Contextual Combinatorial Volatile Multi-armed Bandit with Adaptive Discretization

Andi Nika **Sepehr Elahi** **Cem Tekin**

Department of Electrical and Electronics Engineering, Bilkent University, Ankara, Turkey

Abstract

We consider contextual combinatorial volatile multi-armed bandit (CCV-MAB), in which at each round, the learner observes a set of available base arms and their contexts, and then, selects a super arm that contains K base arms in order to maximize its cumulative reward. Under the semi-bandit feedback setting and assuming that the contexts lie in a space $\mathcal{X}$ endowed with the Euclidean norm and that the expected base arm outcomes (expected rewards) are Lipschitz continuous in the contexts (expected base arm outcomes), we propose an algorithm called *Adaptive Contextual Combinatorial Upper Confidence Bound* (ACC-UCB). This algorithm, which adaptively discretizes $\mathcal{X}$ to form estimates of base arm outcomes and uses an α-approximation oracle as a subroutine to select a super arm in each round, achieves $\tilde{O}(T^{(\bar{D}+1)/(\bar{D}+2)+\epsilon})$ regret for any $\epsilon > 0$, where $\bar{D}$ represents the approximate optimality dimension related to $\mathcal{X}$. This dimension captures both the benignness of the base arm arrivals and the structure of the expected reward. In addition, we provide a recipe for obtaining more optimistic regret bounds by taking into account the volatility of the base arms and show that ACC-UCB achieves significant performance gains compared to the state-of-the-art for worker selection in mobile crowdsourcing.

1 INTRODUCTION

The multi-armed bandit (MAB) is a prominent example of the sequential decision-making paradigm under uncertainty [Thompson, 1933, Lai and Robbins, 1985]. In its classical version, the learner selects arms sequentially over rounds, one at a time from a finite set, in order to maximize its cumulative reward. It faces the challenge of exploration and exploitation as it is not aware of the arm reward distributions beforehand and observes in each round only the reward of the selected arm. The metric used to evaluate the performance of the learner is called the (expected) regret, which is defined as the cumulative loss of the learner with respect to an oracle that acts optimally based on arms' reward distributions. It is known that maximizing the cumulative reward is equivalent to minimizing the regret. Many algorithms have been proposed to minimize the regret by balancing exploration and exploitation. Two notable examples are upper confidence bound (UCB) based index policies [Lai and Robbins, 1985, Agrawal, 1995, Auer et al., 2002] and Thompson sampling [Thompson, 1933, Agrawal and Goyal, 2012, Russo and Van Roy, 2014].

An important extension of the standard MAB is the contextual MAB [Langford and Zhang, 2007, Lu et al., 2010, Chu et al., 2011, Slivkins, 2014], where at the beginning of each round the learner observes side-information, also called the context, about the arm rewards in that particular round. As the learner tries to maximize its cumulative reward by taking this information into account, its regret is typically measured with respect to an oracle that selects in each round the best arm given the context of that round. Contextual MAB algorithms have been used in a variety of applications ranging from personalized news article recommendation [Li et al., 2010] to sequential decision-making in mobile healthcare [Tewari and Murphy, 2017].

Another important extension of the standard MAB is the combinatorial MAB, where in each round the learner chooses a subset of base arms, also called the super arm, and obtains a reward that depends on the outcomes of the base arms that are in the chosen super arm [Cesa-Bianchi and Lugosi, 2012, Gai et al., 2012, Chen et al., 2013, Kveton et al., 2015b]. This problem is mainly investigated under the semi-bandit feedback setting, where the learner also observes outcomes of the selected base arms. It is also extended to handle the cases when some base arms can only get probabilistically triggered [Kveton et al., 2015a, Chen

Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS) 2020, Palermo, Italy. PMLR: Volume 108.

Table 1: Comparison with related work.

Properties	This work	[Li et al., 2016]	[Chen et al., 2018]	[Chen et al., 2013]	[Bubeck et al., 2011]	[Slivkins, 2014]	[Kleinberg et al., 2010]
Contextual	Yes	Yes	Yes	No	No	Yes	No
Combinatorial	Yes	Yes	Yes	Yes	No	No	No
Volatile (base) arms	Yes	No	Yes	No	No	Yes	Yes
Arm and/or context space	Infinite	Finite/infinite	Infinite	Finite	Infinite	Infinite	Finite
Adaptive discretization	Yes	No	No	No	Yes	Yes	No
Reward function	General	General	Submodular	General	General	General	General
Feedback type	Semi-bandit	Cascading	Semi-bandit	Semi-bandit	Full-bandit	Full-bandit	Full-bandit
Computation oracle	α	α	$(1-1/e)$	(α, β)	Exact	Exact	Exact

et al., 2016, Huyuk and Tekin, 2019]. Combinatorial MAB have found applications in slate recommendation [Radlinski et al., 2008], crowdsourcing [Yang et al., 2017] and online influence maximization [Chen et al., 2016].

Deviating from the aforementioned works, another strand of literature considers MAB with time-varying arm sets under the names *sleeping* MAB [Kleinberg et al., 2010] or *volatile* MAB [Bnaya et al., 2013], both of which are remnants of *mortal* MAB [Chakrabarti et al., 2009]. In this setting, the learner tries to select the best of the available arms in each round to maximize its cumulative reward. The concept of volatility is quite common in applications that involve sequential decision making. For instance, in online advertising, ads become unavailable after they expire [Chakrabarti et al., 2009]. Similarly, in crowdsourcing, the set of available tasks and workers may change over time [Jain et al., 2017].

We consider CCV-MAB that includes the characteristics of all the MAB models mentioned above: (i) base arms are volatile, (ii) expected outcome of a base arm depends on its context, (iii) the learner selects a super arm in each round that consists of a subset of the available base arms, observes outcomes of the selected base arms and receives a reward that depends on these outcomes. We propose an algorithm called ACC-UCB that achieves $\tilde{O}(T^{(\bar{D}+1)/(\bar{D}+2)+\epsilon})$ regret for any $\epsilon > 0$ with respect to an α-approximation oracle under the assumptions that the expected base arm outcomes and the expected rewards are Lipschitz continuous in the contexts and the expected base arm outcomes respectively. Here, dimension $\bar{D}$, which can usually be much smaller than the dimension D of the context space, captures the benignness of the base arm arrivals and the structure of the expected reward. Our model and algorithm can be applied to solve dynamic resource allocation problems ranging from crowdsourcing to online advertising to multi-user channel allocation.

Contribution and Comparison with the Related Works

The most closely related work to ours is [Chen et al., 2018], which also investigates a variant of CCV-MAB. This work assumes that the reward function is submodular and the expected base arm outcomes are Hölder continuous in contexts with exponent $\beta > 0$, and uses a greedy algorithm as the approximation oracle. Their proposed learning algorithm, CCMAB, uses the similarity information in the space of contexts to learn the expected base arm outcomes. For this, it uniformly discretizes the context space $\mathcal{X}$ into hypercubes whose sizes are set according to the time horizon T, resulting in a regret of $\tilde{O}(T^{(2\beta+D)/(3\beta+D)})$. As opposed to that work, ACC-UCB adaptively discretizes the context space to leverage the benignness of base arm arrivals and the structure of the expected reward function. Thereby, the regret bounds proven for ACC-UCB do not directly depend on D and it achieves a strictly smaller regret compared to CC-MAB under Lipschitz continuity ($\beta = 1$). We also provide a recipe for obtaining more optimistic regret bounds while taking into account the volatility of the base arms. In addition, we test ACC-UCB in worker selection for mobile crowdsourcing and show that it achieves significant performance gains compared to the state-of-the-art.

