Thresholding Bandits with Augmented UCB

Author names withheld

Abstract

In this paper we propose the Augmented-UCB (AugUCB) algorithm for a fixed-budget version of the thresholding bandit problem (TBP), where the objective is to identify a set of arms whose quality is above a threshold. A key feature of AugUCB is that it uses both mean and variance estimates to eliminate arms that have been sufficiently explored; to the best of our knowledge this is the first algorithm to employ such an approach for the considered TBP. Theoretically, we obtain an upper bound on the loss (probability of mis-classification) incurred by AugUCB. Although UCBEV in literature provides a better guarantee, it is important to emphasize that UCBEV has access to problem complexity (whose computation requires arms' mean and variances), and hence is not realistic in practice; this is in contrast to AugUCB whose implementation does not require any such complexity inputs. We conduct extensive simulation experiments to validate the performance of AugUCB. Through our simulation work, we establish that AugUCB, owing to its utilization of variance estimates, performs significantly better than the state-of-the-art APT, CSAR and other non variance-based algorithms.

1 Introduction

Stochastic multi-armed bandit (MAB) problems are instances of the classic sequential decision-making scenario; specifically an MAB problem comprises of a learner and a collection of actions (or arms), denoted $\mathcal{A}$. In each trial the learner plays (or pulls) an arm $i \in \mathcal{A}$ which yields independent and identically distributed (i.i.d.) reward samples from a distribution (corresponding to arm i), whose expectation is denoted by r_i. The learner's objective is to identify an arm corresponding to the maximum expected reward, denoted r^*. Thus, at each time-step the learner is faced with the *exploration vs. exploitation dilemma*, where it can pull an arm which has yielded the highest mean reward (denoted $\hat{r}_i$) thus far (*exploitation*) or continue to explore other arms with the prospect of finding a better arm whose performance has not been observed sufficiently (*exploration*).

Pure-exploration MAB problems are unlike their traditional (exploration vs. exploitation) counterparts where the objective is to minimize the cumulative regret (which is the total loss incurred by the learner for not playing the optimal arm throughout the time horizon T). Instead, in pure-exploration problems a learning algorithm, until time T, can invest entirely on exploring the arms without being concerned about the loss incurred while exploring; the objective is to minimize the probability that the arm recommended at time T is not the best arm. In this paper, we further consider a combinatorial version of the pure-exploration MAB, called the thresholding bandit problem (TBP). Here, the learning algorithm is provided with a threshold τ, and the objective, after exploring for T rounds, is to output all arms i whose r_i is above τ. It is important to emphasize that the *thresholding* bandit problem is different from the *threshold* bandit setup studied in [Abernethy *et al.*, 2016], where the learner receives an unit reward whenever the value of an observation is above a threshold.

Formally, the problem we consider is the following. First, we define the set $S_\tau = \{i \in \mathcal{A} : r_i \geq \tau\}$. Note that, S_τ is the set of all arms whose reward mean is greater than τ. Let S_τ^c denote the complement of S_τ, i.e., $S_\tau^c = \{i \in \mathcal{A} : r_i < \tau\}$. Next, let $\hat{S}_\tau = \hat{S}_\tau(T) \subseteq \mathcal{A}$ denote the recommendation of a learning algorithm (under consideration) after T time units of exploration, while $\hat{S}_\tau^c$ denotes its complement. The performance of the learning agent is measured by the accuracy with which it can classify the arms into S_τ and S_τ^c after time horizon T. Equivalently, using $\mathbb{I}(E)$ to denote the indicator of an event E, the *loss* $\mathcal{L}(T)$ is defined as

$$\mathcal{L}(T) = \mathbb{I}\big(\{S_\tau \cap \hat{S}_\tau^c \neq \emptyset\} \cup \{\hat{S}_\tau \cap S_\tau^c \neq \emptyset\}\big).$$

Finally, the goal of the learning agent is to minimize the expected loss:

$$\mathbb{E}[\mathcal{L}(T)] = \mathbb{P}\big(\{S_\tau \cap \hat{S}_\tau^c \neq \emptyset\} \cup \{\hat{S}_\tau \cap S_\tau^c \neq \emptyset\}\big).$$

Note that the expected loss is simply the *probability of mis-classification* (i.e., error), that occurs either if a good arm is rejected or a bad arm is accepted as a good one.

The above TBP formulation has several applications, for instance, from areas ranging from anomaly detection and classification ([Locatelli *et al.*, 2016]) to industrial application. Particularly in industrial applications a learners objective is to choose (i.e., keep in operation) all machines whose

productivity is above a threshold. The TBP also finds applications in mobile communications [Audibert and Bubeck, 2010] where the users are to be allocated only those channels whose quality is above an acceptable threshold.

1.1 Related Work

Significant amount of literature is available on the stochastic MAB setting with respect to minimizing the cumulative regret. While the seminal work of [Robbins, 1952], [Thompson, 1933], and [Lai and Robbins, 1985] prove asymptotic lower bounds on the cumulative regret, the more recent work of [Auer et al., 2002] propose the UCB1 algorithm that provides finite time-horizon guarantees. Subsequent work such as [Audibert and Bubeck, 2009] and [Auer and Ortner, 2010] have improved the upper bounds on the cumulative regret. The authors in [Auer and Ortner, 2010] have proposed a *round-based*[1] version of the UCB algorithm, referred to as UCB-Improved. Of special mention is the work of [Audibert et al., 2009] where the authors have introduced a *variance-aware* UCB algorithm, referred to as UCB-V; it is shown that the algorithms that take into account variance estimation along with mean estimation tends to perform better than the algorithms that solely focuses on mean estimation, for instance, such as UCB1. For a more detail survey of literature on UCB algorithms, we refer the reader to [Bubeck and Cesa-Bianchi, 2012].

