' profile (gender, order history, viewing history, age, location, etc.) and products' characteristics (e.g., category, price, style, etc.). Since some products might be irrelevant to certain users, they can be treated as unavailable to those users. Nowadays, it is common that shopping websites collaborate with brands to promote for them, with profits for exchange. Thus, the website administrator also needs to guarantee a minimum display frequency of its partners' products, regardless of users' responses to them. After seeing a recommended product, the user might click or not click it, and after clicking, she might end up purchasing or not purchasing. The click-through rate is unknown *a priori*, and each click will generate a (multi-level) revenue for the website, which is the reward.

The goal of the product recommender is to maximize the cumulative revenues by selecting a subset of products to display in the face of unknown click-through rates. The minimum display frequency that needs to be guaranteed for certain products corresponds to the fairness constraint in our framework.

2) Stock Portfolio Building: A portfolio manager's job responsibility is to choose a combination of stocks from the stock pool for her clients. Assume that there are N stocks in total, and that the client is willing to invest in only m of them each time. The context in this setting consists of clients' profile (gender, investment history, salary, savings, etc.) and stocks' characteristics (e.g., profits history, company profile etc.). Since some stocks might be irrelevant to certain clients, they can be considered as unavailable to those clients. Besides, portfolio manager also needs to guarantee a minimum selection frequency for certain stocks, regardless of clients' response to them. This might because of manager's personal professional preference, or partnership with certain companies. After receiving the portfolio advice, the client might invest or not invest in each stock. The investment rate is unknown *a priori*, and each investment will generate a revenue for the portfolio manager.

The goal of the portfolio manager is to maximize the cumulative revenues by selecting a subset of stocks to recommend to the client in the face of unknown investment rate. The minimum selection frequency that needs to be guaranteed for certain stocks corresponds to the fairness constraint in our framework.

3) Multiuser Channel Allocation: Consider the problem of multiuser channel allocation in a cognitive radio network with minimum utilization constraints, and time is slotted. Assume there are N orthogonal channels and M secondary users. Each secondary user requires a single channel to transmit packets. There is a parameter associated with each user-channel pair, which translates the original throughput into a reward. The parameter is unknown and needs to be learnt. Note than some channels might be unavailable because of poor channel conditions (e.g., deep channel fading or occupied by primary users). We assume that the reward corresponding to each user-channel pair is a random variable and its mean is unknown *a priori*. In this problem setting, denote the parameter between user i and channel j by θ_{ij} and the throughput achieved by the user by $p_{ij,t}$. Thus, the sum reward is $r_t = \sum_{i=1}^{M}\sum_{j=1}^{N}\theta_{ij}p_{ij,t}$.

The reward in this application is a linearly weighted combination of the throughput, and the context is the user-channel throughput matrix, the unknown parameter is the weight matrix converting throughput into a reward. The relationship between reward and context is linear. The goal is to maximize the accumulative reward in the face of unknown weight matrix. Since there can be utilization constraints for each channel, a minimum selection ratio must be guaranteed for each arm.

B. Experimental Results

Next, we choose two applications — movie recommendation and weight matrix matching — for empirical evaluations of CLFC.

1) Movie Recommendation: We apply CLFC to the MovieLens 100k dataset [26] to simulate the movie recommendation application. The dataset consists of 943 users' rating on 1682 movies. The rating scale is 1-5 (score is 0 if no rating available), which we normalize to 0-1 since we assume that $r_{t,a} \in [0,1]$ in section III-A. To make sure there are enough data points for learning, we pick N most rated movies and eliminate users whose total number of rating for these N movies are less than m, since m arms are played simultaneously in each round. For each of these N arms in round t, we assign a feature vector $x_{t,a}$ which is concatenation of the user's occupation, gender and the movie's genre. To simulate the sleeping bandits, we assume the availability of arms are subject to *i.i.d.* binomial distribution in each round, with mean $\boldsymbol{p} = (p_1, p_2, ..., p_N)$.

We experiment with $N = 3$ arms, the number of simultaneously played arms $m = 2$, the arm weights $\boldsymbol{\omega} = (\omega_1, \omega_2, \omega_3) = (1,1,1)$, minimum selection fraction $\boldsymbol{f} = (f_1, f_2, f_3) = (0.7, 0.4, 0.8)$, and the probability distribution of arms is $\boldsymbol{p} = (p_1, p_2, p_3) = (0.8, 0.6, 0.9)$. Since MovieLens 100k dataset has discrete rating values and a linear relationship between reward and the feature vector $x_{t,a}$ may not be guaranteed, we add perturbation noise to part of the data points to ensure as much linearity as possible. We run the simulation for $T = 20000$ rounds, which is enough for CLFC to converge. The optimal reward we use to compute the time-average regret (4) is to replace the second term (LinUCB estimate) in (7) with the actual optimal reward.

The baseline we choose is the modified Learning with Linear Rewards algorithm (LLR) [27], called LLR for Sleeping bandits (LLRS). The arm selection policy of LLRS is to simply choose the m arms with largest LinUCB estimate ($U_t = \arg\max_{U_t \in \mathcal{P}(M_t)} \sum_{a \in U_t} p_{t,a}$), no fairness constraint is considered. Namely, LLRS follows the LinUCB policy only with restrictions of combinatorial and sleeping bandits. Note that the algorithm in [2] cannot handle contexts and hence does not apply to our setting. On the other hand, the fairness-oblivious LLRS achieves the lowest regret compared to any fairness-constrained learning algorithms. Thus, LLRS is a powerful baseline to compare CLFC with.

