Bandits with Temporal Stochastic Constraints

Priyank Agrawal\textsuperscript{1} and Theja Tulabandhula\textsuperscript{2}
\textsuperscript{1}Indian Institute of Science
\textsuperscript{2}University of Illinois at Chicago

May 27, 2021

Abstract

We study the effect of impairment on stochastic multi-armed bandits and develop new ways to mitigate it. Impairment effect is the phenomena where an agent only accrues reward for an action if they have played it at least a few times in the recent past. It is practically motivated by repetition and recency effects in domains such as advertising (here consumer behavior may require repeat actions by advertisers) and vocational training (here actions are complex skills that can only be mastered with repetition to get a payoff). Impairment can be naturally modelled as a temporal constraint on the strategy space, and we provide two novel algorithms that achieve sublinear regret, each working with different assumptions on the impairment effect. We introduce a new notion called bucketing in our algorithm design, and show how it can effectively address impairment as well as a broader class of temporal constraints. Our regret bounds explicitly capture the cost of impairment and show that it scales (sub-)linearly with the degree of impairment. Our work complements recent work on modeling delays and corruptions, and we provide experimental evidence supporting our claims.

1 Introduction

The multi-armed bandit (MAB) problem captures exploration-exploitation tradeoff in sequential decision making. In these problems, the algorithm is tasked with learning which arm (action) is the best while simultaneously exploiting its information to obtain rewards. It has been extensively studied in various disciplines such as operations research, statistics and computer science and various solution strategies have been used successfully in domains such as advertising, recommendation systems and clinical trials. In a typical problem, stochastic reward distributions are associated with each arm in a set of arms $\mathcal{K}$. When the algorithm plays arm $j$, it immediately receives a reward with which it can learn, and also suffers a regret, which is the difference between the obtained reward versus the reward that it could have obtained, had it played the best arm.
In one of the key applications of MAB, viz., advertising, there has been a large body of work that models consumer behavior [Hawkins et al., 2009, Solomon et al., 2014]. One such effect that is relatively well studied is the repetition effect [Machleit and Wilson 1988, Campbell and Keller 2003], under which, an advertiser’s payoff (for instance, click through rate) depends on how frequently they have presented the same ad to the same audience in the recent past. If the advertiser presents a specific ad sporadically, then the aggregated payoff is much lower. If this ad is the best among a collection of ads, then the advertiser will not be able to deduce this from their observations. Further, different ads may need different levels of repetition to obtain payoffs, and this may not be known a priori to the advertiser. This phenomenon also translates to recommendations, such as for products and movies, where repeated display of item(s) can cause positive reinforcement to build over time, culminating in a conversion. And since conversions depend on the past recommendations, they interfere with learning the true underlying (mean) payoffs of different recommendations.

In the domain of skill acquisition [DeKeyser 2007], especially those which are complex [Bosse et al., 2015], a learner may have to repeatedly try each action several times to advance or acquire a reward. Further, they may have to repeat these actions frequently enough so as to not lose the acquired skill [Kang 2016]. Under this effect, their ability to learn the best action (or set of actions) may be severely impaired.

Motivated by the above discussion, we define a new class of problems, which we call bandit learning with stochastic impairment, to address the need for repetitions. The defining characteristic here is that a learning algorithm can only accrue rewards if it has played the same arm enough number of times in the recent past. The amount by which the algorithm needs to replay the same arm is a measure of impairment (see Section 2 for a formal treatment). When the arm has been played enough times in the past, then the reward for playing that arm at the current time is instantaneous. In other words, the reward accrued by the algorithm is a function of the current reward generated and the sequence of arms played in the recent past. A diagram illustrating this is shown in Figure 1. Our formulation is an example of combining empirically validated behavioral models with sequential decision making, and a few other examples are discussed towards the end of this section.