Adaptive discretization was first introduced to address the problem of the *continuum-armed* bandit [Auer et al., 2007, Kleinberg, 2005], where there are infinitely many arms to choose from, and thus, learning their expected rewards independently becomes practically impossible. This problem was generalized in [Bubeck et al., 2011] to generic measurable spaces of arms, which also introduced Hierarchical Optimistic Optimization (HOO) algorithm. Under a set of weak continuity assumptions on the mean reward function around its maxima, HOO was shown to achieve $\tilde{O}(T^{(D_n+1)/(D_n+2)+\epsilon})$ regret, for any $\epsilon > 0$, where D_n is the near optimality dimension related to the arm space. ACC-UCB significantly differs from HOO in the following aspects: it selects multiple arms in each round by approximately solving a combinatorial optimization problem and the set of arms that it can select from changes in every round. This makes both the algorithmic structure and the regret analysis of ACC-UCB significantly different from that of HOO.

A variant of HOO was proposed for the Bayesian setting of Gaussian process MAB in [Shekhar et al., 2018]. Similarly, [Slivkins, 2014] proposed the contextual zooming algorithm for the contextual MAB, and proved regret bounds that depend on the contextual zooming dimension. Another related work [Li et al., 2016] developed algorithms for the contextual combinatorial cascading bandit. A detailed comparison of our work with other related works is given in Table 1.

The rest of the paper is organized as follows: Section 2 formulates CCV-MAB. Section 3 describes ACC-UCB. Section 4 contains the regret analysis. Section 5 gives the experimental results and Section 6 provides a conclusion and possible future lead. Full proofs and a table of notation are given in the supplemental document.

2 PROBLEM FORMULATION

2.1 Base Arms, Outcomes, Super Arms and Rewards

Our setup involves base arms that are defined by their D-dimensional contexts, x, belonging to the context set $\mathcal{X} = [0,1]^D$. A base arm with context x has an outcome denoted by $r(x)$, which is a random variable that takes values in $[0,1]$. We assume that there exists a function $\mu(x) : \mathcal{X} \to [0,1]$ such that $\mu(x) = \mathbb{E}[r(x)]$, $\forall x \in \mathcal{X}$. We assume that base arms with similar contexts have similar expected outcomes. This is captured by imposing the following smoothness condition on μ, which is a standard assumption in the contextual MAB setting [Slivkins, 2014].

Assumption 1. *(Lipschitz continuity for the expected outcome in contexts). For any given pair of contexts $x, x' \in \mathcal{X}$, we have $|\mu(x) - \mu(x')| \leq \|x - x'\|_2$, where $\|\cdot\|_2$ is the Euclidean norm in $\mathbb{R}^D$.*

A super arm is a set of K base arms that is defined by the contexts of its base arms. Consider a super arm associated with the context tuple $\boldsymbol{x} = [x_1, \dots, x_K]$ such that $x_m \in \mathcal{X}$, $\forall m \in [K]$. The corresponding outcome and expected outcome vectors (the latter is also called the expectation vector) are denoted by $r(\boldsymbol{x}) = [r(x_1), \dots, r(x_K)]$ and $\mu(\boldsymbol{x}) = [\mu(x_1), \dots, \mu(x_K)]$. We assume that the reward received from playing this super arm is a non-negative real number that is given by $u(r(\boldsymbol{x}))$ and that the base arm outcomes are independent of each other. We only make the following mild assumptions on the reward, which allow for a very large class of functions to fit our model. We note that these assumptions are standard in the combinatorial MAB setting [Chen et al., 2013].

Assumption 2. *$\forall \boldsymbol{x} = [x_1, \dots, x_K]$ such that $x_m \in \mathcal{X}$, $\forall m \in [K]$, we have $\mathbb{E}[u(r(\boldsymbol{x}))] = u(\mu(\boldsymbol{x}))$.*

Assumption 2 states that the expected reward of playing a super arm is only a function of the expectation vector of that super arm. As noted in [Chen et al., 2013], this assumption holds for linear reward functions, but can also hold for non-linear ones if we know the distribution type of the base arms and the outcomes of different base arms are independent.

Assumption 3. *(Monotonicity) For any $\mu = [\mu_1, \dots, \mu_K] \in [0,1]^K$ and $\mu' = [\mu'_1, \dots, \mu'_K] \in [0,1]^K$ if $\mu_m \leq \mu'_m$, $\forall m \in [K]$, then $u(\mu) \leq u(\mu')$.*

Assumption 3 states that the expected reward is monotonically non-decreasing with respect to the expected outcome vector.

Assumption 4. *(Lipschitz continuity of the expected reward in expected outcomes) $\exists B > 0$ such that for any $\mu = [\mu_1, \dots, \mu_K] \in [0,1]^K$ and $\mu' = [\mu'_1, \dots, \mu'_K] \in [0,1]^K$, we have $|u(\mu) - u(\mu')| \leq B \sum_{i=1}^K |\mu_i - \mu'_i|$.*

Assumption 4 implies that one can get a good estimate of the expected reward of a super arm if one can get good estimates of the expected outcomes of the base arms that are in that super arm.

2.2 The Learning Problem

We consider a sequential decision-making problem with volatile base arms that proceeds over T rounds indexed by $t \in [T]$. The learner knows u perfectly, but does not know μ beforehand. In each round t, $M^t > K$ base arms indexed by the set $\mathcal{M}^t = [M^t]$ arrive.[1] We assume $M^t < \infty$, for all $t > 0$. The context of base arm $m \in \mathcal{M}^t$ is represented by $x^t_m \in \mathcal{X}$. We denote by $\mathcal{X}^t = \{x^t_m\}_{m \in \mathcal{M}^t}$ the set of available contexts and by $\mu^t = [\mu(x^t_m)]_{m \in \mathcal{M}^t}$ the vector of expected outcomes of the available base arms in round t. We denote by $\mathcal{S}^t = \{S \subset \mathcal{M}^t : |S| = K\}$ the set of available super arms in round t.

At the beginning of round t, the learner first observes $\mathcal{M}^t$ and $\mathcal{X}^t$ and then selects a super arm S^t from $\mathcal{S}^t$. At the end of round t, the learner collects the reward $u(r(\boldsymbol{x}^t_{S^t}))$ where $\boldsymbol{x}^t_{S^t} = [x^t_{s^t_1}, \dots, x^t_{s^t_K}]$ is the set of context vectors associated with super arm S^t and $r(\boldsymbol{x}^t_{S^t}) = [r(x^t_{s^t_1}), \dots, r(x^t_{s^t_K})]$ is the outcome vector of super arm S^t. It also observes $r(\boldsymbol{x}^t_{S^t})$ as a part of the semi-bandit feedback. The goal of the learner is to maximize its expected cumulative reward by round T.