In this work we are particularly interested in *pure-exploration MABs*, where the focus in primarily on simple regret rather than the cumulative regret. The relationship between cumulative regret and simple regret is proved in [Bubeck et al., 2011] where the authors prove that minimizing the simple regret necessarily results in maximizing the cumulative regret. The pure exploration problem has been explored mainly under the following two settings:

1. Fixed Budget setting: Here the learning algorithm has to suggest the best arm(s) within a fixed time-horizon T, that is usually given as an input. The objective is to maximize the probability of returning the best arm(s). This is the scenario we consider in our paper. In [Audibert and Bubeck, 2010] the authors propose the UCBE and the Successive Reject (SR) algorithm, and prove simple-regret guarantees for the problem of identifying the single best arm. In the combinatorial fixed budget setup [Gabillon et al., 2011] propose the GapE and GapE-V algorithms that suggest, with high probability, the best m arms at the end of the time budget. Similarly, [Bubeck et al., 2013] introduce the Successive Accept Reject (SAR) algorithm, which is an extension of the SR algorithm; SAR is a round based algorithm whereby at the end of each round an arm is either accepted or rejected (based on certain confidence conditions) until the top m arms are suggested at the end of the budget with high probability. A similar combinatorial setup was explored in [Chen et al., 2014] where the authors propose the Combinatorial Successive Accept Reject (CSAR) algorithm, which is similar in concept to SAR but with a more general setup.

2. Fixed Confidence setting: In this setting the learning algorithm has to suggest the best arm(s) with a fixed confidence (given as input) with as fewer number of attempts as possible. The single best arm identification has been studied in [Even-Dar et al., 2006], while for the combinatorial setup [Kalyanakrishnan et al., 2012] have proposed the LUCB algorithm which, on termination, returns m arms which are at least ϵ close to the true top-m arms with probability at least $1 - \delta$. For a detail survey of this setup we refer the reader to [Jamieson and Nowak, 2014].

Apart from these two settings some unified approaches has also been suggested in [Gabillon et al., 2012] which proposes the algorithms UGapEb and UGapEc which can work in both the above two settings. The thresholding bandit problem is a specific instance of the pure-exploration setup of [Chen et al., 2014]. In the latest work of [Locatelli et al., 2016] Anytime Parameter-Free Thresholding (APT) algorithm comes up with an improved anytime guarantee than CSAR for the thresholding bandit problem.

1.2 Our Contribution

In this paper we propose the Augmented UCB (AugUCB) algorithm for the fixed-budget setting of a specific combinatorial, pure-exploration, stochastic MAB called the thresholding bandit problem. AugUCB essentially combines the approach of UCB-Improved, CCB [Liu and Tsuruoka, 2016] and APT algorithms. Our algorithm takes into account the empirical variances of the arms along with mean estimates; to the best of our knowledge this is the first variance-based algorithm for the considered TBP. Thus, we also address an open problem discussed in [Auer and Ortner, 2010] of designing an algorithm that can eliminate arms based on variance estimates. In this regard, note that both CSAR and APT are not variance-based algorithms.

Our theoretical contribution comprises proving an upper bound on the expected loss incurred by AugUCB (Theorem 3.1). In Table 1 we compare the upper bound on the losses incurred by the various algorithms, including AugUCB. The terms $H_1, H_2, H_{CSAR,2}, H_{\sigma,1}$ and $H_{\sigma,2}$ represent various problem complexities, and are as defined in Section 3. From Section 3 we note that, for all $K \geq 8$, we have

$$\log\left(K \log K\right) H_{\sigma,2} > \log(2K) H_{\sigma,2} \geq H_{\sigma,1}.$$

Thus, it follows that the upper bound for UCBEV is better than that for AugUCB. However, implementation of UCBEV algorithm requires $H_{\sigma,1}$ as input, whose computation is not

Table 1: AugUCB vs. State of the art

Algorithm	Upper Bound on Expected Loss
AugUCB	$\exp\left(-\frac{T}{4096 \log(K \log K) H_{\sigma,2}} + \log\left(2KT\right)\right)$
UCBEV	$\exp\left(-\frac{1}{512} \frac{T-2K}{H_{\sigma,1}} + \log\left(6KT\right)\right)$
APT	$\exp\left(-\frac{T}{64H_1} + 2\log((\log(T)+1)K)\right)$
CSAR	$\exp\left(-\frac{T-K}{72\log(K)H_{CSAR,2}} + 2\log(K)\right)$

---

[1]An algorithm is said to be *round-based* if it pulls all the arms equal number of times in each round, and then proceeds to eliminate one or more arms that it identifies to be sub-optimal.

realistic in practice. In contrast, our AugUCB algorithm requires no such complexity factor as input.

Proceeding with the comparisons, we emphasize that the upper bound for AugUCB is, in fact, not comparable with that of APT and CSAR; this is because the complexity term $H_{\sigma,2}$ is not explicitly comparable with either H_1 or $H_{CSAR,2}$. However, through extensive simulation experiments we find that AugUCB significantly outperforms both APT, CSAR and other non variance-based algorithms. In particular, we consider experimental scenarios comprising large number of arms, with the variances of arms in S_τ being large. AugUCB, being variance based, exhibits superior performance under these settings.