Fig. 1 shows the time-average regret under different V values, which is a critical parameter. We can see that the time-average regret converges to a small value under all four cases. The larger the V is, the smaller time-average regret achieved. Indeed, V controls the weight between fairness and reward. If V is extremely large, the problem becomes almost equivalent to reward maximization in contextual MAB without fairness constraints. In comparison, the time-average regret of LLRS is the best. This is beacuse LLRS obliviously chooses the arms with largest estimated reward, which is similar to the case when V is approaching infinity. Fig. 2 demonstrates each

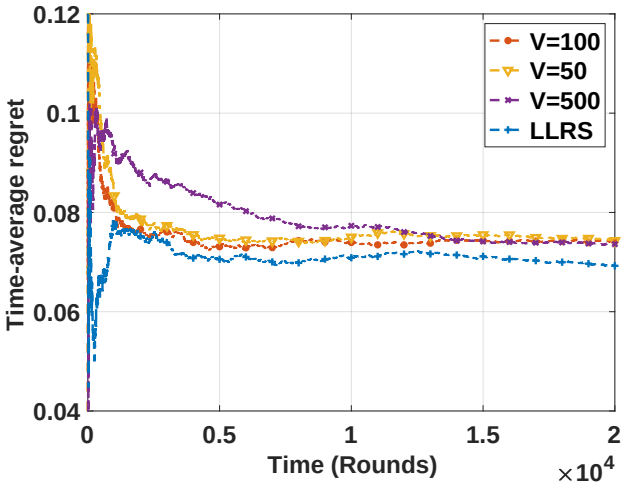


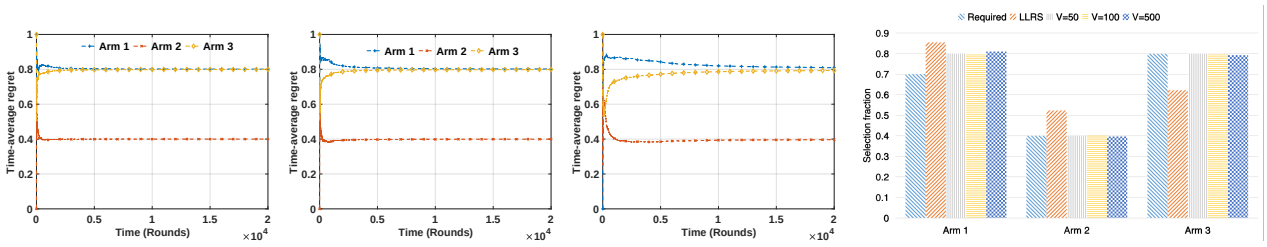
Fig. 1: Time-average regret under different values of V.

arm's selection fraction change over time under different V values. We can see that $V = 50$ and $V = 100$ achieve better time-average regret while meeting the fairness constraints for each individual arm. Fig. 2d shows the selection statistics of each arm under different V values, compared to the required minimum selection ratio. Note that under LLRS, the fairness constraint of arm 3 is violated, which is the cost of achieving the best time-average regret.

2) Multiuser Channel Allocation: In this application, we utilize synthetic dataset to simulate CLFC. Channel is the arm, reward is the weighted throughput, weight is the unknown parameter we need to learn, and the known original throughput is the context. We generate a 1000×10 reward matrix, each element of which denotes the real reward of arm a in round t, and hence the number of arms is $N = 10$. Each reward is in the range $[0,1]$. To simulate the sleeping bandits, we consider that the availability of arms are subject to *i.i.d.* binomial distribution in each round, with mean $\boldsymbol{p} = (p_1, p_2, ..., p_N)$.

We experiment with $N = 10$ arms, the number of simultaneously played arms $m = 6$, the arm weight $\boldsymbol{\omega} = (1,1,1,1,1,1,1,1,1,1)$, minimum selection fraction $\boldsymbol{f} = (0.3, 0.4, 0.8, 0.2, 0.5, 0.7, 0.8, 0.1, 0.3, 0.4)$, and the probability distribution of arms is $\boldsymbol{p} = (0.5, 0.6, 0.9, 0.3, 0.5, 0.8, 0.9, 0.2, 0.5, 0.6)$. We simulate for T=20000 rounds in this application. The optimal reward we use to compute the time-average regret (4) is still to replace the second term (LinUCB estimate) in (7) with the actual reward.

Fig. 3 shows the time-average regret under different V values. Time-average regret decreases with larger V and LLRS achieves the best regret, which is consistent with results on the movie dataset. A negligible value is achieved under all V values, and all are smaller than the regret in previous experiment. This is because our synthesized dataset is much more ideal and strictly linear than the MovieLens 100K. Fig. 4 demonstrates each arm's selection fraction change over time under different V values. With this ideal dataset, fairness constraints are met



(a) V=50 (b) V=100 (c) V=500 (d) Statistics

Fig. 2: Selection fraction of 3 arms under different values V.

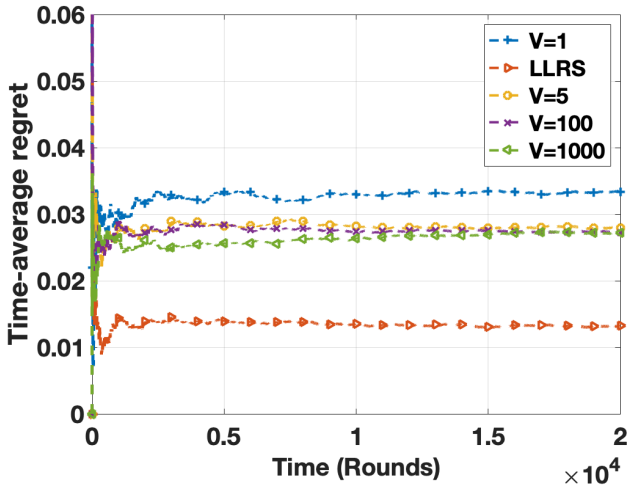
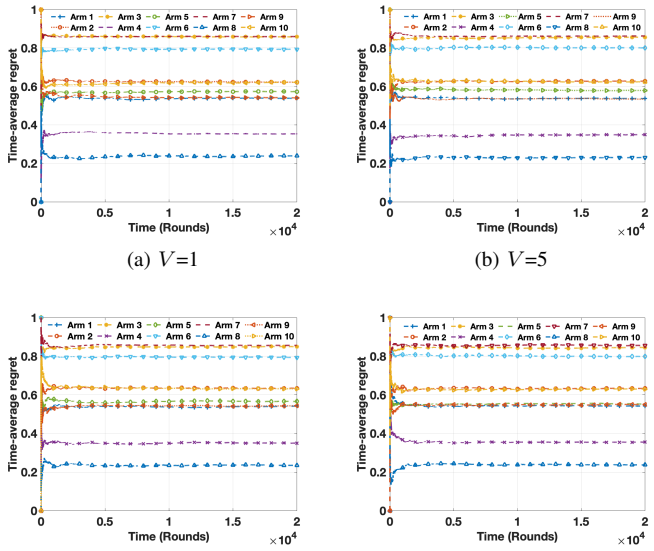