Since combinatorial optimization is NP-hard in general [Wolsey and Nemhauser, 2014], finding an optimal super arm S^{*t} that achieves an expected reward $\text{opt}(\mu^t) = \max_{S \in \mathcal{S}^t} u(\mu(\boldsymbol{x}^t_S))$ is computationally intractable even when μ is perfectly known. Thus, we assume that the learner has access to an α-approximation oracle, which when given as input μ^t returns an α optimal solution. Since the learner does not know μ^t in our case, it gives an M^t-dimensional parameter vector θ^t as input to the approximation oracle to get $S^t = \text{Oracle}(\theta^t)$, which is an approximately optimal solution under θ^t but not necessarily under μ^t.

To measure the loss of the learner in this setting by round T for a fixed sequence of context arrivals $\{\mathcal{X}^t\}_{t=1}^T$, we use

[1] We assume that the reward monotonically increases as the set of selected base arms grows. Therefore, when $M^t \leq K$, the learner will select all available base arms and incur zero regret. Here, we focus on the non-trivial case where $M^t > K$ for all t.

the standard notion of α-approximation regret [Chen et al., 2013] (referred to as the regret hereafter), given as

$$R_\alpha(T) = \alpha \sum_{t=1}^{T} \text{opt}(\mu^t) - \sum_{t=1}^{T} u(\mu(\boldsymbol{x}^t_{S^t})) \ .$$

Our goal is to devise a learning algorithm that minimizes the growth rate of the regret of the learner. This is a challenging problem since the expected outcomes of the base arms are unknown a priori and volatility of the base arms poses a challenge in estimating the expected outcomes accurately. Since the learner does not have any control over M^t and $\mathcal{X}^t$, in the worst-case, a different set of arms can arrive in each round. Therefore, achieving sublinear regret in time would be impossible without further structure on the outcomes and rewards. Thus, the assumptions made in the previous section are necessary to integrate what has been learned in the past to form accurate estimates of expected base arm outcomes and expected rewards in the current round. In particular, they allow us to partition the context space into regions, each of which is assumed to contain contexts with similar expected outcomes, and create partition-based estimates of expected base arm outcomes. In the next section, we list the properties of the context space and reward function that follow from the definitions and assumptions in Section 2.1. These will be used in both algorithm design and regret analysis.

2.3 Properties of the Context Space

We first give the definition of a well-behaved metric space.

Definition 1. *(Well-behaved metric space [Bubeck et al., 2011]) A compact metric space $(\mathcal{X},\, d)$ is said to be well-behaved if there exists a sequence of subsets $(\mathcal{X}_h)_{h\geq 0}$ of $\mathcal{X}$ satisfying the following properties:*

1. *Given $N \in \mathbb{N}$, each subset $\mathcal{X}_h$ has N^h elements, i.e. $\mathcal{X}_h = \{x_{h,i},\ 1 \leq i \leq N^h\}$ and to each element $x_{h,i}$ is associated a cell $X_{h,i} = \{x \in \mathcal{X} :\ d(x, x_{h,i}) \leq d(x, x_{h,j}),\ \forall j \neq i\}$.*
2. *For all $h \geq 0$ and $1 \leq i \leq N^h$, we have: $X_{h,i} = \cup_{j=N(i-1)+1}^{Ni} X_{h+1,j}$. The nodes $x_{h+1,j}$ for $N(i-1)+1 \leq j \leq Ni$ are called the children of $x_{h,i}$, which in turn is referred to as the parent.*
3. *We assume that the cells have geometrically decaying radii, i.e. there exists $0 < \rho < 1$ and $0 < v_2 \leq 1 \leq v_1$ such that we have $B(x_{h,i},\, v_2\rho^h/2) \subseteq X_{h,i} \subseteq B(x_{h,i},\, v_1\rho^h/2)$, where $B(x, r)$ denotes a closed ball centered at x with radius r. Note that we have $v_2\rho^h \leq \text{diam}(X_{h,i}) \leq v_1\rho^h$, where $\text{diam}(X_{h,i}) := \sup_{x,y\in X_{h,i}} d(x, y)$.*

The first property implies that for every $h \geq 0$ the cells $X_{h,i}$, $1 \leq i \leq N^h$ partition $\mathcal{X}$. This can be observed trivially by reductio ad absurdum. The second property intuitively means that as h grows, we get a more refined sequence of partitions. The third property implies that the nodes $x_{h,i}$ are evenly spread out in the space. As previously indicated in [Bubeck et al., 2011, Shekhar et al., 2018], $(\mathcal{X}, \|\cdot\|_2)$ is well behaved.

Our regret bounds depend on the notion of approximate optimality dimension, which relates to the dimensions of sets of optimistic contexts that yield approximately optimal expected rewards. First, we define the approximate optimality dimension of the context space, tailored to our combinatorial setting, which is inspired by definitions of the near optimality dimension given in [Bubeck et al., 2011, Shekhar et al., 2018, Munos, 2011].

Definition 2. *(The approximate optimality dimension)*

- *A subset $\mathcal{X}_2$ of $\mathcal{X}$ is called r-separated if for any $x_1, x_2 \in \mathcal{X}_2$ such that $x_1 \neq x_2$, we have $\|x_1 - x_2\|_2 \geq r$. The cardinality of the largest such set is called the* ***r-packing*** *number of $\mathcal{X}$ with respect to $\|\cdot\|_2$, and is denoted by $M(\mathcal{X}, \|\cdot\|_2, r)$. Equivalently, the r-packing number of $\mathcal{X}$ is the maximum number of disjoint $\|\cdot\|_2$-balls of radius r that are contained in $\mathcal{X}$.*
- *Let $\mathcal{Z} = \mathcal{X}^K$. For any $\kappa > 0$, $r > 0$, $t > 0$ and $f : \mathbb{R}^+ \to \mathbb{R}^+$, we define the set*

$$\mathcal{X}^\kappa_{f(r)} := \{x \in \mathcal{X} : \kappa - u(\mu(\mathbf{x})) \leq f(r)\, , \text{for some } \mathbf{x} \in \mathcal{Z} \text{ such that } x \in \mathbf{x}\}$$

*to be an $(f(r), u, \kappa)$-**optimal** set. Let $M(\mathcal{X}^\kappa_{f(r)}, \|\cdot\|_2\, , r)$ be its r-packing number. We define the (f, u, κ)-**optimality dimension** $D^u(f, \kappa)$ associated with $\mathcal{X}^\kappa_{f(r)}$ and u as follows:*

$$D^u(f, \kappa) = \max\left\{0, \limsup_{r\to 0} \frac{\log(M(\mathcal{X}^\kappa_{f(r)}\, \|\cdot\|_2\, , r))}{\log(r^{-1})}\right\}$$

Note that if $\boldsymbol{x} \in \mathcal{Z}$ is such that $\kappa - u(\mu(\mathbf{x})) \leq f(r)$, then all elements of $\boldsymbol{x}$ will be in $\mathcal{X}^\kappa_{f(r)}$. Use of the κ-optimality dimension allows us to bound the regret in the volatile setting in a way that the time order of the regret depends on $D^u(f, \kappa)$ which is in most of the cases strictly smaller than D as opposed to the prior work [Chen et al., 2018] which has bounds that depend on D. For instance, we can let $u^*_{\min} = \min_{t\in[T]} u(\mu(\boldsymbol{x}^t_{S^{*t}}))$ and $\kappa = \alpha u^*_{\min}$ to obtain a worst-case approximate optimality dimension $\bar{D} = D^u(f, \alpha u^*_{\min})$.