The remainder of the paper is organized as follows. In section 2 we present our AugUCB algorithm. Section 3 contains our main theorem on expected loss, while section 4 contains simulation experiments. We finally draw our conclusions in section 5.

2 Augmented-UCB Algorithm

Notation and assumptions: $\mathcal{A}$ denotes the set of arms, and $|\mathcal{A}| = K$ is the number of arms in $\mathcal{A}$. For arm $i \in \mathcal{A}$, we use r_i to denote the true mean of the distribution from which the rewards are sampled, while $\hat{r}_i(t)$ denotes the estimated mean at time t. Formally, using $n_i(t)$ to denote the number of times arm i has been pulled until time t, we have $\hat{r}_i(t) = \frac{1}{n_i(t)} \sum_{z=1}^{n_i(t)} X_{i,z}$, where $X_{i,z}$ is the reward sample received when arm i is pulled for the z-th time. Similarly, we use σ_i^2 to denote the true variance of the reward distribution corresponding to arm i, while $\hat{v}_i(t)$ is the estimated variance, i.e., $\hat{v}_i(t) = \frac{1}{n_i(t)} \sum_{z=1}^{n_i(t)} (X_{i,z} - \hat{r}_i)^2$. Whenever there is no ambiguity about the underlaying time index t, for simplicity we neglect t from the notations and simply use $\hat{r}_i, \hat{v}_i$, and n_i, to denote the respective quantities. Let $\Delta_i = |\tau - r_i|$ denote the distance of the true mean from the threshold τ. Along the lines of [Locatelli *et al.*, 2016] we assume that all the reward distributions are 1-sub-Gaussian (note that, 1-sub-Gaussian includes Gaussian distributions with variance less than 1, distributions supported on an interval of length less than 2, etc). Also, the rewards are assumed to take values in the interval $[0, 1]$.

The Algorithm: The Augmented-UCB (AugUCB) algorithm is presented in Algorithm 1. AugUCB is essentially based on the arm elimination method of the UCB-Improved [Auer and Ortner, 2010], but adapted to the thresholding bandit setting proposed in [Locatelli *et al.*, 2016]. However, unlike the UCB improved (which is based on mean estimation) our algorithm employs *variance estimates* (as in [Audibert *et al.*, 2009]) for arm elimination; to the best of our knowledge this is the first variance-aware algorithm for the thresholding bandit problem. Further, we allow for arm-elimination at each time-step, which is in contrast to the earlier work (e.g., [Auer and Ortner, 2010; Chen *et al.*, 2014]) where the arm elimination task is deferred to the end of the respective exploration rounds. The finer details of the algorithm are presented below.

The active set B_0 is initialized with all the arms from $\mathcal{A}$. We divide the entire budget T into rounds/phases like in UCB-Improved, CCB, SAR and CSAR. At every time-step

---

Algorithm 1 AugUCB

Input: Time budget T; parameter ρ; threshold τ
Initialization: $B_0 = \mathcal{A}$; $m = 0$; $\epsilon_0 = 1$;

$$M = \left\lfloor \frac{1}{2} \log_2 \frac{T}{e} \right\rfloor; \quad \psi_0 = \frac{T\epsilon_0}{\left(\log\left(\frac{3}{16} K \log K\right)\right)^2};$$

$$\ell_0 = \left\lceil \frac{2\psi_0 \log(T\epsilon_0)}{\epsilon_0} \right\rceil; \quad N_0 = K\ell_0$$

for $t = K+1, .., T$ **do**
    Pull arm $j \in \arg\min_{i \in B_m} \left\{ |\hat{r}_i - \tau| - 2s_i \right\}$
    $t \leftarrow t + 1$
    for $i \in B_m$ **do**
        if $(\hat{r}_i + s_i < \tau - s_i)$ or $(\hat{r}_i - s_i > \tau + s_i)$ **then**
            $B_m \leftarrow B_m \backslash \{i\}$ (Arm deletion)
        end if
    end for

    if $t \geq N_m$ and $m \leq M$ **then**
        Reset Parameters
        $\epsilon_{m+1} \leftarrow \frac{\epsilon_m}{2}$
        $B_{m+1} \leftarrow B_m$
        $\psi_{m+1} \leftarrow \frac{T\epsilon_{m+1}}{(\log(\frac{3}{16} K \log K))^2}$
        $\ell_{m+1} \leftarrow \left\lceil \frac{2\psi_{m+1} \log(T\epsilon_{m+1})}{\epsilon_{m+1}} \right\rceil$
        $N_{m+1} \leftarrow t + |B_{m+1}|\ell_{m+1}$
        $m \leftarrow m + 1$
    end if
end for
Output: $\hat{S}_\tau = \{i : \hat{r}_i \geq \tau\}$.

AugUCB checks for arm elimination conditions, while updating parameters at the end of each round. As suggested by [Liu and Tsuruoka, 2016] to make AugUCB to overcome too much early exploration, we no longer pull all the arms equal number of times in each round. Instead, we choose an arm in the active set B_m that minimizes $(|\hat{r}_i - \tau| - 2s_i)$ where

$$s_i = \sqrt{\frac{\rho \psi_m (\hat{v}_i + 1) \log(T\epsilon_m)}{4n_i}}$$

with ρ being the arm elimination parameter and ψ_m being the exploration regulatory factor. The above condition ensures that an arm closer to the threshold τ is pulled; parameter ρ can be used to fine tune the elimination interval. The choice of exploration factor, ψ_m, comes directly from [Audibert and Bubeck, 2010] and [Bubeck *et al.*, 2011] where it is stated that in pure exploration setup, the exploring factor must be linear in T (so that an exponentially small probability of error is achieved) rather than being logarithmic in T (which is more suited for minimizing cumulative regret).