Fig. 3: Time-average regret under different values of V.



(a) V=1 (b) V=5 (c) V=100 (d) V=1000

Fig. 4: Selection fraction of 10 arms over time under different values of V.

regardless of the V. Fig. 5 shows the selection ratio statistics of each arm under different V values. All the arms are guaranteed with the required selection ratios except for arm 3 under LLRS. Since LLRS is fairness-oblivious, violation of fairness is the cost of achieving the best regret.

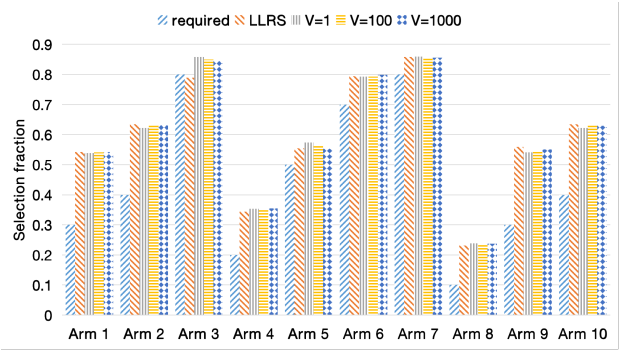


Fig. 5: Selection fraction of 10 arms over time under different values of V.

VII. Conclusions

In this work, we formulate the contextual MAB problem with combinatorial, sleeping bandits and importantly fairness constraints, which is defined as a required minimum selection ratio for each arm over the entire time horizon. Then, we propose CLFC based on the LinUCB and Lyapunov optimization framework. We not only show the fairness constraint satisfaction, regret upper bound of CLFC, but also run simulations on both real-world and synthetic datasets to validate CLFC. Our results show that by tuning the control parameter V, time-average regret of CLFC converges to a negligible value while meeting the fairness constraints. In particular, a significant difference between the literature and our work is that context information is exploited, which serves as a reference for accurate arm selection given incoming contexts.

References

[1] S. Bubeck, N. Cesa-Bianchi, *et al.*, "Regret analysis of stochastic and nonstochastic multi-armed bandit problems," *Foundations and Trends® in Machine Learning*, vol. 5, no. 1, pp. 1–122, 2012.

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

[3] L. Zhou, "A survey on contextual multi-armed bandits," *arXiv preprint arXiv:1508.03326*, 2015.

[4] 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*, pp. 661–670, ACM, 2010.

[5] M. J. Neely, "Stochastic network optimization with application to communication and queueing systems," *Synthesis Lectures on Communication Networks*, vol. 3, no. 1, pp. 1–211, 2010.

[6] T. L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Advances in applied mathematics*, vol. 6, no. 1, pp. 4–22, 1985.

[7] P. Auer, "Using confidence bounds for exploitation-exploration tradeoffs," *Journal of Machine Learning Research*, vol. 3, no. Nov, pp. 397–422, 2002.

[8] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine learning*, vol. 47, no. 2-3, pp. 235–256, 2002.

[9] T. J. Walsh, I. Szita, C. Diuk, and M. L. Littman, "Exploring compact reinforcement-learning representations with linear regression," in *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pp. 591–598, AUAI Press, 2009.

[10] 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*, pp. 208–214, 2011.

[11] V. Anantharam, P. Varaiya, and J. Walrand, "Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays-part i: Iid rewards," *IEEE Transactions on Automatic Control*, vol. 32, no. 11, pp. 968–976, 1987.

[12] R. Combes, M. S. T. M. Shahi, A. Proutiere, *et al.*, "Combinatorial bandits revisited," in *Advances in Neural Information Processing Systems*, pp. 2116–2124, 2015.

[13] R. Kleinberg, A. Niculescu-Mizil, and Y. Sharma, "Regret bounds for sleeping experts and bandits," *Machine learning*, vol. 80, no. 2-3, pp. 245–272, 2010.

[14] A. Chatterjee, G. Ghalme, S. Jain, R. Vaish, and Y. Narahari, "Analysis of thompson sampling for stochastic sleeping bandits.," in *UAI*, 2017.

[15] W. Chen, Y. Wang, and Y. Yuan, "Combinatorial multi-armed bandit: General framework and applications," in *International Conference on Machine Learning*, pp. 151–159, 2013.

[16] W. Chen, W. Hu, F. Li, J. Li, Y. Liu, and P. Lu, "Combinatorial multi-armed bandit with general reward functions," in *Advances in Neural Information Processing Systems*, pp. 1659–1667, 2016.

[17] R. Combes, C. Jiang, and R. Srikant, "Bandits with budgets: Regret lower bounds and optimal algorithms," *ACM SIGMETRICS Performance Evaluation Review*, vol. 43, no. 1, pp. 245–257, 2015.

[18] A. Badanidiyuru, R. Kleinberg, and A. Slivkins, "Bandits with knapsacks," in *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pp. 207–216, IEEE, 2013.