Remark 1. *$\bar{D}$ is obviously less than or equal to D since we are narrowing the space to specific subspaces. We illustrate this in the example below, which is a modified version of Example 3 in [Bubeck et al., 2011]. We thus exploit the nature of the reward function u. If we remove the volatility assumption from the problem setting and assume that* *the learner can choose only one arm in a given round, $\bar{D}$ will be the near optimality dimension of the context space (where we let κ be the optimal expected reward), thus recovering the notion as introduced in previous works. In Section 4.4 we explain ways to construct more optimistic regret bounds using the approximate optimality dimension. Moreover, a case for which $\bar{D} < D$ when $\alpha < 1$ is given in Section 4.3.*

Example 1. *Let $\mathcal{X} = [0,1]^D$ and $K = 1$. Let us define $u(\mu(x)) = 1 - \|x\|_2^a$ for some $a > 1$. Let $\|\cdot\|_2$ be the norm defined on the metric space $\mathcal{X}$. Fix $c > 0$, let $f(r) = cr$ and $\kappa = u^*_{\min}$ (exact oracle). Let $\tilde{t} = \text{argmin}_{t \in [T]} u(\mu(x^{*t}))$, where $x^{*t} := \text{argmax}_{x \in \mathcal{X}^t} u(\mu(x))$. Denote $x^* = x^{*\tilde{t}}$ and note that in this case $\kappa = u(\mu(x^*))$. When $\|x^*\|_2 = 0$, we have $\bar{D} \le (1 - 1/a)D < D$ (proof is in the supplemental document).*

3 THE LEARNING ALGORITHM

Our algorithm is called *Adaptive Contextual Combinatorial Upper Confidence Bound* (ACC-UCB) and is motivated by several tree based methods that have been used for function optimization under continuity assumptions [Bubeck et al., 2011, Shekhar et al., 2018, Munos, 2011] (pseudocode given in Algorithm 1). By Definition 1 there exists a sequence $(\mathcal{X}_h)_{h \ge 0}$, each containing N^h nodes whose associated cells form a tree of partitions of $\mathcal{X}$. The procedure is described as follows: At round t, we observe arrived base arms and contexts. We maintain an active set of leaf nodes and denote it by $\mathcal{L}^t$. For the arrived base arms, we identify the set of available active leaf nodes, whose regions contain the available contexts and denote it by $\mathcal{N}^t$. By $p(x_{h,i})$ we denote the parent of node $x_{h,i}$. For each active leaf node we maintain an index which is an upper confidence bound on the maximum expected outcome of base arms which have contexts in the region associated with the node. The index is defined as $g^t(x_{h,i}) := b^t(x_{h,i}) + v_1 \rho^h$ where the term $b^t(x_{h,i})$ is a high probability upper bound on $\mu(x_{h,i})$ defined as

$$b^t(x_{h,i}) := \min\{\hat{\mu}^{t-1}(x_{h,i}) + c^{t-1}(x_{h,i}),\\ \hat{\mu}^{t-1}(p(x_{h,i})) + c^{t-1}(p(x_{h,i})) + v_1 \rho^{(h-1)}\}$$

and $c^t(x_{h,i}) := \sqrt{2 \log T / C^t(x_{h,i})}$ is the confidence radius, tailored to give high probability upper bounds on μ. Here $C^t(x_{h,i})$ is the number of times a base arm associated with a context from the cell $X_{h,i}$ was selected by the algorithm, formally defined as $C^t(x_{h,i}) := \sum_{t'=1}^{t} \sum_{k=1}^{K} \mathbb{I}\{(H_k^{t'}, I_k^{t'}) = (h,i)\}$ where we denote by $(H_k^{t'}, I_k^{t'})$ the active leaf node associated with the cell containing the context of the kth selected base arm at time t'. We define the total reward accumulated by the algorithm until round t from selecting arms with contexts associated with the node $x_{h,i}$ as follows:

$$v^t(x_{h,i}) := \sum_{t'=1}^{t} \sum_{k=1}^{K} r(x^{t'}_{s^{t'}_k}) \mathbb{I}\{(H_k^{t'}, I_k^{t'}) = (h,i)\}$$

where we denote by s_k^t the kth base arm selected by the algorithm at round t. Consequently, we define the empirical mean used in the index as

$$\hat{\mu}^t(x_{h,i}) := \begin{cases} v^t(x_{h,i})/C^t(x_{h,i}) & \text{for } C^t(x_{h,i}) > 0 \\ 0 & \text{otherwise} \end{cases}$$

$C^t(x_{h,i})$ can be larger than t, since at a certain round, the algorithm may select base arms with contexts belonging to the same cell. However, it always holds that $C^t(x_{h,i}) \le Kt$. The constants v_1 and ρ are parameters as described in Definition 1 that are given as input to the algorithm.

Using the same approach, we next define the index of an arm $m \in \mathcal{M}^t$. Let $\phi_m^t = x_{\tilde{h}_m^t, \tilde{i}_m^t}$ be the active leaf node associated to the cell containing x_m^t. Then,

$$g^t(x_m^t) := g^t(\phi_m^t) + N(v_1/v_2) v_1 \rho^{\tilde{h}_m^t}$$

where the second term guarantees (with high probability) that $g^t(x_m^t)$ upper bounds $\mu(x_m^t)$.

Remark 2. *Since at any round t, a cell associated with an active leaf node $x_{h,i}$ may contain several contexts of the available base arms, we have $g^t(x_m^t) = g^t(x_n^t)$ when m and n are two available base arms, both of which have contexts that live in the cell $X_{h,i}$. As a consequence, indices of all base arm contexts inside one cell are equal.*

After the indices of the available base arms, i.e., $\{g^t(x_m^t)\}_{m \in \mathcal{M}^t}$ are computed, they are given as input θ^t to the approximation oracle in round t to obtain the super arm $S^t \subset \mathcal{M}^t$ that will be played in round t.[2] At this point, we identify the active leaf nodes that are "selected", denote their collection by $\mathcal{P}^t$, and update their statistics (after these are played and their outcomes are observed) according to the following rules. For each $x_{h,i} \in \mathcal{P}^t$:

$$\hat{\mu}^t(x_{h,i}) = \frac{C^{t-1}(x_{h,i}) \hat{\mu}^{t-1}(x_{h,i}) + rew^t(x_{h,i})}{C^{t-1}(x_{h,i}) + num^t(x_{h,i})} \quad (1)$$

$$C^t(x_{h,i}) = C^{t-1}(x_{h,i}) + num^t(x_{h,i}) \quad (2)$$

where $rew^t(x_{h,i}) = \sum_{k=1}^{K} r(x^t_{s^t_k}) \mathbb{I}\{(H_k^t, I_k^t) = (h,i)\}$ and $num^t(x_{h,i}) = \sum_{k=1}^{K} \mathbb{I}\{(H_k^t, I_k^t) = (h,i)\}$. Statistics of the other active leaf nodes do not change. Subsequently, for each node $x_{h,i} \in \mathcal{P}^t$, we decide whether or not to expand it into N children nodes, according to the following condition:

[2] The base arms are randomly chosen in the first round.

- *Refine*. If $c^t(x_{h,i}) \leq v_1\rho^h$, then the node $x_{h,i}$ is expanded into N children nodes $\{x_{h+1,j} : N(i-1)+1 \leq j \leq Ni\}$ which are added to the set of active leaves, whereas $x_{h,i}$ is removed from it.