3 Theoretical Results

Let us begin by recalling the following definitions of the *problem complexity* as introduced in [Locatelli *et al.*, 2016]:

$$H_1 = \sum_{i=1}^{K} \frac{1}{\Delta_i^2} \quad \text{and} \quad H_2 = \min_{i \in \mathcal{A}} \frac{i}{\Delta_{(i)}^2}$$

where $(\Delta_{(i)} : i \in \mathcal{A})$ is obtained by arranging $(\Delta_i : i \in \mathcal{A})$ in an increasing order. Also, from [Chen *et al.*, 2014] we have

$$H_{CSAR,2} = \max_{i \in \mathcal{A}} \frac{i}{\Delta_{(i)}^2}.$$

$H_{CSAR,2}$ is the complexity term appearing in the bound for the CSAR algorithm. The relation between the above complexity terms are as follows (see [Locatelli *et al.*, 2016]):

$$H_1 \leq \log(2K)H_2 \quad \text{and} \quad H_1 \leq \log(K)H_{CSAR,2}.$$

As ours is a variance-aware algorithm, we require H_1^σ (as defined in [Gabillon *et al.*, 2011]) that incorporates reward variances into its expression as given below:

$$H_{\sigma,1} = \sum_{i=1}^K \frac{\sigma_i + \sqrt{\sigma_i^2 + (16/3)\Delta_i}}{\Delta_i^2}.$$

Finally, analogous to $H_{CSAR,2}$, in this paper we introduce the complexity term $H_{\sigma,2}$, which is given by

$$H_{\sigma,2} = \max_{i \in \mathcal{A}} \frac{i}{\tilde{\Delta}_{(i)}^2}$$

where $\tilde{\Delta}_i^2 = \frac{\Delta_i^2}{\sigma_i + \sqrt{\sigma_i^2 + (16/3)\Delta_i}}$, and $(\tilde{\Delta}_{(i)})$ is an increasing ordering of $(\tilde{\Delta}_i)$. Following the results in [Audibert and Bubeck, 2010], we can show that

$$H_{\sigma,2} \leq H_{\sigma,1} \leq \overline{\log}(K)H_{\sigma,2} \leq \log(2K)H_{\sigma,2}.$$

Our main result is summarized in the following theorem where we prove an upper bound on the expected loss.

Theorem 3.1. *For $K \geq 4$ and $\rho = 1/3$, the expected loss of the AugUCB algorithm is given by,*

$$\mathbb{E}[\mathcal{L}(T)] \leq 2KT \exp\left(-\frac{T}{4096 \log(K \log K)H_{\sigma,2}}\right).$$

Proof. The proof comprises of two modules. In the first module we investigate the necessary conditions for arm elimination within a specified number of rounds, which is motivated by the technique in [Auer and Ortner, 2010]. Bounds on the arm-elimination probability is then obtained; however, since we use variance estimates, we invoke the Bernstein inequality (as in [Audibert *et al.*, 2009]) rather that the Chernoff-Hoeffding bounds (which is appropriate for the UCB-Improved [Auer and Ortner, 2010]). In the second module, as in [Locatelli *et al.*, 2016], we first define a favourable event that will yield an upper bound on the expected loss. Using union bound, we then incorporate the result from module-1 (on the arm elimination probability), and finally derive the result through a series of simplifications. The details are as follows.

Arm Elimination: Recall the notations used in the algorithm, Also, for each arm $i \in \mathcal{A}$, define $m_i = \min\left\{m | \sqrt{\rho\epsilon_m} < \frac{\Delta_i}{2}\right\}$. In the m_i-th round, whenever $n_i = \ell_{m_i} \geq \frac{2\psi_{m_i} \log(T\epsilon_{m_i})}{\epsilon_{m_i}}$, we obtain (as $\hat{v}_i \in [0,1]$)

$$s_i \leq \sqrt{\frac{\rho(\hat{v}_i + 1)\epsilon_{m_i}}{8}} \leq \frac{\sqrt{\rho\epsilon_{m_i}}}{2} < \frac{\Delta_i}{4}. \tag{1}$$

First, let us consider a bad arm $i \in \mathcal{A}$ (i.e., $r_i < \tau$). We note that, in the m_i-th round whenever $\hat{r}_i \leq r_i + 2s_i$, then arm i is eliminated as a bad arm. This is easy to verify as follows: using (1) we obtain,

$$\hat{r}_i \leq r_i + 2s_i < r_i + \Delta_i - 2s_i = \tau - 2s_i$$

which is precisely one of the elimination conditions in Algorithm 1. Thus, the probability that a bad arm is not eliminated correctly in the m_i-th round (or before) is given by

$$\mathbb{P}(\hat{r}_i > r_i + 2s_i) \leq \mathbb{P}(\hat{r}_i > r_i + 2\bar{s}_i) + \mathbb{P}\left(\hat{v}_i \geq \sigma_i^2 + \sqrt{\rho\epsilon_{m_i}}\right) \tag{2}$$

where

$$\bar{s}_i = \sqrt{\frac{\rho\psi_{m_i}(\sigma_i^2 + \sqrt{\rho\epsilon_{m_i}} + 1)\log(T\epsilon_{m_i})}{4n_i}}$$