[19] E. V. Denardo, E. A. Feinberg, and U. G. Rothblum, "The multi-armed bandit, with constraints," *Annals of Operations Research*, vol. 208, no. 1, pp. 37–62, 2013.

[20] K. Chen, K. Cai, L. Huang, and J. Lui, "Beyond the click-through rate: Web link selection with multi-level feedback," *arXiv preprint arXiv:1805.01702*, 2018.

[21] K. Cai, X. Liu, Y.-Z. J. Chen, and J. C. Lui, "An online learning approach to network application optimization with guarantee," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*, pp. 2006–2014, IEEE, 2018.

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

[23] M. S. Talebi and A. Proutiere, "Learning proportionally fair allocations with low regret," *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 2, no. 2, p. 36, 2018.

[24] Appendix. https://www.dropbox.com/s/dnal2fj1w8aviek/Infocom2020_%231570578380_Appendix.pdf?dl=0.

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

[26] https://grouplens.org/datasets/movielens/.

[27] Y. Gai, B. Krishnamachari, and R. Jain, "Combinatorial network optimization with unknown variables: Multi-armed bandits with linear rewards and individual observations," *IEEE/ACM Transactions on Networking (TON)*, vol. 20, no. 5, pp. 1466–1478, 2012.

VIII. Appendix

A. Proof of Theorem 1

Proof. The Lyapunov function is defined as:

$$L(\boldsymbol{Q_t}) \triangleq \frac{1}{2}\sum_{a=1}^{N} Q_{t,a}^2 \tag{15}$$

Then Lyapunov function drift is:

$$\begin{aligned}
&L(\boldsymbol{Q_{t+1}}) - L(\boldsymbol{Q_t}) \\
&= \frac{1}{2}\sum_{a=1}^{N} Q_{t+1,a}^2 - \frac{1}{2}\sum_{a=1}^{N} Q_{t,a}^2 \\
&\overset{(a)}{\leq} \frac{1}{2}\sum_{a=1}^{N}(Q_{t,a} - y_{t,a} + f_a)^2 - \frac{1}{2}\sum_{a=1}^{N} Q_{t,a}^2 \\
&= \frac{1}{2}\sum_{a=1}^{N}(y_{t,a} - f_a)^2 - \sum_{a=1}^{N}(y_{t,a} - f_a)Q_{t,a} \\
&\overset{(b)}{\leq} \frac{N}{2} + \sum_{a=1}^{N} f_a Q_{t,a} - \sum_{a=1}^{N} y_{t,a}Q_{t,a}
\end{aligned} \tag{16}$$

where (a) is from the virtual queue length definition in (6), and (b) is because of $y_{t,a}$=0 or 1 and $f_a \in [0,1]$, thus $(y_{t,a}-f_a)^2 \leq 1$. Then taking conditional expectation of both sides gives the *conditional Lyapunov drift* for slot t:

$$\begin{aligned}
&\mathbb{E}[L(\boldsymbol{Q_{t+1}}) - L(\boldsymbol{Q_t})|\boldsymbol{Q_t}] \\
&\leq \frac{N}{2} + \sum_{a=1}^{N} f_a Q_{t,a} - \mathbb{E}[\sum_{a=1}^{N} y_{t,a}Q_{t,a}|\boldsymbol{Q_t}] \\
&= \frac{N}{2} + \sum_{a=1}^{N} f_a Q_{t,a} - \mathbb{E}[\sum_{a\in U_t} Q_{t,a}|\boldsymbol{Q_t}] \\
&= \frac{N}{2} + \sum_{a=1}^{N} f_a Q_{t,a} + \mathbb{E}[\sum_{a\in U_t} V\omega_a r_{t,a}|\boldsymbol{Q_t}] \\
&- \mathbb{E}[\sum_{a\in U_t}(Q_{t,a} + V\omega_a r_{t,a})|\boldsymbol{Q_t}] \\
&\overset{(a)}{\leq} \frac{N}{2} + \sum_{a=1}^{N} f_a Q_{t,a} + mV\omega_{max} \\
&- \mathbb{E}[\sum_{a\in U_t}(Q_{t,a} + V\omega_a r_{t,a})|\boldsymbol{Q_t}] \\
&= B + \sum_{a=1}^{N} f_a Q_{t,a} - \mathbb{E}[\sum_{a\in U_t}(Q_{t,a} + V\omega_a r_{t,a})|\boldsymbol{Q_t}]
\end{aligned} \tag{17}$$

where (a) is because $r_{t,a} \in [0,1]$ and at most m arms are played simultaneously. B is a constant defined as $B \triangleq \frac{N}{2} + Vm\omega_{max}$, and ω_{max} is the maximum weight among all the arms.

Then, combine the steps (16) and (17) in the proof of **Theorem 1** in [2], we have:

$$\begin{aligned}
&\mathbb{E}[L(\boldsymbol{Q_{t+1}}) - L(\boldsymbol{Q_t})|\boldsymbol{Q_t}] \\
&\leq B + \sum_{a=1}^{N} f_a Q_{t,a} - \sum_{a=1}^{N} Q_{t,a}(f_a + \phi) \\
&= B - \phi\sum_{a=1}^{N} Q_{t,a}
\end{aligned} \tag{18}$$

Finally, we know from the Lyapunov Drift Theorem in [5] that if (18) is satisfied, then the queue $\boldsymbol{Q_t}$ is strongly stable and $\limsup_{T\to\infty} \frac{1}{T}\sum_{t=0}^{T-1}\sum_{a=1}^{N}\mathbb{E}[Q_{t,a}] \leq \frac{B}{\phi} < \infty$.