If the above condition is not satisfied, we continue. Basically, we refine the partitions when we are confident enough about the μ values inside that cell.

Algorithm 1 ACC-UCB

```
Input: $\mathcal{X}$, $(\mathcal{X}_h)_{h\geq 0}$, $v_1$, $v_2$, $\rho$, $N$, $x_{0,1}$, $K$, $T$.
  Initialize: $C^0(x_{h,i}) = 0$, $\hat{\mu}^0(x_{h,i}) = 0$, $\forall x_{h,i} \in \mathcal{X}$; $\mathcal{X}_0 = \mathcal{X}$, $\mathcal{L}^1 = \{x_{0,1}\}$.
  for $t = 1, 2, ..., T$ do:
      Observe base arms in $\mathcal{M}^t$ and their contexts $\mathcal{X}^t$.
      Identify available active leaf nodes $\mathcal{N}^t \subseteq \mathcal{L}^t$.
      Compute indices $g^t(x_{h,i})$, $x_{h,i} \in \mathcal{N}^t$.
      Compute indices $g^t(x_m^t)$, $m \in \mathcal{M}^t$.
      $S^t \leftarrow$ Oracle$(g^t(x_1^t), \ldots, g^t(x_{M^t}^t))$.
      Observe outcomes of base arms in $S^t$ and collect the reward.
      Identify the set of selected nodes $\mathcal{P}^t$.
      for $x_{h,i} \in \mathcal{P}^t$ do:
          Update $\hat{\mu}^t(x_{h,i})$ as in (1) and $C^t(x_{h,i})$ as in (2).
          if $c^t(x_{h,i}) \leq v_1\rho^h$ then:
              $\mathcal{L}^{t+1} \leftarrow \mathcal{L}^t \cup \{x_{h+1,j} : N(i-1)+1 \leq j \leq Ni\} \setminus \{x_{h,i}\}$.   ▷ Refine the tree of partitions.
          end if
      end for
  end for
```

4 REGRET ANALYSIS

4.1 Preliminaries

First, we show that the index of a given node $x_{h,i}$ is an upper bound of its true mean with overwhelming probability.

Lemma 1. *Given the event $\mathcal{F} = \big\{\forall t \leq T, \forall x_{h,i} \in \mathcal{L}^t : |\hat{\mu}^t(x_{h,i}) - \mu(x_{h,i})| \leq c^t(x_{h,i}) + v_1\rho^h\big\}$ under the assumptions made in Section 2, we have $\mathbb{P}\big\{\mathcal{F}\big\} \geq 1 - 2K^2T^{-1}$.*

The next result gives high probability bounds of the difference between the index and the mean of a given node. Furthermore, we give an upper bound on the number of times a node can be selected before expansion.

Lemma 2. *Consider that event $\mathcal{F}$ happens. Then, if at round t, the node $x_{h,i} \in \mathcal{P}^t$ is not expanded by the algorithm, we have $g^t(x_{h,i}) - \mu(x_{h,i}) \leq (5Nv_1/v_2 + 1)v_1\rho^h$. Moreover, a node $x_{h,i}$ may be selected by the algorithm no more than q_h times before it is expanded, where $q_h \leq \left\lceil \frac{2\log T}{(v_1\rho^h)^2} \right\rceil \leq \frac{2\log(Tv_3)}{(v_1\rho^h)^2}$ with $v_3 := \sqrt{e^{v_1^2}}$.*

Next, we give a high probability upper bound on the true mean of any given arm.

Lemma 3. *Consider that event $\mathcal{F}$ happens. Then, we have $\forall t \leq T$ and $m \in \mathcal{M}^t$, $g^t(x_m^t) \geq \mu(x_m^t)$.*

The following lemma upper bounds the suboptimality gap of any suboptimal super arm with high probability.

Lemma 4. *If the reward function u satisfies the Lipschitz continuity and the monotonicity condition and if event $\mathcal{F}$ holds, then in any round t, the regret incurred by super arm S^t is upper bounded by $\sum_{m\in S^t} B(6Nv_1/v_2 + 2)v_1\rho^{\tilde{h}_m^t}$ or by $BK(6Nv_1/v_2+2)v_1\rho^{h(t)}$, where $h(t) = \min\{\tilde{h}_m^t : m \in S^t\}$.*

Finally, we will make use of the following fact when upper bounding the cardinality of the set of nodes from which the algorithm selects.

Lemma 5. *Fix $\kappa > 0$. Let $\bar{D} = D^u(f, \kappa)$ and $f(r) = cr$ for a given $c > 0$. Fix $D_1 > \bar{D}$. Then, there exists a constant Q, such that for all $r \leq v_2$ we have $M(\mathcal{X}_{cr}^{\kappa}, \|\cdot\|_2, r) \leq Qr^{-D_1}$.*

4.2 A Sublinear Regret Bound

Theorem 1. *Fix $T > 0$. Given the parameters of the problem $0 < \alpha \leq 1$, $N \in \mathbb{N}$, $K \in \mathbb{N}$, $B > 0$ and $0 < v_2 \leq 1 \leq v_1$, define $\bar{D} = D^u(f, \alpha u_{\min}^*)$ and $f(r) = cr$, where $c = BK(6Nv_1/v_2+2)v_1/v_2$. Then, for any $D_1 > \bar{D}$, there exists $Q = Q(\mathcal{X}, u, \mu, \alpha u_{\min}^*, c) > 0$ (independent of T), for which the α-regret incurred by ACC-UCB is upper bounded with probability at least $1 - 2K^2T^{-1}$ as follows:*

$$R_\alpha(T) \leq C_1 \cdot T^{1-\frac{1}{D_1+2}} \cdot (\log(Tv_3))^{\frac{1}{D_1+2}} \\ + C_2 \cdot T^{1-\frac{1}{D_1+2}} \cdot (\log(Tv_3))^{\frac{1}{D_1+2}}$$

where $C_1 := 2QBK(6Nv_1/v_2+2)\frac{{v_2}^{-D_1}}{v_1(\rho^{-1}-1)}$ and $C_2 := KB(6Nv_1/v_2+2)v_1$.

Sketch of proof. By Lemma 4 we have $R_\alpha(T) \leq \sum_{t\leq T} BK(6Nv_1/v_2+2)v_1\rho^{h(t)}$. In order to obtain sublinear regret, we express the summation in terms of the levels of the tree and then we fix some level H and consider two summations separately, that correspond to the levels $h < H$ and $h \geq H$. Then, we use the fact that any two nodes in $\mathcal{X}_h$ are at least $v_2\rho^h$ apart, the definition of $v_2\rho^h$-packing number and Lemma 5 to bound the cardinality of the set of nodes from which the algorithm may select base arms. After merging the two summations, we design the prefixed constant H so that we can have a sublinear bound. Finally, by a clever choice of H, we obtain $\tilde{O}(T^{(\bar{D}+1)/(\bar{D}+2)+\epsilon})$ regret for any $\epsilon > 0$. ■

In the previous work closest to our setting [Chen et al., 2018], the algorithm CC-MAB is shown to achieve $\tilde{O}(T^{(2\beta+D)/(3\beta+D)})$, regret, where β is a positive real number ($\beta = 1$ gives our Assumption 1) and D is the dimension of the context space. Their setting assumes a submodular reward function, whereas we, except from Assumptions 2-4, do not assume anything else on the nature of this function. While they uniformly partition the context space, we adaptively partition it, making full use of benign contexts. The difference is that after a certain point, the learning accuracy of CC-MAB stops growing, while that of ACC-UCB converges to an optimum. As a result, ACC-UCB learns much faster than CC-MAB (see Section 5 for numerical results). While their regret bounds depend on the dimension of the context space D, our bounds depend on the approximate optimality dimension $\bar{D}$, which is less than or equal to D. Moreover, for $\beta = 1$ the exponent of T in their bounds is $(D+2)/(D+3)$, while in our bounds it is $(D_1+1)/(D_1+2)$, for any $D_1 > \bar{D}$. As a conclusion, ACC-UCB outperforms the current state-of-the-art in the contextual combinatorial volatile setting.