Note that, substituting $n_i = \ell_{m_i} \geq \frac{2\psi_{m_i}\log(T\epsilon_{m_i})}{\epsilon_{m_i}}$, $\bar{s}_i$ can be simplified to obtain,

$$2\bar{s}_i \leq \frac{\sqrt{\rho\epsilon_{m_i}(\sigma_i^2 + \sqrt{\rho\epsilon_{m_i}} + 1)}}{2} \leq \sqrt{\rho\epsilon_{m_i}}. \tag{3}$$

The first term in the LHS of (2) can be bounded using the Bernstein inequality as below:

$$\mathbb{P}\left(\hat{r}_i > r_i + 2\bar{s}_i\right)$$
$$\leq \exp\left(-\frac{(2\bar{s}_i)^2 n_i}{2\sigma_i^2 + \frac{4}{3}\bar{s}_i}\right)$$
$$\leq \exp\left(-\frac{\rho\psi_{m_i}(\sigma_i^2 + \sqrt{\rho\epsilon_{m_i}} + 1)\log(T\epsilon_{m_i})}{2\sigma_i^2 + \frac{2}{3}\sqrt{\rho\epsilon_{m_i}}}\right)$$
$$\overset{(a)}{\leq} \exp\left(-\frac{3\rho T\epsilon_{m_i}}{256a^2}\left(\frac{\sigma_i^2 + \sqrt{\rho\epsilon_{m_i}} + 1}{3\sigma_i^2 + \sqrt{\rho\epsilon_{m_i}}}\right)\log(T\epsilon_{m_i})\right)$$
$$:= \exp(-Z_i) \tag{4}$$

where, for simplicity, we have used α_i to denoted the exponent in the inequality (a). Also, note that (a) is obtained by using $\psi_{m_i} = \frac{T\epsilon_{m_i}}{128a^2}$, where $a = (\log(\frac{3}{16}K\log K))$.

The second term in the LHS of (2) can be simplified as follows:

$$\mathbb{P}\left\{\hat{v}_i \geq \sigma_i^2 + \sqrt{\rho\epsilon_{m_i}}\right\}$$
$$\leq \mathbb{P}\left\{\frac{1}{n_i}\sum_{t=1}^{n_i}(X_{i,t} - r_i)^2 - (\hat{r}_i - r_i)^2 \geq \sigma_i^2 + \sqrt{\rho\epsilon_{m_i}}\right\}$$
$$\leq \mathbb{P}\left\{\frac{\sum_{t=1}^{n_i}(X_{i,t} - r_i)^2}{n_i} \geq \sigma_i^2 + \sqrt{\rho\epsilon_{m_i}}\right\}$$
$$\overset{(a)}{\leq} \mathbb{P}\left\{\frac{\sum_{t=1}^{n_i}(X_{i,t} - r_i)^2}{n_i} \geq \sigma_i^2 + 2\bar{s}_i\right\}$$
$$\overset{(b)}{\leq} \exp\left(-\frac{3\rho\psi_{m_i}}{2}\left(\frac{\sigma_i^2 + \sqrt{\rho\epsilon_{m_i}} + 1}{3\sigma_i^2 + \sqrt{\rho\epsilon_{m_i}}}\right)\log(T\epsilon_{m_i})\right)$$
$$= \exp(-Z_i) \tag{5}$$

where inequality (a) is obtained using (3), while (b) follows from the Bernstein inequality.

Thus, using (4) and (5) in (2) we obtain $\mathbb{P}(\hat{r}_i > r_i + 2s_i) \le 2\exp(-Z_i)$. Proceeding similarly, for a good arm $i \in \mathcal{A}$, the probability that it is not correctly eliminated in the m_i-th round (or before) is also bounded by $\mathbb{P}(\hat{r}_i < r_i - 2s_i) \le 2\exp(-Z_i)$. In general, for any $i \in \mathcal{A}$ we have

$$\mathbb{P}(|\hat{r}_i - r_i| > 2s_i) \le 4\exp(-Z_i). \tag{6}$$

Favourable Event: Following the notation in [Locatelli *et al.*, 2016] we define the event

$$\xi = \left\{ \forall i \in \mathcal{A}, \forall t = 1, 2, .., T : |\hat{r}_i - r_i| \le 2s_i \right\}.$$

Note that, on ξ each arm $i \in \mathcal{A}$ is eliminated correctly in the m_i-th round (or before). Thus, it follows that $\mathbb{E}[\mathcal{L}(T)] \le P(\xi^c)$. Since ξ^c can be expressed as an union of the events $(|\hat{r}_i - r_i| > 2s_i)$ for all $i \in \mathcal{A}$ and all $t = 1, 2, \cdots, T$, using union bound we can write

$$\mathbb{E}[\mathcal{L}(T)] \le \sum_{i \in \mathcal{A}} \sum_{t=1}^{T} \mathbb{P}(|\hat{r}_i - r_i| > 2s_i)$$

$$\le \sum_{i \in \mathcal{A}} \sum_{t=1}^{T} 4\exp(-Z_i)$$

$$\le 4T \sum_{i \in \mathcal{A}} \exp\left(-\frac{3\rho T\epsilon_{m_i}}{256a^2} \left(\frac{\sigma_i^2 + \sqrt{\rho\epsilon_{m_i}} + 1}{3\sigma_i^2 + \sqrt{\rho\epsilon_{m_i}}} \right) \log(T\epsilon_{m_i}) \right)$$