□

B. Proof of Theorem 2

Proof. Suppose the optimal arm set selected in round t is U_t^*, and the corresponding binary variable for arm a is $y_{t,a}^*$. Thus the optimal accumulated reward on the time horizon of T is:

$$R^* = \mathbb{E}[\sum_{t=0}^{T-1}\sum_{a\in U_t^*}\omega_a r_{t,a}] \tag{19}$$

The time-average regret of our algorithm is:

$$\begin{aligned}
R_A(T) &= \frac{1}{T}R^* - \frac{1}{T}\mathbb{E}[\sum_{t=0}^{T-1}\sum_{a\in U_t}\omega_a r_{t,a}] \\
&= \frac{1}{T}\mathbb{E}[\sum_{t=0}^{T-1}\sum_{a\in U_t^*}\omega_a r_{t,a}] - \frac{1}{T}\mathbb{E}[\sum_{t=0}^{T-1}\sum_{a\in U_t}\omega_a r_{t,a}] \\
&= \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[\sum_{a\in U_t^*}\omega_a r_{t,a} - \sum_{a\in U_t}\omega_a r_{t,a}] \\
&= \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}[P(t)]
\end{aligned} \tag{20}$$

where we define:

$$\begin{aligned}
P(t) &\triangleq \sum_{a\in U_t^*}\omega_a r_{t,a} - \sum_{a\in U_t}\omega_a r_{t,a} \\
&= \sum_{a=1}^{N}\omega_a r_{t,a} y_{t,a}^* - \sum_{a=1}^{N}\omega_a r_{t,a} y_{t,a}
\end{aligned} \tag{21}$$

Then $P(t)$ is the penalty term (scaled by V) to be added to Lyapunov drift (16), and the drift-plus-regret is given by:

$$\begin{aligned}
&L(\boldsymbol{Q_{t+1}}) - L(\boldsymbol{Q_t}) + VP(t) \\
&\leq \frac{N}{2} + \sum_{a=1}^{N} f_a Q_{t,a} - \sum_{a=1}^{N} y_{t,a}Q_{t,a} \\
&+ V\sum_{a=1}^{N}\omega_a r_{t,a} y_{t,a}^* - V\sum_{a=1}^{N}\omega_a r_{t,a} y_{t,a} \\
&= \frac{N}{2} + \sum_{a=1}^{N}(Q_{t,a} + V\omega_a r_{t,a})(y_{t,a}^* - y_{t,a}) \\
&+ \sum_{a=1}^{N} Q_{t,a}(f_a - y_{t,a}^*)
\end{aligned} \tag{22}$$

Take expectation of both sides of the drift-plus-regret:

$$\begin{aligned}
&\mathbb{E}[L(\boldsymbol{Q_{t+1}}) - L(\boldsymbol{Q_t}) + VP(t)] \\
&\leq \frac{N}{2} + \sum_{a=1}^{N} \mathbb{E}[(Q_{t,a} + V\omega_a r_{t,a})(y^*_{t,a} - y_{t,a})] \\
&+ \sum_{a=1}^{N} \mathbb{E}[Q_{t,a}(f_a - y^*_{t,a})] \\
&\overset{(a)}{\leq} \frac{N}{2} + \sum_{a=1}^{N} \mathbb{E}[(Q_{t,a} + V\omega_a r_{t,a})(y^*_{t,a} - y_{t,a})] \\
&= \frac{N}{2} + \mathbb{E}[\sum_{a=1}^{N} (Q_{t,a} + V\omega_a r_{t,a})(y^*_{t,a} - y_{t,a})]
\end{aligned} \tag{23}$$

where is (a) is because of $\mathbb{E}[Q_{t,a}(f_a - y^*_{t,a})] = \mathbb{E}[Q_{t,a}]\mathbb{E}[f_a - y^*_{t,a}]$ (current arm selection decision only affects future queue lengths) and $\mathbb{E}[y^*_{t,a}] \geq f_a$ (fairness constraints (2)). Then $\mathbb{E}[Q_{t,a}(f_a - y^*_{t,a})] \leq 0$ and thus (a).

Let $K_1(t) = \sum_{a=1}^{N} (Q_{t,a} + V\omega_a r_{t,a})(y^*_{t,a} - y_{t,a})$. Then sum (23) for time horizon T, using the telescope summing trick we have:

$$\begin{aligned}
&\mathbb{E}[L(\boldsymbol{Q_T}) - L(\boldsymbol{Q_0})] + V \sum_{t=0}^{T-1} \mathbb{E}[P(t)] \\
&\leq T\frac{N}{2} + \sum_{t=0}^{T-1} \mathbb{E}[K_1(t)]
\end{aligned} \tag{24}$$

Divide both sides of (24) by VT we have:

$$\begin{aligned}
&\frac{1}{VT}(\mathbb{E}[L(\boldsymbol{Q_T}) - L(\boldsymbol{Q_0})] + V \sum_{t=0}^{T-1} \mathbb{E}[P(t)]) \\
&\leq \frac{N}{2V} + \frac{1}{VT} \sum_{t=0}^{T-1} \mathbb{E}[K_1(t)]
\end{aligned} \tag{25}$$

From the definition of Lyapunov function (15) we know that $L(\boldsymbol{Q_t}) \geq 0$ for any t. Since $L(\boldsymbol{Q_0}) = 0$, we have:

$$\begin{aligned}
&\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[P(t)] \\
&\leq \frac{1}{VT}(\mathbb{E}[L(\boldsymbol{Q_T}) - L(\boldsymbol{Q_0})]) + \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[P(t)]) \\
&\leq \frac{N}{2V} + \frac{1}{VT} \sum_{t=0}^{T-1} \mathbb{E}[K_1(t)]
\end{aligned} \tag{26}$$

where $\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}[P(t)])$ is the time-average regret we want to bound. In what follows, we will show the bound of $\frac{1}{VT}\sum_{t=0}^{T-1} \mathbb{E}[K_1(t)]$.