In the presence of impairment effect, usual MAB algorithms such as UCB1, SE, MOSS
or Thompson Sampling are ineffective because one cannot directly control the number of times an arm is played in a given time window. In fact, if we have an instance in which a couple of arms have almost equal mean payoffs, then the aforementioned algorithms may switch between these very frequently (see Section 5), potentially causing linear regret. To address these issues, we develop two algorithms, Phased-SE and UCB-Revisited+, which expand on the phase-based algorithmic template to mitigate impairments in reward accrual. Central to the Phased-SE algorithm is the notion of bucketing. The algorithm divides the time horizon into several phases and eliminates arms between phases. In addition, within each phase, it also partitions the existing arms into buckets. This partitioning ensures that arms in each bucket are repeated often. It also lets us use an arm elimination strategy based on SE [Even-Dar et al., 2006] to incentivise exploration-exploitation, and facilitates arm elimination within phases. As we show in our regret upper bound derivations, having buckets has a non-trivial effect on the leading order terms, and also smoothly degrades to a phase-based algorithm’s performance (such as that of UCB-Revisited+ in Section 4) depending on the instance. Our bucketing strategy is quite general and can be used not only in settings with endogenous effects such as the above, but also in settings with exogenous effects. For instance, in a variant of the distributed bandit learning setting [Hillel et al., 2013; Buccapatnam et al., 2015; Gamarnik et al., 2018], one may have to partition the set of arms such that each node can only play its local set. These nodes then have to communicate with each other and minimize regret. Our bucketing strategy works out of the box for this, where communication between nodes happens between phases, and within a phase, nodes run the arm elimination strategy discussed above. For simplicity of exposition, we focus on impairment as the main setting in the rest of the paper.

A unifying view of our contribution in the stochastic MAB landscape, and recent works such as [Pike-Burke et al., 2018; Lykouris et al., 2018] is as follows. In all three works, one can assume that there is a intermediate function that allows an MAB algorithm to accrue a transformation of the rewards. [Pike-Burke et al., 2018] address a challenging setting where the algorithm does not accrue a reward for an arm played immediately. Instead it accrues a linear combination of components of rewards of a stochastic number of arms played in the recent past. [Lykouris et al., 2018] address the setting where the algorithm accrues the stochastic reward most of the time, but an adversary can corrupt some of the rewards arbitrarily, and the algorithm accrues this incorrect reward as well. In our case, we accrue rewards that are modulated by the sequence of actions that our algorithm took. These transformations from true rewards generated to accrued rewards that depend on various factors make the learning problem more challenging.

Our setting is very different than the corruption effect studied in [Gajane et al., 2017; Lykouris et al., 2018]. Gajane et al. [2017] exploit the corruptions in the reward process as a means to preserve privacy. On the other hand, Lykouris et al. [2018] consider settings with arbitrary exogenous corruptions and propose a randomized algorithm (based on SE) that achieves smooth degradation as corruption level increases, and recovers the vanilla MAB guarantee when there is no corruption. Unlike their setting, impairment effect is endogenous and correlates past actions with the current reward. While the amount of impairment in our
setting (if one uses a phase based strategy) would be equivalent to $O(d_{\text{max}}K \log T)$ amount of corruption (see Section 3), we cannot reuse their techniques because of endogeneity. Note that corruption can be unbounded if one uses a regular MAB algorithm such as UCB1.

A stream of works have look at the impact of delayed rewards on learning and regret. In these settings, an algorithm may accrue a reward for its action after a (random) duration of time, possibly mixed with other delayed rewards. For instance, Perchet et al. [2016] consider a setting in which actions are played in batches and rewards are only available at some fixed time points. Works such as Joulani et al. [2013], Cesa-Bianchi et al. [2018] have quantified the impact of delay on regret. In particular, Joulani et al. [2013] provide a recipe to use any regular MAB algorithm in this setting and show that delay causes an additive regret penalty. In Cesa-Bianchi et al. [2018] for adversarial bandits and Pike-Burke et al. [2018] for stochastic bandits, the authors provide regret guarantees for a much weaker setting where rewards can get mixed up and may partially accrue over time. That is, components of rewards due to multiple previous actions may appear collectively. In contrast to the delay effect, which is affecting the future accrual of rewards, the impairment effect can be viewed as being caused by the trajectory of past actions. In both settings, using a phase based algorithmic strategy can mitigate the corresponding effect: for delay, repeating an arm can allow the algorithm to map the reward to the right arm that generated it; for impairment, repeating an arm can allow the algorithm to accrue instantaneous rewards as the arm has been played multiple times in the recent past.