$$\overset{(a)}{\le} 4T \sum_{i \in \mathcal{A}} \exp\left(-\frac{3T\Delta_i^2}{4096a^2} \left(\frac{4\sigma_i^2 + \Delta_i + 4}{12\sigma_i^2 + \Delta_i} \right) \log(\tfrac{3}{16}T\Delta_i^2) \right)$$

$$\overset{(b)}{\le} 4T \sum_{i \in \mathcal{A}} \exp\left(-\frac{12T\Delta_i^2}{(12\sigma_i + 12\Delta_i)} \frac{\log(\tfrac{3}{16}K\log K)}{4096a^2} \right)$$

$$\overset{(c)}{\le} 4T \sum_{i \in \mathcal{A}} \exp\left(-\frac{T\Delta_i^2 \log(\tfrac{3}{16}K\log K)}{4096(\sigma_i + \sqrt{\sigma_i^2 + (16/3)\Delta_i})a^2} \right)$$

$$\overset{(d)}{\le} 4T \sum_{i \in \mathcal{A}} \exp\left(-\frac{T\log(\tfrac{3}{16}K\log K)}{4096\tilde{\Delta}_i^{-2}a^2} \right)$$

$$\overset{(e)}{\le} 4T \sum_{i \in \mathcal{A}} \exp\left(-\frac{T\log(\tfrac{3}{16}K\log K)}{4096\max_j(j\tilde{\Delta}_{(j)}^{-2})(\log(\tfrac{3}{16}K\log K))^2} \right)$$

$$\overset{(f)}{\le} 4KT \exp\left(-\frac{T}{4096\log(K\log K)H_{\sigma,2}} \right).$$

The justification for the above simplifications are as follows:
- (a) is obtained by noting that in round m_i we have $\frac{\Delta_i}{4} \le \sqrt{\rho\epsilon_{m_i}} < \frac{\Delta_i}{2}$.
- For (b), we note that the function $x \mapsto x\exp(-Cx^2)$, where $x \in [0, 1]$, is decreasing on $[1/\sqrt{2C}, 1]$ for any $C > 0$ (see [Bubeck *et al.*, 2011; Auer and Ortner, 2010]). Thus, using $C = \lfloor\sqrt{e/T}\rfloor$ and $\min_{j \in \mathcal{A}} \Delta_j = \Delta = \sqrt{\frac{K\log K}{T}} > \sqrt{\frac{e}{T}}$, we obtain (b).
- To obtain (c) we have used the inequality $\Delta_i \le \sqrt{\sigma_i^2 + (16/3)\Delta_i}$ (which holds because $\Delta_i \in [0, 1]$).

- (d) is obtained simply by substituting $\tilde{\Delta}_i = \frac{\Delta_i^2}{\sigma_i + \sqrt{\sigma_i^2 + (16/3)\Delta_i}}$ and $a = \log(\tfrac{3}{16}K\log K)$.
- Finally, to obtain (e) and (f), note that $\tilde{\Delta}_i^{-2} \le i\tilde{\Delta}_i^{-2} \le \max_{j \in \mathcal{A}} j\Delta_{(j)}^{-2} = H_{\sigma,2}$. $\square$

4 Numerical Experiments

In this section, we empirically compare the performance of AugUCB against APT, UCBE, UCBEV, CSAR and the uniform-allocation (UA) algorithms. A brief note about these algorithms are as follows:

- APT: This algorithm is from [Locatelli *et al.*, 2016]; we set $\epsilon = 0.05$, which is the precision with which APT suggests the set of good arms.
- AugUCB: This is the Augmented-UCB algorithm proposed in this paper; as in Theorem 3.1 we set $\rho = \frac{1}{3}$.
- UCBE: This is a modification of the algorithm in [Audibert *et al.*, 2009] (as it was originally proposed for the best arm identification problem); here, we set $a = \frac{T-K}{H_1}$, and at each time-step an arm $i \in \arg\min \left\{ |\hat{r}_i - \tau| - \sqrt{\frac{a}{n_i}} \right\}$ is pulled.
- UCBEV: This is a modification of the algorithm in [Gabillon *et al.*, 2011] (proposed for the TopM problem); its implementation is identical to UCBE, but with $a = \frac{T-2K}{H_{\sigma,1}}$. For theoretical reasons, we show the performance achievable by UCBEV. As mentioned earlier, note that UCBEV's implementation would not be possible in real scenario, as it requires computing the problem complexity $H_{\sigma,1}$.
- CSAR: Modification of the successive-reject algorithm in [Chen *et al.*, 2014]; here, we reject the arm farthest from τ after each round.
- UA: The naive strategy where at each time-step an arm is uniformly sampled from $\mathcal{A}$ (the set of all arms); however, UA is known to be optimal if all arms are equally difficult to classify.

Motivated by the settings considered in [Locatelli *et al.*, 2016], we design six different experimental scenarios that are obtained by varying the arm means and variances. Across all experiments consists of $K = 100$ arms (indexed $i = 1, 2, \cdots, 100$) of which $S_\tau = \{6, 7, \cdots, 10\}$, where we have fixed $\tau = 0.5$. In all the experiments, each algorithm is run independently for 10000 time-steps. At every time-step, the output set, $\hat{S}_\tau$, suggested by each algorithm is recorded; the output is counted as an error if $\hat{S}_\tau \ne S_\tau$. In Figure 1, for each experiment, we have reported the percentage of error incurred by the different algorithms as a function of time; Error percentage is obtained by repeating each experiment independently for 500 iterations, and then respectively computing the fraction of errors. The details of the considered experiments are as follows.