Finally, we note that if each super arm is a base arm, the set of available base arms is time-invariant, $u(\mu(x)) = \mu(x)$ and $\alpha = 1$, then our problem becomes a special case of the $\mathcal{X}$-armed bandit [Bubeck et al., 2011]. In this case, $\bar{D}$ becomes the near optimality dimension in [Bubeck et al., 2011] and time order of the upper bound in Theorem 1 matches with the regret lower bound in Theorem 13 of [Bubeck et al., 2011] (which holds for metric spaces) up to a multiplicative logarithmic factor.

4.3 Regret When the Set of Contexts is Finite

When the context space is of finite cardinality (even though of a combinatorial nature, and thus potentially very large), the r-packing number of (f, u, κ)-optimal sets will stop growing after some point, and thus we can upper bound it. Let $P(f, u, \kappa) := \limsup_{r \to 0} M(\mathcal{X}^{\kappa}_{f(r)}, \|\cdot\|_2, r)$. Then, the regret incurred by ACC-UCB is upper bounded as shown below.

Theorem 2. *Under the assumptions of Theorem 1 and the additional assumption that the context space is of finite cardinality, for any $\epsilon > 0$, the α-regret incurred by ACC-UCB in T rounds is upper bounded with probability at least $1 - 2K^2T^{-1}$ as follows:*

$$R_\alpha(T) \le C_3 T^{1-\frac{1}{\epsilon+2}} \cdot (\log(Tv_3))^{\frac{1}{\epsilon+2}} + C_4 \cdot \left[T^{1-\frac{1}{\epsilon+2}} \cdot (\log(Tv_3))^{\frac{1}{\epsilon+2}}\right]$$

where $C_3(\epsilon) := BK(6Nv_1/v_2 + 2)\frac{2v_2^{-\epsilon}P(f,u,\alpha u^*_{\min})}{v_1(1/\rho-1)}$ and $C_4 := BK(6Nv_1/v_2+2)v_1$.

4.4 Optimistic Regret Bounds

The regret bound in Theorem 1 depends on the worst super arm among the optimal super arms over all the rounds. While this being the worst among the best is reasonable, can we obtain a tighter bound? For this, we partition the set of the rounds $[T]$ into subsets and consider their contribution to the regret separately, in order to obtain different dimensions not depending on the worst of the best super arms overall, but the worst of the best super arms over smaller sets.

Formally, let $\pi : [T] \to [T]$ be a permutation of the rounds such that $u(\mu(\boldsymbol{x}^{\pi(1)}_{S^{*\pi(1)}})) \ge \ldots \ge u(\mu(\boldsymbol{x}^{\pi(T)}_{S^{*\pi(T)}}))$ and let us denote by $\pi([T])$ the new ordered set. Now let $\Gamma = \{\mathcal{T}^\Gamma_1, \ldots, \mathcal{T}^\Gamma_{|\Gamma|}\}$ be an ordered partition of $\pi([T])$ and let us denote by $T^\Gamma_\lambda = |\mathcal{T}^\Gamma_\lambda|$ the cardinality of $\mathcal{T}^\Gamma_\lambda$, for some $\lambda \in [|\Gamma|]$ (there are finitely many such partitions). Also, for any $0 < \lambda \le |[\Gamma]|$, denote the expected reward of the "worst" optimal super arm in $\mathcal{T}^\Gamma_\lambda$ as $u^*_{\min}(\Gamma, \lambda) = \min_{t \in \mathcal{T}^\Gamma_\lambda} u(\mu(\boldsymbol{x}^t_{S^{*t}}))$. Let $\mathcal{P}(\pi([T]))$ be the set of all partitions of $\pi([T]))$. Now let us denote by $\bar{D}^\Gamma_\lambda$ the $(f, u, \alpha u^*_{\min}(\Gamma, \lambda))$-optimality dimension[3] associated with the subset $\mathcal{T}^\Gamma_\lambda$ and by $R_{\alpha,\lambda}(\mathcal{T}^\Gamma_\lambda)$ the regret incurred by the learner over the rounds in $\mathcal{T}^\Gamma_\lambda$.

Theorem 3. *Under the assumptions of Theorem 1, for any $\Gamma \in \mathcal{P}(\pi([T]))$ let $\{D^\Gamma_\lambda\}_{\lambda \le |\Gamma|}$ be a sequence of constants such that $D^\Gamma_\lambda > \bar{D}^\Gamma_\lambda$, for $\lambda \le |\Gamma|$. Then, there exists $Q^\Gamma_\lambda = Q^\Gamma_\lambda(\mathcal{X}, u, \mu, \alpha u^*_{\min}(\Gamma, \lambda), c) > 0$, $\lambda \le |\Gamma|$ such that the α-regret incurred by ACC-UCB is upper bounded with probability at least $1 - 2K^2T^{-1}$ as follows:*

$$R_\alpha(T) \le \sum_{\lambda=1}^{|\Gamma|} \left[C_5(\Gamma,\lambda) \cdot (T^\Gamma_\lambda)^{1-\frac{1}{D^\Gamma_\lambda+2}} \cdot (\log(Tv_3))^{\frac{1}{D^\Gamma_\lambda+2}} + C_6 \cdot (T^\Gamma_\lambda)^{1-\frac{1}{D^\Gamma_\lambda+2}} \cdot (\log(Tv_3))^{\frac{1}{D^\Gamma_\lambda+2}}\right]$$

where $C_5(\Gamma, \lambda) := 2Q^\Gamma_\lambda KB(6Nv_1/v_2 + 2)\frac{{v_2}^{-D^\Gamma_\lambda}}{v_1(1/\rho-1)}$ and $C_6 := KB(6Nv_1/v_2+2)v_1$.

For $\xi \in (0,1)$, let $\mathcal{T}^\xi = \{\pi(1), \ldots, \pi(T - \lfloor T^\xi \rfloor)\}$, $u^*_{\min}(\xi) = \min_{t \in \mathcal{T}^\xi} u(\mu(\boldsymbol{x}^t_{S^{*t}}))$ and $u^*_{\max} = \max_{t \in [T]} u(\mu(\boldsymbol{x}^t_{S^{*t}}))$. The following corollary gives another bound on $R_\alpha(T)$ in terms of a specific partition of $\pi[T]$ defined by $\mathcal{T}^\xi$.