1.1 Our Results and Techniques

Owing to the nature of impairment effect, algorithms necessarily have to ensure that the arms still under consideration are played frequently, perhaps in batches. In particular, if one plays an arm for a sufficiently long period of time, then they can ensure that the instantaneous rewards are always accrued (except some at the beginning). Phase based algorithms are a natural choice given these considerations. This family of algorithms date back to Agrawal et al. [1988], who considered arm switching costs. Our algorithms, PHASED-SE and UCB-Revisited+, follow this design pattern, wherein the focus is to eliminate arms between stages where each arm is played consecutively for multiple rounds. In particular, UCB-Revisited+ is a based on UCB-Revisited Auer and Ortner [2010] and works under the setting when the expected impairment effect is known. On the other hand, PHASED-SE extends UCB-Revisited+ and incorporates the notion of bucketing to further explore and exploit within phases. It assumes knowledge of an upper bound on the stochastic impairment effect. Naturally, both algorithms also work when the impairment is deterministic.

Following the analysis techniques of Auer and Ortner [2010] and Pike-Burke et al. [2018], where the latter uses Freedman’s inequality and the Azuma-Hoeffding inequality based Doob’s optimal skipping theorem (see Section 4), we are able to show additive influence of impairment on worst case regret. While our analysis partially overlaps with these two previous works, the random variables related to impairment in our setting do need a qualitatively different treatment. Assume that an algorithm can accrue a reward for pulling arm $j$ at time $t$ if it has been tried at least $d_{t,j}$ times in the past $N$ rounds, where $d_{t,j}$ is a random variable
Algorithm| Setting | Without impairment | With impairment |
---|---|---|---|
**Lower bound** | | $O(\sqrt{KT})$ | - |
**UCB1** | | $O(\sqrt{KT\log T})$ | - |
**MOSS** | | $O(\sqrt{KT})$ | - |
**PHASED-SE** | | $O(\sqrt{KT\log T})$ | $O\left(\sqrt{KT\log T} + \sqrt{\frac{K^3\log^3(T)}{T}}d_{\text{max}}\right)$ |
**UCB-REVISITED+** | | $O(\sqrt{KT\log T})$ | $O\left(\sqrt{KT\log T} + K\sqrt{\log^2 T\mathbb{E}[d]}\right)$ |

Table 1: Results summary.

with mean $\mathbb{E}[d]$ and upper bound $d_{\text{max}}$ and $N$ is a fixed positive integer. Then, the regret upper bounds of our methods are listed in Table 1. The impact of impairment is sublinear for UCB-REVISITED+ because of a tighter analysis.

### 1.2 Related work

Shah et al. [2018] study bandit problems under a different behavioral effect rooted in microeconomic theory, namely self-reinforcement. In their setting, the algorithm interacts with a sequence of incoming users, who develop preference to the arms that have been played in the past. This positive externality affects the performance of algorithms, where the best arm’s performance may get overshadowed by that of a suboptimal arm due to reinforcement. In our case, the frequency of arm plays in the recent past influences the rewards accrued for that arm, which is a different type of externality.

Impact of imperfect memory, or the difficulty of remembering the observed rewards of arms, on an algorithm’s performance was studied in [Xu and Yun, 2018] and [Gamarnik et al., 2018]. The models are in continuous time and either the rewards are associated with certain arrival and departure processes [Xu and Yun, 2018], or the incoming set of actions are sent to a distributed set of nodes [Gamarnik et al., 2018]. While the spirit of these is similar to our work, they are not closely related to the discrete time bandit learning setting. Our work is one of the first to address impairment in this case, where impairment can be interpreted in terms of constraints on memory.