Experiment-1: The reward distributions are Gaussian with means $r_{1:4} = 0.2 + (0:3) \cdot 0.05$, $r_5 = 0.45$, $r_6 = 0.55$, $r_{7:10} = 0.65 + (0:3) \cdot 0.05$ and $r_{11:100} = 0.4$. Thus, the means of the first 10 arms follow an arithmetic progression. The remaining arms have identical means; this setting is chosen because now a significant budget is required in exploring these arms, thus increasing the problem complexity.

(a) Expt-1: Arithmetic Progression (Gaussian)

(b) Expt-2: Geometric Progression (Gaussian)

(c) Expt-3: Three Group Setting (Gaussian)

(d) Expt-4: Two Group Setting (Gaussian)

(e) Expt-5: Two Group Setting (Bernoulli)

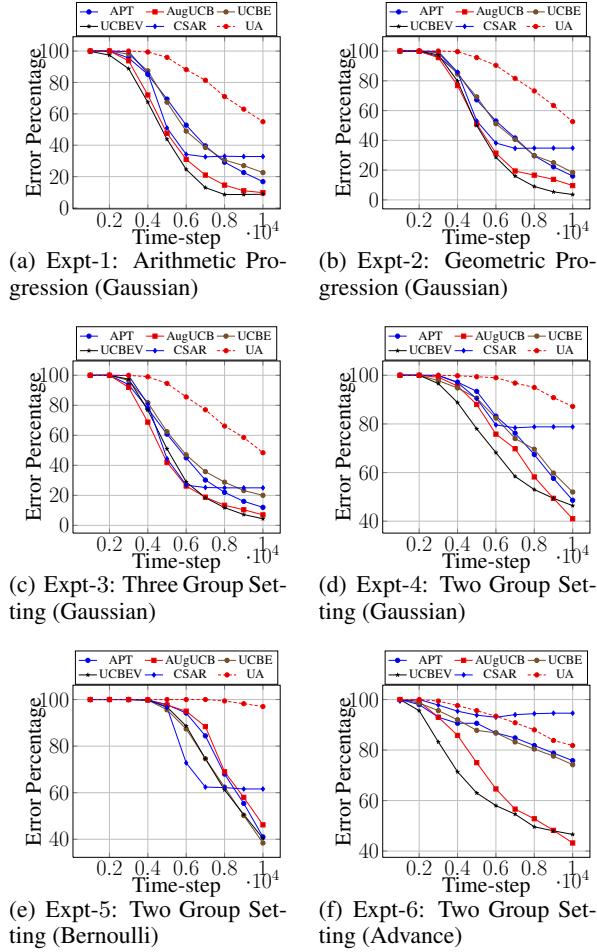
(f) Expt-6: Two Group Setting (Advance)

Figure 1: Performances of the various TBP algorithms in terms of error percentage vs. time-step, for six different experimental scenarios.

The corresponding variances are $\sigma^2_{1:5} = 0.5$ and $\sigma^2_{6:10} = 0.6$, while $\sigma^2_{11:100}$ is chosen independently and uniform in the interval $[0.38, 0.42]$; note that, the variances of the arms in S_τ are higher than those of the other arms. The corresponding results are shown in Figure 1(a), from where we see that UCBEV, which has access to the problem complexity while being variance-aware, outperforms all other algorithm (including UCBE which also has access to the problem complexity but does not take into account the variances of the arms). Interestingly, the performance of our AugUCB (without requiring any complexity input) is comparable with UCBEV, while it outperforms UCBE, APT and the other non variance-aware algorithms that we have considered.

Experiment-2: We again consider Gaussian reward distributions. However, here the means of the first 10 arms constitute a geometric progression. Formally, the reward means are $r_{1:4} = 0.4 - (0.2)^{1:4}$, $r_5 = 0.45$, $r_6 = 0.55$, $r_{7:10} = 0.6 + (0.2)^{5-(1:4)}$ and $r_{11:100} = 0.4$; the arm variances are as in experiment-1. The corresponding results are shown in Figure 1(b). We again observe AugUCB outperforming the other algorithms, except UCBEV.

Experiment-3: Here, the first 10 arms are partitioned into three groups, with all arms in a group being assigned the same mean; the reward distributions are again Gaussian. Specifically, the reward means are $r_{1:3} = 0.1$, $r_{4:7} = \{0.35, 0.45, 0.55, 0.65\}$ and $r_{8:10} = 0.9$; as before, $r_{11:100} = 0.4$ and all the variances are as in Experiment-1. The results for this scenario are presented in Figure 1(c). The observations are inline with the observations made in the previous experiments.

Experiment-4: The setting is similar to that considered in Experiment-3, but with the first 10 arms partitioned into two groups; the respective means are $r_{1:5} = 0.45$, $r_{6:10} = 0.55$. The corresponding results are shown in Figure 1(d), from where the good performance of AugUCB is again validated.

Experiment-5: This is similar to the two group setting considered in Experiment-4, but with the reward distributions being Bernoulli instead of Gaussian. The results for this case are shown in Figure 1(e). We observe that UCBE and UCBEV outperforms AugUCB, while the performance of AugUCB is comparable with that achieved by APT.

Experiment-6: This is again the two group setting involving Gaussian reward distributions. The reward means are as in Experiment-4, while the variances are $\sigma^2_{1:5} = 0.3$ and $\sigma^2_{6:10} = 0.8$; $\sigma^2_{11:100}$ are independently and uniformly chosen in the interval $[0.2, 0.3]$. The corresponding results are shown in Figure 1(f). We refer to this setup as *Advanced* because here the chosen variance values are such that only variance-aware algorithms will perform well. Hence, we see that UCBEV performs very well in comparison with the other algorithms. However, it is interesting to note that the performance of AugUCB catches-up with UCBEV as the time-step increases, while significantly outperforming the other non-variance aware algorithms.