1) Bounding $K_1(t)$: Assume there is another arm set selection policy which chooses arm set $U_t^{'}$ in round t (different from U_t in our C3 algorithm and the optimal selection U_t^*) according to the following rule:

$$U_t^{'} = \underset{U_t' \in \mathcal{P}(M_t)}{argmax} \sum_{a \in U_t'} (Q_{t,a} + V\omega_a p_{t,a}) \tag{27}$$

Since our C3 algrithm chooses arm set U_t according to alternative of (7), we have:

$$\sum_{a \in U_t} (Q_{t,a} + V\omega_a p_{t,a}) \geq \sum_{a \in U_t^{'}} (Q_{t,a} + V\omega_a p_{t,a}) \tag{28}$$

For $K_1(t)$ we have:

$$\begin{aligned}
K_1(t) &= \sum_{a=1}^{N} (Q_{t,a} + V\omega_a r_{t,a})(y^*_{t,a} - y_{t,a}) \\
&= \sum_{a=1}^{N} (Q_{t,a} + V\omega_a r_{t,a}) y^*_{t,a} - \sum_{a=1}^{N} (Q_{t,a} + V\omega_a r_{t,a}) y_{t,a} \\
&= \sum_{a \in U_t^*} (Q_{t,a} + V\omega_a r_{t,a}) - \sum_{a \in U_t} (Q_{t,a} + V\omega_a r_{t,a}) \\
&\overset{(a)}{\leq} \sum_{a \in U_t^{'}} (Q_{t,a} + V\omega_a r_{t,a}) - \sum_{a \in U_t} (Q_{t,a} + V\omega_a r_{t,a}) \\
&\overset{(b)}{\leq} \sum_{a \in U_t^{'}} (Q_{t,a} + V\omega_a r_{t,a}) - \sum_{a \in U_t} (Q_{t,a} + V\omega_a r_{t,a}) \\
&+ \sum_{a \in U_t} (Q_{t,a} + V\omega_a p_{t,a}) - \sum_{a \in U_t^{'}} (Q_{t,a} + V\omega_a p_{t,a}) \\
&= V(\sum_{a \in U_t} \omega_a (p_{t,a} - r_{t,a}) + \sum_{a \in U_t^{'}} \omega_a (r_{t,a} - p_{t,a})) \\
&= V(K_2(t) + K_3(t))
\end{aligned} \tag{29}$$

where we define $K_2(t) \triangleq \sum_{a \in U_t} \omega_a (p_{t,a} - r_{t,a})$ and $K_3(t) \triangleq \sum_{a \in U_t^{'}} \omega_a (r_{t,a} - p_{t,a})$. (a) is from (27) and (b) is from (28). Thus the time-average regret can be written as:

$$\begin{aligned}
&\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[P(t)] \\
&\leq \frac{N}{2V} + \frac{1}{VT} \sum_{t=0}^{T-1} \mathbb{E}[K_1(t)] \\
&\leq \frac{N}{2V} + \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[K_2(t) + K_3(t)] \\
&= \frac{N}{2V} + \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[K_2(t)] + \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[K_3(t)]
\end{aligned} \tag{30}$$

Next we will show the bound of $K_2(t)$ and $K_3(t)$ respectively.

C. Bounding $K_2(t)$

Denote the round in which arm a is played for i-th time by t_a^i, and number of times arm a has been played by round t by $z_{t,a}$. Thus we have $y_{t_a^i,a} = 1$, $z_{t_a^i,a} = i$ and $z_{t_a^i - 1,a} = i - 1$, where $i = 1, 2, ..., z_{T-1,a}$. And obviously $0 \leq t_a^1 < t_a^2 < \cdots < t_a^{z_{T-1,a}} < T$.

$$
\begin{aligned}
&\sum_{t=0}^{T-1}\mathbb{E}[K_2(t)] = \sum_{t=0}^{T-1}\mathbb{E}[\sum_{a\in U_t}\omega_a(p_{t,a}-r_{t,a})] \\
&= \sum_{t=0}^{T-1}\mathbb{E}[\sum_{a=1}^{N}\omega_a(p_{t,a}-r_{t,a})y_{t,a}] \\
&= \sum_{a=1}^{N}\mathbb{E}[\sum_{t=0}^{T-1}\omega_a(p_{t,a}-r_{t,a})y_{t,a}] \\
&\le \omega_{max}\sum_{a=1}^{N}\mathbb{E}[\sum_{t=0}^{T-1}(p_{t,a}-r_{t,a})y_{t,a}] \\
&= \omega_{max}\sum_{a=1}^{N}\mathbb{E}[\sum_{i=1}^{z_{T-1,a}}(p_{t_a^i,a}-r_{t_a^i,a})]
\end{aligned} \tag{31}
$$

where ω_{max} is the largest weight among all the arms.

From (3) and (5) we know that $p_{t_a^i,a} = x_{t_a^i,a}^T\hat{\theta}_{t_a^i,a} + \alpha\sqrt{x_{t_a^i,a}^T A_{t_a^i,a}^{-1}x_{t_a^i,a}}$ and $r_{t_a^i,a} = x_{t_a^i,a}^T\theta_a$. For simplicity, denote t_a^i by t_1. Thus:

$$
\begin{aligned}
&\sum_{i=1}^{z_{T-1,a}}(p_{t_a^i,a}-r_{t_a^i,a}) \\
&= \sum_{i=1}^{z_{T-1,a}}x_{t_1,a}^T(\hat{\theta}_{t_1,a}-\theta_a) + \sum_{i=1}^{z_{T-1,a}}\alpha\sqrt{x_{t_1,a}^T A_{t_1,a}^{-1}x_{t_1,a}}
\end{aligned} \tag{32}
$$