Bandits with switching costs have also been studied extensively [Banks and Sundaram, 1994], [Dekel et al., 2014] and consider penalties for switching arms too often. Phase based strategies such as the algorithms that we propose can also be considered as solutions to this problem. Impairment effect imposes a hard constraint on switching arms too frequently, which is approximately equivalent to having large switching costs.

Shivaswamy and Joachims [2012] consider the setting where the algorithm has access to previous plays of each arm and focus on warm-starting the exploration-exploitation process. This could be useful in our setting, where one can partition the initial set of arms into buckets such that impairment is mitigated, learn the local best arms in each bucket. Next, these best arms can be combined together (hierarchically agglomerate) and their historical plays
can be used to warm-start. A rigorous analysis of this heuristic is a possible future work. In a similar vein of learning with previously collected observational data, Swaminathan and Joachims [2015] focus on counterfactual risk minimization. While these works work with a one-time use of historical data to improve future payoffs, in our setting we work around the continual impact of historical actions on future payoffs. Maillard and Munos [2011] consider the problem of history-dependent arm selection against possibly adaptive adversaries, which is a much weaker setting compared to ours.

The notion of bucketing incentivises exploration and exploitation within each phase (see Section 3). And this is an important attribute when the environment and specifications are changing. For instance, Kleinberg et al. [2010] introduce the sleeping bandits problem, where the set of available arms changes with time and the algorithm needs to balance exploration and exploitation when new actions are made available. Similar models have also been considered in Chakrabarti et al. [2009], Kanade and Steinke [2014]. In Garivier and Moulines [2008, 2011], the authors assume piece-wise stationarity of the rewards distributions (i.e., the reward distributions abruptly change at certain breakpoints in the time horizon). Under this model, they propose algorithms that estimate mean rewards of arms using only recent history (e.g., using a sliding-window or by discounting). However, none of these overlap with or subsume the impairment effect considered in this work.

2 Problem Definition

There are $K > 1$ arms in the set $\mathcal{K}$. Each arm $j \in \mathcal{K}$ is associated with a reward distribution $\xi_j$, which has support $[0, 1]$. The mean reward for arm $j$ is $\mu_j$. $\mu^*$ is the maximum of all $\mu_j$ and corresponds to the arm $j^*$. Define $\Delta_j = \mu^* - \mu_j$. Additionally, let $\epsilon := \max_{j, k \in \mathcal{K}\backslash\{j^*\}, \Delta_j \leq \Delta_k} \frac{\Delta_j}{\Delta_k}$. This instance dependent parameter captures how dissimilar the arms are with respect to each other. It also influences the performance of our algorithm in Section 3. In particular, a smaller value of $\epsilon$ corresponds to an easy instance (i.e., the true means of arms are well separated), whereas a larger value corresponds to a relatively difficult instance (the true means are close to each other). For easy instances, our algorithm’s regret guarantee is strictly better than a benchmark (see Section 3 for more details).

One of the key aspects of this work is a novel modeling of impairment that temporally correlates the rewards accrued by any algorithm with its past sequence of actions. Each arm $j$ comes with an i.i.d. stochastic process $\{d_{t,j}\}$ that controls reward accrual in the following way: let $R_{t,j}$ be the reward that would be generated if the $j^{th}$ arm is played at time $t$. Further, let $J_t \in \mathcal{K}$ represent the arm that is played at time $t$. The reward accrued by the algorithm at time $t$ is not $R_{t,J_t}$ but the following value:

$$X_{t,j} = R_{t,j} \mathbb{I} \left[ \sum_{k=\max(t-N,0)}^{t} \mathbb{I}[J_k = j] \geq d_{t,j} \right],$$

where $\mathbb{I}[]$ is the indicator function, $N \in \mathbb{N}$ is a instance specific parameter, and for all
\( j \in K \), the random variables \( d_{t,j} \) are supported on the set \( \{0, 1, \ldots, d_{\text{max}}\} \) with \( d_{\text{max}} \leq N \). For simplicity, we assume that the mean of each random variable \( d_{t,j} \) is equal to \( \mathbb{E}[d] \).