Finally, note that in all the above experiments, the CSAR algorithm, although performs well initially, quickly exhausts its budget and saturates at a higher error percentage. This is because it pulls all arms equally in each round, with the round lengths being non-adaptive.

5 Conclusion

We proposed the AugUCB algorithm for a fixed-budget pure-exploration TBP. Our algorithm employs both mean and variance estimates for arm elimination. This to our knowledge is the first variance-based algorithm for the specific TBP that we have considered. We first prove an upper bound on the expected loss incurred by AugUCB. We then conduct simulation experiments to validate the performance of AugUCB. In comparison with APT, CSAR and other non variance-based algorithms, we find that the performance of AugUCB is significantly better. Further, the performance of AugUCB is comparable with UCBEV (which is also variance-based), although the latter exhibits a slightly better performance. However, UCBEV is not implementable in practice as it requires computing problem complexity, $H_{\sigma,1}$, while AugUCB (requiring no such inputs) can be easily deployed in real-life scenarios. It would be interesting future research to come up with an anytime version of AugUCB algorithm.

References

[Abernethy *et al.*, 2016] Jacob D Abernethy, Kareem Amin, and Ruihao Zhu. Threshold bandits, with and without censored feedback. In *Advances In Neural Information Processing Systems*, pages 4889–4897, 2016.

[Audibert and Bubeck, 2009] Jean-Yves Audibert and Sébastien Bubeck. Minimax policies for adversarial and stochastic bandits. In *COLT*, pages 217–226, 2009.

[Audibert and Bubeck, 2010] Jean-Yves Audibert and Sébastien Bubeck. Best arm identification in multi-armed bandits. In *COLT-23th Conference on Learning Theory-2010*, pages 13–p, 2010.

[Audibert *et al.*, 2009] Jean-Yves Audibert, Rémi Munos, and Csaba Szepesvári. Exploration–exploitation tradeoff using variance estimates in multi-armed bandits. *Theoretical Computer Science*, 410(19):1876–1902, 2009.

[Auer and Ortner, 2010] Peter Auer and Ronald Ortner. Ucb revisited: Improved regret bounds for the stochastic multi-armed bandit problem. *Periodica Mathematica Hungarica*, 61(1-2):55–65, 2010.

[Auer *et al.*, 2002] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

[Bubeck and Cesa-Bianchi, 2012] Sébastien Bubeck and Nicolo Cesa-Bianchi. Regret analysis of stochastic and non-stochastic multi-armed bandit problems. *arXiv preprint arXiv:1204.5721*, 2012.

[Bubeck *et al.*, 2011] Sébastien Bubeck, Rémi Munos, and Gilles Stoltz. Pure exploration in finitely-armed and continuous-armed bandits. *Theoretical Computer Science*, 412(19):1832–1852, 2011.

[Bubeck *et al.*, 2013] Sébastien Bubeck, Tengyao Wang, and Nitin Viswanathan. Multiple identifications in multi-armed bandits. In *ICML (1)*, pages 258–265, 2013.

[Chen *et al.*, 2014] Shouyuan Chen, Tian Lin, Irwin King, Michael R Lyu, and Wei Chen. Combinatorial pure exploration of multi-armed bandits. In *Advances in Neural Information Processing Systems*, pages 379–387, 2014.

[Even-Dar *et al.*, 2006] Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *The Journal of Machine Learning Research*, 7:1079–1105, 2006.

[Gabillon *et al.*, 2011] Victor Gabillon, Mohammad Ghavamzadeh, Alessandro Lazaric, and Sébastien Bubeck. Multi-bandit best arm identification. In *Advances in Neural Information Processing Systems*, pages 2222–2230, 2011.

[Gabillon *et al.*, 2012] Victor Gabillon, Mohammad Ghavamzadeh, and Alessandro Lazaric. Best arm identification: A unified approach to fixed budget and fixed confidence. In *Advances in Neural Information Processing Systems*, pages 3212–3220, 2012.

[Jamieson and Nowak, 2014] Kevin Jamieson and Robert Nowak. Best-arm identification algorithms for multi-armed bandits in the fixed confidence setting. In *Information Sciences and Systems (CISS), 2014 48th Annual Conference on*, pages 1–6. IEEE, 2014.

[Kalyanakrishnan *et al.*, 2012] Shivaram Kalyanakrishnan, Ambuj Tewari, Peter Auer, and Peter Stone. Pac subset selection in stochastic multi-armed bandits. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 655–662, 2012.

[Lai and Robbins, 1985] Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

[Liu and Tsuruoka, 2016] Yun-Ching Liu and Yoshimasa Tsuruoka. Modification of improved upper confidence bounds for regulating exploration in monte-carlo tree search. *Theoretical Computer Science*, 2016.

[Locatelli *et al.*, 2016] Andrea Locatelli, Maurilio Gutzeit, and Alexandra Carpentier. An optimal algorithm for the thresholding bandit problem. *arXiv preprint arXiv:1605.08671*, 2016.

[Robbins, 1952] Herbert Robbins. Some aspects of the sequential design of experiments. In *Herbert Robbins Selected Papers*, pages 169–177. Springer, 1952.

[Thompson, 1933] William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, pages 285–294, 1933.