Based on the proof of **Theorem 3** in [25], for the first term we have:

$$
\begin{aligned}
&\sum_{i=1}^{z_{T-1,a}}x_{t_1,a}^T(\hat{\theta}_{t_1,a}-\theta_a) \\
&\le \sum_{i=1}^{z_{T-1,a}}2\sqrt{\beta_{t_a^{i-1}}(\delta)}||x_{t1,a}||^2_{A_{t_1,a}^{-1}} \\
&\le \sqrt{z_{T-1,a}\sum_{i=1}^{z_{T-1,a}}(x_{t_1,a}^T(\hat{\theta}_{t_1,a}-\theta_a))^2} \\
&\le \sqrt{8\beta_{z_{T-1,a}}(\delta)z_{T-1,a}\sum_{i=1}^{z_{T-1,a}}\min(||x_{t1,a}||^2_{A_{t_1,a}^{-1}},1)} \\
&\le 4\sqrt{\beta_{z_{T-1,a}}(\delta)z_{T-1,a}\log(\det(A_{z_{T-1,a}}))} \\
&\le 4\sqrt{z_{T-1,a}d\log(\lambda+z_{T-1,a}\frac{L}{d})}(\lambda^{\frac{1}{2}}S \\
&+ R\sqrt{2\log(\frac{1}{\delta})+d\log(1+\frac{z_{T-1,a}L}{\lambda d})}) \\
&\le 4\sqrt{d\log(\lambda+\frac{TL}{d})}\sqrt{z_{T-1,a}} \\
&(\lambda^{\frac{1}{2}}S + R\sqrt{2\log(\frac{1}{\delta})+d\log(1+\frac{z_{T-1,a}L}{\lambda d})}) \\
&\le 4\sqrt{d\log(\lambda+\frac{TL}{d})}\sqrt{z_{T-1,a}} \\
&(\lambda^{\frac{1}{2}}S + R\sqrt{2\log(\frac{1}{\delta})+d\log(1+\frac{TL}{\lambda d})})
\end{aligned} \tag{33}
$$

For the second term we have:

$$
\begin{aligned}
&\sum_{i=1}^{z_{T-1,a}}\alpha\sqrt{x_{t_1,a}^T A_{t_1,a}^{-1}x_{t_1,a}} = \alpha\sum_{i=1}^{z_{T-1,a}}||x_{t_1,a}||_{A_{t_1,a}^{-1}} \\
&\le \alpha\sqrt{z_{T-1,a}\sum_{i=1}^{z_{T-1,a}}||x_{t_1,a}||^2_{A_{t_1,a}^{-1}}} \\
&\le \alpha\sqrt{2z_{T-1,a}\log(\det(A_{t_1,a}))} \\
&\le \alpha\sqrt{2z_{T-1,a}d\log(\lambda+z_{T-1,a}\frac{L}{d})} \\
&\le \alpha\sqrt{2d\log(\lambda+\frac{TL}{d})}\sqrt{z_{T-1,a}}
\end{aligned} \tag{34}
$$

Then $\sum_{t=0}^{T-1}\mathbb{E}[K_2(t)]$ can be written as:

$$
\begin{aligned}
&\sum_{t=0}^{T-1}\mathbb{E}[K_2(t)] \\
&\le \omega_{max}\sum_{a=1}^{N}\mathbb{E}[\sum_{t=0}^{T-1}(p_{t,a}-r_{t,a})y_{t,a}] \\
&\le \omega_{max}\sum_{a=1}^{N}\mathbb{E}[4\sqrt{d\log(\lambda+\frac{TL}{d})}\sqrt{z_{T-1,a}} \\
&(\lambda^{\frac{1}{2}}S + R\sqrt{2\log(\frac{1}{\delta})+d\log(1+\frac{TL}{\lambda d})}) \\
&+ \alpha\sqrt{z_{T-1,a}}\sqrt{2d\log(\lambda+\frac{TL}{d})}] \\
&= \omega_{max}(4\sqrt{d\log(\lambda+\frac{TL}{d})} \\
&(\lambda^{\frac{1}{2}}S + R\sqrt{2\log(\frac{1}{\delta})+d\log(1+\frac{TL}{\lambda d})}) \\
&+ \alpha\sqrt{2d\log(\lambda+\frac{TL}{d})})\sum_{a=1}^{N}\mathbb{E}[\sqrt{z_{T-1,a}}] \\
&\overset{(a)}{\le} \omega_{max}(4\sqrt{d\log(\lambda+\frac{TL}{d})} \\
&(\lambda^{\frac{1}{2}}S + R\sqrt{2\log(\frac{1}{\delta})+d\log(1+\frac{TL}{\lambda d})}) \\
&+ \alpha\sqrt{2d\log(\lambda+\frac{TL}{d})})\sqrt{NTm}
\end{aligned} \tag{35}
$$

where (a) is from Jensen's inequality and thus $\frac{1}{N}\sum_{a=1}^{N}\sqrt{z_{T-1,a}} \le \sqrt{\frac{1}{N}\sum_{a=1}^{N}z_{T-1,a}} \le \sqrt{\frac{Tm}{N}}$ (since at most m arms can be played simultaneously).