Intuitively, the algorithm only accrues reward for playing arm \( j \) at time \( t \) if it has played the same arm at least \( d_{t,j} \) number of times in the past \( N \) rounds. Thus, the bandit instance in our work has additional stochastic processes and a parameter \( N \) for modeling impairment.

Given this context, our objective is to design an online algorithm that receives the temporally dependent bandit feedback shown in (2) and minimizes expected (pseudo-)regret. That is, for a given time horizon of plays \( T \), the algorithm chooses a sequence \( \{J_t\}_{t=1}^T \) and is scored by:

\[
R_T = \max_{k \in [K]} \mathbb{E} \left[ \sum_{t=1}^T X_{t,k} \right] - \mathbb{E} \left[ \sum_{t=1}^T X_{t,J_t} \right].
\]  

Unlike the standard setting, the above terms cannot be further simplified because the random variables \( \{X_{t,j}\} \) depend on both the instantaneous rewards as well as the historical sequence of arms played. Further, due to impairment, even if one knows the optimal arm \( j^* \), they may still receive no rewards for \( O(d_{\text{max}}) \) number of rounds. We assume that the algorithm can differentiate between accruing a reward and not accruing a reward (i.e., not count the latter as 0 reward).

Given the above model, one can deduce that methods such as UCB1 [Auer et al., 2002], MOSS [Audibert and Bubeck, 2009] or Thompson Sampling [Chapelle and Li, 2011] will not succeed for any reasonable values of \( \mathbb{E}[d], d_{\text{max}} \) and \( N \). For instance, the UCB1 algorithm, which is based on the optimism principle, adaptively decreases its optimism over mean rewards such that arms are easily distinguishable from each other. But while doing this, it does not allow for a direct control on how arms switch between rounds. When \( N \to \infty \), the impairment effect fades away completely and the above algorithms will perform well.

To control the switching between arms, one could employ algorithms such as SE [Even-Dar et al., 2006] or UCBR [Auer and Ortner, 2010], which either play arms in a predictable round robin fashion or play the same arm consecutively. For instance, when \( N = d_{\text{max}} \), an algorithm such as UCBR is necessary but not sufficient. In this work, we build on the family of phase based algorithms, which include UCBR and ODAAF [Pike-Burke et al., 2018], and propose a new algorithm that is able to control the arm switching behavior using a novel bucketed strategy. This algorithm works under the setting when only \( d_{\text{max}} \) and \( N \) are known (see Section 3). Further, we also show how a minor modification of UCBR/ODAAF is also a very effective regret minimizing strategy. This latter algorithm works when \( \mathbb{E}[d] \) and \( N \) are known, and comes with a tighter regret upper bound (see Section 4).

### 3 Impairment with Known Support

We introduce a new algorithmic innovation (the idea of bucketing) that significantly extends the phased-based UCB algorithm (UCBR) [Auer and Ortner, 2010], which has been extensively studied and extended by a number of works such as [Hillel et al., 2013, Bui et al.].
Many optimism-based algorithms (for instance, variants of UCB1) tend to switch arms a lot, especially when there are multiple (near-)optimal arms present. In contrast, a phase-based structure enables one to control the temporal pattern of arms being played and limit switching. In previous works, this pattern was limited to consecutively playing the same arm multiple times. For instance, in Pike-Burke et al. [2018], the authors exploit the repetition to tackle delay and anonymous aggregation of rewards.

Using the algorithmic template of UCBR can itself let us meet the impairment constraint (playing $d_{i,j}$ number of times in the past $N$ rounds). But since each arm is played consecutively, there is little exploration and exploitation within a phase. For instance, say there are $K_m$ arms in phase $m$. Then each arm is played an equal number of times successively in a block of rounds, one after the other. Instead, if we reassign the number of times each arm is played so that we spend more rounds on those arms that are more promising, then our regret would be lower. We achieve this in our algorithm by playing arms in a round-robin fashion, with exploration and exploitation similar to the SE algorithm [Even-Dar et al.] [2006], while ensuring that the impairment constraint is not violated. This enables us to eliminate arms during a phase, in addition to elimination between phases. Our proposed algorithm, Phased-SE is described in Algorithm 1.