Corollary 1. *For $\xi \in (0,1)$ let $\bar{D}(\xi)$ be the $(f, u, \alpha u^*_{\min}(\xi))$-optimality dimension associated with the set $\mathcal{T}^\xi$. Let $D : (0,1) \to \mathbb{R}_+$ be any function such that $D(\xi) > \bar{D}(\xi)$, for all $\xi \in (0,1)$. Then, there exists $Q(\xi) = Q(\mathcal{X}, u, \mu, \alpha u^*_{\min}(\xi), c) > 0$ such that the α-regret incurred by ACC-UCB in T rounds is upper bounded with probability at least $1 - 2K^2T^{-1}$ as follows:*

$$R_\alpha(T) \le \inf_{\xi \in (0,1)} \left(C_5(\xi) \cdot T^{1-\frac{1}{D(\xi)+2}} \cdot (\log(Tv_3))^{\frac{1}{D(\xi)+2}} + C_6 \cdot T^{1-\frac{1}{D(\xi)+2}} \cdot (\log(Tv_3))^{\frac{1}{D(\xi)+2}} + \alpha u^*_{\max} T^\xi\right)$$

[3] f is the same as in Theorem 1.

where $C_5(\xi) := 2Q(\xi)KB(6Nv_1/v_2 + 2)\frac{{v_2}^{-D(\xi)}}{v_1(1/\rho - 1)}$ *and* C_6 *is defined as in Theorem 3.*

$\bar{D}(\xi)$ in Corollary 1 is non-increasing in ξ. The optimal value for ξ for which the time order of the terms in the regret bound are balanced is such that $T^{\xi} = T^{1-\frac{1}{D(\xi)+2}}$.

5 EXPERIMENTS

We consider a mobile crowdsourcing problem where the goal is to assign a subset of available workers to location dependent tasks arriving sequentially over time. Formally, in each round t, a task arrives with a location (normalized longitude and latitude) sampled uniformly at random from $[0, 1]^2$. Then, the learner selects $K \in \{2, 4\}$ workers from the set of available workers, $\mathcal{M}^t$, which is sampled from the Poisson distribution with mean 50. A worker is characterized by its location that lies in $[0, 1]^2$ and its energy and willingness to work (i.e. battery status) sampled uniformly at random from $[0, 1]$. We generate workers using the Gowalla dataset [Cho et al., 2011]. This dataset consists of 6,442,892 user check-ins in the social networking platform Gowalla. Each check-in contains the user id, time of check-in and location of check-in. In each round t, we randomly select M^t of these check-ins without replacement and normalize and assign each of their locations to a worker. Each base arm m in round t is a task-worker pair with a two-dimensional context $x^t_m = (x^t_{m,1}, x^t_{m,2})$. Here, $x^t_{m,1}$ represents the normalized[4] Euclidean distance between the worker and task locations, while $x^t_{m,2}$ represents the battery status of the worker.

We define the expected base arm outcome as $\mu(x^t_m) = f(x^t_{m,1}) \cdot (x^t_{m,2})^2$ where f is a Gaussian probability density function with mean 0 and standard deviation 1. Note that μ is decreasing in the distance between the worker and task and increasing in the worker's battery. Furthermore, the outcome $r(x^t_m)$ of worker m in round t is Bernoulli distributed with probability $\mu(x^t_m)$; it is 1 when the worker successfully completes the task and 0 otherwise. We assume that the task is successfully completed if at least one of the assigned workers completes the task, which is true for tasks such as cryptocurrency mining [Mukhopadhyay et al., 2016]. Hence, $u(r(\boldsymbol{x}^t_{S_t})) = 1$ if $\sum_{m \in S^t} r(x^t_m) \geq 1$ and 0 otherwise.

We implemented the simulations in Python[5] and ran them for 50,000 rounds using ACC-UCB, CC-MAB [Chen et al., 2018] and random selections. For ACC-UCB, we set $v_1 = \sqrt{5}, v_2 = 1, \rho = 0.5$, and $N = 2$. Also, $x_{0,1}$ is a square with edge length 1 and center $(0.5, 0.5)$. For CC-MAB we set $\alpha = 1$, $h_T = \left\lceil 50000^{\frac{1}{5}} \right\rceil$. We also used an exact oracle in both algorithms. Reported results correspond to averages over 10 independent runs. Figure 1 shows the cumulative regret and Figure 2 shows the average task reward up to t for all algorithms. As can be seen, ACC-UCB outperforms CC-MAB and random selections.

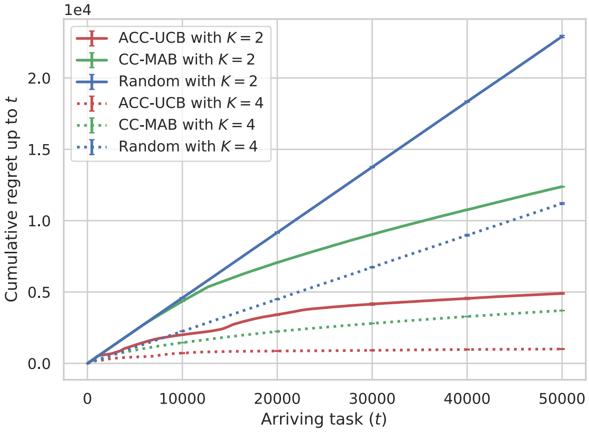

Figure 1: Cumulative regrets of algorithms.

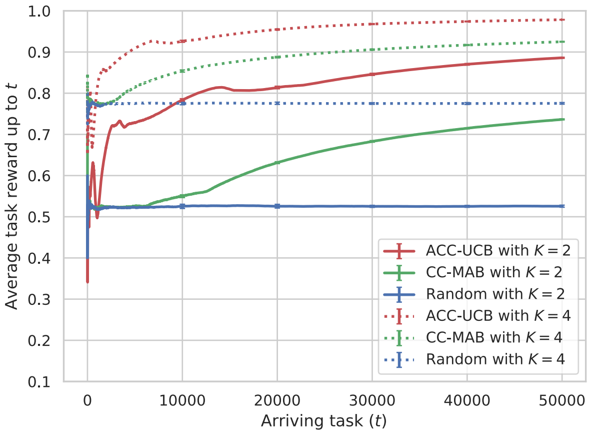

Figure 2: Average task reward up to round t.

6 CONCLUSION

We considered the contextual combinatorial volatile MAB with semi-bandit feedback. We proposed an algorithm, called ACC-UCB, that tradeoffs exploration and exploitation by performing adaptive discretization of the context space under mild continuity assumptions on the expected base arm outcomes and the expected reward. ACC-UCB is proven to achieve $\tilde{O}(T^{(\bar{D}+1)/(\bar{D}+2)+\epsilon})$ regret for any $\epsilon > 0$, where $\bar{D}$ represents the approximate optimality dimension associated with the context space. An interesting future research direction is to investigate CCV-MAB in the Bayesian setting.

[4] To normalize the distance we simply divide it by the maximum possible distance, $\sqrt{2}$.

[5] Full code is provided at https://github.com/Bilkent-CYBORG/ACC-UCB

Acknowledgments

This work was supported by the Scientific and Technological Research Council of Turkey (TUBITAK) under Grant 215E342 and by BAGEP Award of the Science Academy.

References

[Agrawal, 1995] Agrawal, R. (1995). Sample mean based index policies by o (log n) regret for the multi-armed bandit problem. *Adv. Appl. Probability*, 27(4):1054–1078.