D. Bounding $K_3(t)$

$$
\begin{aligned}
\mathbb{E}[K_3(t)] &= \mathbb{E}[\sum_{a\in U_t'}\omega_a(r_{t,a}-p_{t,a})] \\
&= \mathbb{E}[\sum_{a=1}^{N}\omega_a(r_{t,a}-p_{t,a})y_{t,a}']
\end{aligned} \tag{36}
$$

Define an event as $E(t) \triangleq \{p_{t,a} < r_{t,a}\}$

$$\begin{aligned}
&\mathbb{E}[\sum_{a=1}^{N} \omega_a (r_{t,a} - p_{t,a}) y_{t,a}^{'}] \\
&= \mathbb{E}[\sum_{a=1}^{N} \omega_a (r_{t,a} - p_{t,a}) y_{t,a}^{'} \mathbb{I}_{\{E(t)\}}] \\
&+ \mathbb{E}[\sum_{a=1}^{N} \omega_a (r_{t,a} - p_{t,a}) y_{t,a}^{'} \mathbb{I}_{\{E^c(t)\}}] \\
&\overset{(a)}{\leq} \mathbb{E}[\sum_{a=1}^{N} \omega_a (r_{t,a} - p_{t,a}) y_{t,a}^{'} \mathbb{I}_{\{E(t)\}}] \\
&\leq \omega_{max} \sum_{a=1}^{N} \mathbb{E}[(r_{t,a} - p_{t,a}) y_{t,a}^{'} \mathbb{I}_{\{E(t)\}}]
\end{aligned} \tag{37}$$

where (a) is beacuse when $E^c(t)$ happens, $r_{t,a} \leq p_{t,a}$. Define $K_4(t) \triangleq (r_{t,a} - p_{t,a}) y_{t,a}^{'} \mathbb{I}_{\{E(t)\}}$, we have:

$$\begin{aligned}
&\mathbb{E}[K_4(t)] = \mathbb{E}[(r_{t,a} - p_{t,a}) y_{t,a}^{'} \mathbb{I}_{\{E(t)\}}] \\
&\overset{(a)}{\leq} \mathbb{E}[\mathbb{I}_{\{E(t)\}}] \\
&= Pr[p_{t,a} - r_{t,a} < 0]
\end{aligned} \tag{38}$$

where (a) is because when $E(t)$ happens, $p_{t,a} < r_{t,a} \in [0, 1]$ and $y_{t,a}^{'}$ is 0 or 1, so $(r_{t,a} - p_{t,a}) y_{t,a}^{'} \leq 1$. And notation $Pr[\,]$ denotes the probability.

We know from **Lemma 1** in [10] that with probability at least $1 - \delta/T$ that $|\hat{\theta}_{t,a}^T x_{t,a} - \theta_a^T x_{t,a}| \leq \alpha \sqrt{x_{t,a}^T A_{t,a}^{-1} x_{t,a}}$. Namely:

$$Pr[|\hat{\theta}_{t,a}^T x_{t,a} - \theta_a^T x_{t,a}| > \alpha \sqrt{x_{t,a}^T A_{t,a}^{-1} x_{t,a}}] < \delta/T \tag{39}$$

Thus we have:

$$\begin{aligned}
&Pr[\hat{\theta}_{t,a}^T x_{t,a} - \theta_a^T x_{t,a} > \alpha \sqrt{x_{t,a}^T A_{t,a}^{-1} x_{t,a}}] \\
&+ Pr[\hat{\theta}_{t,a}^T x_{t,a} - \theta_a^T x_{t,a} < -\alpha \sqrt{x_{t,a}^T A_{t,a}^{-1} x_{t,a}}] < \delta/T
\end{aligned} \tag{40}$$

Then:

$$\begin{aligned}
&Pr[p_{t,a} - r_{t,a} < 0] = Pr[p_{t,a} < r_{t,a}] \\
&= Pr[\hat{\theta}_{t,a}^T x_{t,a} + \alpha \sqrt{x_{t,a}^T A_{t,a}^{-1} x_{t,a}} < \theta_a^T x_{t,a}] \\
&= Pr[\hat{\theta}_{t,a}^T x_{t,a} - \theta_a^T x_{t,a} < -\alpha \sqrt{x_{t,a}^T A_{t,a}^{-1} x_{t,a}}] \\
&\leq Pr[\hat{\theta}_{t,a}^T x_{t,a} - \theta_a^T x_{t,a} > \alpha \sqrt{x_{t,a}^T A_{t,a}^{-1} x_{t,a}}] \\
&+ Pr[\hat{\theta}_{t,a}^T x_{t,a} - \theta_a^T x_{t,a} < -\alpha \sqrt{x_{t,a}^T A_{t,a}^{-1} x_{t,a}}] < \delta/T
\end{aligned} \tag{41}$$

Now we have $\mathbb{E}[K_4(t)] = Pr[p_{t,a} - r_{t,a} < 0] < \delta/T$, and $\mathbb{E}[K_3(t)] \leq \omega_{max} \sum_{a=1}^{N} \mathbb{E}[K_4(t)] < \omega_{max} \frac{N\delta}{T}$.

Finally, combine the bound of $\mathbb{E}[K_2(t)]$ and $\mathbb{E}[K_3(t)]$ together:

$$\begin{aligned}
&\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[P(t)] \\
&\leq \frac{N}{2V} + \frac{1}{VT} \sum_{t=0}^{T-1} \mathbb{E}[K_1(t)] \\
&\leq \frac{N}{2V} + \frac{1}{VT} \sum_{t=0}^{T-1} \mathbb{E}[V(K_2(t) + K_3(t))] \\
&= \frac{N}{2V} + \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[K_2(t)] + \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E}[K_3(t)] \\
&< \frac{N}{2V} + \frac{1}{T} \omega_{max} (4 \sqrt{d \log(\lambda + \frac{TL}{d})} \\
&(\lambda^{\frac{1}{2}} S + R \sqrt{2 \log(\frac{1}{\delta}) + d \log(1 + \frac{TL}{\lambda d})}) \\
&+ \alpha \sqrt{2d \log(\lambda + \frac{TL}{d})}) \sqrt{NTm} \\
&= \frac{N}{2V} + \omega_{max} \sqrt{\frac{Nm}{T}} (4 \sqrt{d \log(\lambda + \frac{TL}{d})} \\
&(\lambda^{\frac{1}{2}} S + R \sqrt{2 \log(\frac{1}{\delta}) + d \log(1 + \frac{TL}{\lambda d})}) \\
&+ \alpha \sqrt{2d \log(\lambda + \frac{TL}{d})}) + \omega_{max} \frac{N\delta}{T}
\end{aligned} \tag{42}$$

□