We execute the round-robin strategy described above in Phased-SE by first constructing multiple subsets of arms in each phase, called buckets, and then running SE on each bucket separately. Thus, at the beginning of each phase, the arms that have not been eliminated yet, are divided up into buckets. Each bucket of arms runs a round-robin arm elimination strategy (SE, see Algorithm 2) that incentivizes exploration and exploitation, with the net result being that: (a) arms which are more promising in each bucket are played more, (b) the impairment constraints are satisfied due to bucket sizing, and (c) the overall regret is reduced. In our performance guarantee (Theorem 3.1), this effect of bucketing is a leading order effect. Bucketing allows our algorithm to work on a strictly larger class of impaired bandit instances when compared to a direct phase based strategy, because if the impairment constraint is tighter than the phase length (which is typically an algorithm parameter), then off-the-shelf phase base strategies will incur linear regret. Finally, note that not only is bucketing helpful in meeting an impairment constraint, it is also useful in general distributed online learning settings (with bandit feedback), where nodes can only play subsets of arms for extended periods of time and communicate with each other to identify the best-arm/minimize regret.

Reward accrual depends on how the buckets are sized. We set the size conservatively to have at most $N/d_{\max}$ number of arms. The arms in buckets can get eliminated before the end of the phase in addition to the possibility of getting eliminated between phases. Within the phase, an arm will get eliminated from its bucket if its mean reward is different from the estimated best arm in that bucket. Whereas, an arm will get eliminated between phases if its mean reward is different from the overall estimated best arm. As the number of phases increase, the estimates of mean rewards of the surviving arms improve. The algorithm uses a monotonically decreasing sequence of thresholds to detect sub-optimal arms, which is a
standard design in phase-based methods. Controlling the thresholds, the phase lengths and
the number of buckets leads to a sub-linear regret for our algorithm in the impaired bandit
setting (Theorem 3.1). While the algorithm takes the time horizon $T$ as input, this can be
relaxed using the doubling trick, which is well known in the literature.

**Notation:** Let $m$ index phases. Let $T_j(m)$ refer to the collection of times when the $j^{th}$
arm is played up to phase $m$ and $n_{t,j}$ denote the number of times the arm $j$ was chosen
till time index $t$. As discussed in earlier sections, $J_t$ refers to the arm pulled at time $t$ and
the reward sequence till time index $t$ is given by $\{X_{k,J_k}\}_{k=0}^t$. The sequence of parameters
$\{n_m|m = 0, 1, 2, \ldots\}$ determine the number of consecutive rounds each active arm is played
in phase $m$, where active arms belong to the set $K_m$. The estimated mean reward for arm $j$
at the end of phase $m$ is denoted by $\hat{X}_{m,j}$. We also denote the collection of buckets in phase
$m$ by $B_m$. In the pseudo-code description of both the Algorithms 1 and 2, we assume that
the flow exits whenever time index crosses the time horizon.

**Gains from Bucketing:** The choice of the size of the bucket is conservative and ensures
that all arms in a bucket satisfy the impairment constraint by construction and at most
$O(d_{\text{max}})$ rewards are lost per arm. The parameter sequence $\{n_m|m = 0, 1, 2, \ldots\}$ is chosen
such that the comparison step between phases, involving progressively shrinking internal
thresholds ($\{\tilde{\Delta}_m\}$), eliminates suitable suboptimal arms with high probability. In particular,
those arms still present in phase $m$, whose regret gaps satisfy $\Delta_i/2 > \tilde{\Delta}_m$, should ideally get
eliminated.