[Agrawal and Goyal, 2012] Agrawal, S. and Goyal, N. (2012). Analysis of Thompson sampling for the multi-armed bandit problem. In *Proc. 25th Annu. Conf. Learn. Theory*, pages 39.1–39.26.

[Auer et al., 2002] Auer, P., Cesa-Bianchi, N., and Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Mach. Learn.*, 47(2-3):235–256.

[Auer et al., 2007] Auer, P., Ortner, R., and Szepesvári, C. (2007). Improved rates for the stochastic continuum-armed bandit problem. In *Proc. 20th Annu. Conf. Learn. Theory*, pages 454–468.

[Bnaya et al., 2013] Bnaya, Z., Puzis, R., Stern, R., and Felner, A. (2013). Volatile multi-armed bandits for guaranteed targeted social crawling. In *Workshops at the Twenty-Seventh AAAI Conf. Artif. Intell.*

[Bubeck et al., 2011] Bubeck, S., Munos, R., Stoltz, G., and Szepesvári, C. (2011). X-armed bandits. *J. Mach. Learn. Res.*, 12(May):1655–1695.

[Cesa-Bianchi and Lugosi, 2012] Cesa-Bianchi, N. and Lugosi, G. (2012). Combinatorial bandits. *J. Comput. Syst. Sci.*, 78(5):1404–1422.

[Chakrabarti et al., 2009] Chakrabarti, D., Kumar, R., Radlinski, F., and Upfal, E. (2009). Mortal multi-armed bandits. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 273–280.

[Chen et al., 2018] Chen, L., Xu, J., and Lu, Z. (2018). Contextual combinatorial multi-armed bandits with volatile arms and submodular reward. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 3247–3256.

[Chen et al., 2013] Chen, W., Wang, Y., and Yuan, Y. (2013). Combinatorial multi-armed bandit: General framework and applications. In *Proc. 30th Int. Conf. Mach. Learn.*, pages 151–159.

[Chen et al., 2016] Chen, W., Wang, Y., Yuan, Y., and Wang, Q. (2016). Combinatorial multi-armed bandit and its extension to probabilistically triggered arms. *J. Mach. Learn. Res.*, 17(1):1746–1778.

[Cho et al., 2011] Cho, E., Myers, S. A., and Leskovec, J. (2011). Friendship and mobility: User movement in location-based social networks. In *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery and Data Mining*, pages 1082–1090.

[Chu et al., 2011] Chu, W., Li, L., Reyzin, L., and Schapire, R. (2011). Contextual bandits with linear payoff functions. In *Proc. 14th Int. Conf. Artif. Intell. and Statist.*, pages 208–214.

[Doob, 1953] Doob, J. L. (1953). *Stochastic processes*, volume 101. New York Wiley.

[Gai et al., 2012] Gai, Y., Krishnamachari, B., and Jain, R. (2012). Combinatorial network optimization with unknown variables: Multi-armed bandits with linear rewards and individual observations. *IEEE/ACM Trans. Netw.*, 20(5):1466–1478.

[Huyuk and Tekin, 2019] Huyuk, A. and Tekin, C. (2019). Analysis of thompson sampling for combinatorial multi-armed bandit with probabilistically triggered arms. In *Proc. 22nd Int. Conf. Artif. Intell. and Statist.*, pages 1322–1330.

[Jain et al., 2017] Jain, A., Sarma, A. D., Parameswaran, A., and Widom, J. (2017). Understanding workers, developing effective tasks, and enhancing marketplace dynamics: a study of a large crowdsourcing marketplace. *Proceedings of the VLDB Endowment*, 10(7):829–840.

[Kleinberg et al., 2010] Kleinberg, R., Niculescu-Mizil, A., and Sharma, Y. (2010). Regret bounds for sleeping experts and bandits. *Mach. Learn.*, 80(2-3):245–272.

[Kleinberg, 2005] Kleinberg, R. D. (2005). Nearly tight bounds for the continuum-armed bandit problem. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 697–704.

[Kveton et al., 2015a] Kveton, B., Szepesvari, C., Wen, Z., and Ashkan, A. (2015a). Cascading bandits: Learning to rank in the cascade model. In *Proc. 32nd Int. Conf. Mach. Learn.*, pages 767–776.

[Kveton et al., 2015b] Kveton, B., Wen, Z., Ashkan, A., and Szepesvari, C. (2015b). Tight regret bounds for stochastic combinatorial semi-bandits. In *Proc. 18th Int. Conf. Artif. Intell. and Statist.*, pages 535–543.

[Lai and Robbins, 1985] Lai, T. L. and Robbins, H. (1985). Asymptotically efficient adaptive allocation rules. *Adv. Appl. Math.*, 6(1):4–22.

[Langford and Zhang, 2007] Langford, J. and Zhang, T. (2007). The epoch-greedy algorithm for contextual multi-armed bandits. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 817–824.

[Li et al., 2010] Li, L., Chu, W., Langford, J., and Schapire, R. E. (2010). A contextual-bandit approach to personalized news article recommendation. In *Proc. 19th Int. Conf. World Wide Web*, pages 661–670.

[Li et al., 2016] Li, S., Wang, B., Zhang, S., and Chen, W. (2016). Contextual combinatorial cascading bandits. In *Proc. 33rd Int. Conf. Mach. Learn.*, volume 16, pages 1245–1253.

[Lu et al., 2010] Lu, T., Pál, D., and Pál, M. (2010). Contextual multi-armed bandits. In *Proc. 13th Int. Conf. Artif. Intell. and Statist.*, pages 485–492.

[Mukhopadhyay et al., 2016] Mukhopadhyay, U., Skjellum, A., Hambolu, O., Oakley, J., Yu, L., and Brooks, R. (2016). A brief survey of cryptocurrency systems. In *Proc. Int. Conf. Privacy, Security and Trust*, pages 745–752.

[Munos, 2011] Munos, R. (2011). Optimistic optimization of a deterministic function without the knowledge of its smoothness. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 783–791.

[Radlinski et al., 2008] Radlinski, F., Kleinberg, R., and Joachims, T. (2008). Learning diverse rankings with multi-armed bandits. In *Proc. 25th Int. Conf. Mach. Learn.*, pages 784–791.

[Russo and Van Roy, 2014] Russo, D. and Van Roy, B. (2014). Learning to optimize via posterior sampling. *Math. Oper. Res.*, 39(4):1221–1243.

[Shekhar et al., 2018] Shekhar, S., Javidi, T., et al. (2018). Gaussian process bandits with adaptive discretization. *Electron. J. Stat.*, 12(2):3829–3874.

[Slivkins, 2014] Slivkins, A. (2014). Contextual bandits with similarity information. *J. Mach. Learn. Res.*, 15(1):2533–2568.

[Tewari and Murphy, 2017] Tewari, A. and Murphy, S. A. (2017). From ads to interventions: Contextual bandits in mobile health. In *Mobile Health*, pages 495–517. Springer International Publishing.

[Thompson, 1933] Thompson, W. R. (1933). On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294.

[Wolsey and Nemhauser, 2014] Wolsey, L. A. and Nemhauser, G. L. (2014). *Integer and combinatorial optimization*. John Wiley & Sons.

[Yang et al., 2017] Yang, P., Zhang, N., Zhang, S., Yang, K., Yu, L., and Shen, X. (2017). Identifying the most valuable workers in fog-assisted spatial crowdsourcing. *IEEE Internet Things J.*, 4(5):1193–1203.