As mentioned before, running SE (Algorithm 2) in each phase incentivizes exploration
and exploitation with the possibility of eliminating arms that are found suboptimal within a
bucket during the phase. SE was proposed under the name Successive elimination in [Even-
Dar et al., 2002] as a (PAC) best-arm identification algorithm and has been further developed
and applied in various subsequent works [Even-Dar et al., 2006, Audibert and Bubeck, 2010,
Gupta et al., 2011, Lykouris et al., 2018]. In this algorithm, arms are played in a round robin
fashion, and in each round their estimated mean reward plus a confidence bound is compared
against the estimated best. A negative outcome of comparison results in the arm getting
eliminated.

The value of $\epsilon$, which is a property of the input instance, governs the regret gap when
SE is run inside each bucket. When the bucket has largely dissimilar arms, the value of $\epsilon$ is
small and Phased-SE is able to explore/exploit via the buckets and eliminate suboptimal
arms relatively quickly. On the other hand, when all arms have the same mean reward in
a bucket, the SE strategy is equivalent in performance to that of consecutively playing the
same arms. In each bucket, potentially $O(d_{\text{max}})$ number of rewards are not accrued due
to impairment (the same happens when arms are played consecutively as well). We show
experimentally, that additional exploration induced by bucketing leads to a non-negligible
effect in decreasing cumulative regret (see Section 5). We also note in passing that we did
not impose any structure other than a size constraint while partitioning $K_m$ into buckets in Phased-SE. There could be different motivations (see Section 1) for several motivating
**Input:** A set of arms $\mathcal{K}$, time horizon $T$, and parameters $\{n_m | m = 0, 1, 2, \ldots \}$.

**Initialization:** phase index $m = 1$, $\mathcal{K}_m = \mathcal{K}$, $\tilde{\Delta}_1 = 1$, $T_j(m - 1) = \emptyset \; \forall j \in \mathcal{K}$, and time index $t = 1$.

**for** $t \leq T$ **do**

**Within Phase Elimination:**

Set $\mathcal{K}_{m+1} = \emptyset$

Partition $\mathcal{K}_m$ into a collection of buckets $\mathcal{B}_m$ of size at most $|\mathcal{K}_m|/(\mathcal{N}/d_{\max})$.

**for each bucket $B_l$ in $\mathcal{B}_m$ do**

Call SE (Algorithm 2) with inputs: set of arms $B_l$, internal time horizon $\mathcal{N}/d(n_m - n_{m-1}), t$ (time offset), $T_j(m - 1) \; \forall j \in B_l$, and overall time horizon $T$.

Get outputs: $B_l' \subseteq B_l$, $\{X_{t,j} \}_{t \in T_j(m) \setminus T_j(m-1), \forall j \in B_l'}$, and new time index $t'$.

$t \leftarrow t'$.

$\mathcal{K}_{m+1} \leftarrow \mathcal{K}_{m+1} \cup B_l'$.

**end**

**End of Phase Elimination:**

**for each active arm $j$ in $\mathcal{K}_m$ do**

$X_{m,j} = \frac{1}{n_m} \sum_{t \in T_j(m)} X_{t,j}$.

**end**

**for each arm $j$ in $\mathcal{K}_{m+1}$ do**

Eliminate it from $\mathcal{K}_{m+1}$ if:

$X_{m,j} + \frac{\Delta_m}{2} < \max_{j' \in \mathcal{K}_{m+1}} X_{m,j'} - \frac{\Delta_m}{2}$.

**end**

Update the confidence bound:

Set $\tilde{\Delta}_{m+1} = \frac{\Delta_m}{2}$.

Increment phase index $m$ by 1.

**end**

---

**Algorithm 1: Phased-SE**
**Required Inputs:** A set of arms $\mathcal{K}$, internal time horizon $H$.

**Optional Inputs:** Time offset $t$, previous counts $T_j$, $\forall j \in \mathcal{K}$, and overall time horizon $T$.

**Outputs:** Set of arms $\mathcal{K}'$, accrued rewards $(\{X_{u,J_u}\}_{u \in T_j' \setminus T_j})$, $\forall j \in \mathcal{K}'$, and new time index $(t+s)$.

**Initialization:** If optional inputs are not provided, then $s = 0$, $T_j = \phi$ $\forall j \in \mathcal{K}$, and overall time horizon $T = H$.

Local round index $s = t+1$, $\mathcal{K}' = \mathcal{K}$, local counts $L_j = 0$ $\forall j \in \mathcal{K}$. $T_j' = T_j$ $\forall j \in \mathcal{K}$.

**while** $s \leq (T+t)$ **do**

**for each active arm $j$ in $\mathcal{K}'$ do**

Play arm $J_s = j$.

$\bar{X}_{s,j} \leftarrow \bar{X}_{s-1,j}$.

if \(\sum_{k=\max\{s-N,0\}}^s I[J_k = j] \geq d_{s,j}\) then

$L_j \leftarrow L_j + 1$.

$\bar{X}_{s,j} \leftarrow \frac{(L_j-1)\bar{X}_{s-1,j} + X_{s,j}}{L_j}$.

$T_j' \leftarrow T_j' \cup \{t + s\}$, and also record the accrued reward $X_{s,j}$.

end

Eliminate arm $j$ from $\mathcal{K}'$ if $\bar{X}_{s,j} + \sqrt{\frac{\log T}{|T_j'|}} < \max_{j' \in \mathcal{K}'} X_{s,j'} - \sqrt{\frac{\log T}{|T_{j'}'|}}$.

$s = s + 1$.

end

**end**

**Algorithm 2:** Successive Elimination (with minor modifications): SE
example) that may lead to preferred partitioning schemes, and our algorithm is general enough to be useful in those settings as well.

**Lemma 3.1.** In Algorithm 2, the expected number of times a suboptimal arm pulled is bounded as: $\mathbb{E}[|T_j'|] \leq 1 + \frac{16 \log(T)}{\Delta_j^2} + \frac{2}{T}$ for $s \leq T$, where $T$ is the overall time horizon, $|T_j'|$ maintains the current count of the number of times that arm $j$ was chosen and $\Delta_j$ is the regret gap.

Proof. Let $CB_{s,j} = \sqrt{\frac{\log(T)}{|T_j'|}}$ and $\mathcal{E} = \mathbb{I}\{|\bar{X}_{t,i} - \mu_i| > CB_{s,j}\}$. Firstly, we see that $\mathbb{P}(\mathcal{E})$ is upper bounded by Hoeffding’s inequality by $\frac{2}{T^2}$. Now, for $T_j'$ satisfying $CB_{s,j} < \Delta_j/4$, the arm elimination condition of 2 is satisfied. This is because:

$$\bar{X}_{s,j} + CB_{s,j} \leq \mu_j + 2CB_{s,j} < \mu_j + \Delta_j - 2CB_{s,j} = \mu_* - 2CB_{s,j} \leq \bar{X}_{s,*} - CB_{s,j}.$$  

With $\Delta_j \geq 4\sqrt{\frac{\log(T)}{|T_j'|}} \to |T_j'| = |T_*'| > 1 + 16\log(T)/\Delta_j = l$, we can calculate as:

$$\mathbb{E}[|T_j'|] = \sum_{s=0}^{T} \left( \mathbb{P}(j = J_s)\mathbb{I}\{|T_j'| \leq l\} + \mathbb{P}(j = J_s)\mathbb{I}\{|T_j'| > l\} \right),$$

which upon simplification gives:

$$\mathbb{E}[|T_j'|] \leq 1 + \frac{16 \log(T)}{\Delta_j^2} + \frac{2}{T}.$$  

\[ \square \]

### 3.1 Analysis

The choice of algorithm parameters $\{n_m|m=0,1,2,\ldots\}$ is as follows: set $n_0 = 0$, and $n_m$ as given by (8). The following regret guarantee holds for Phased-SE.

**Theorem 3.1.** The expected (pseudo-)regret of Phased-SE (Algorithm 1) is upper bounded as shown below:

$$R_T \leq \sum_{i \in K': \Delta_i > \lambda} \frac{4\Delta_i}{T}$$

$$+ \sum_{i \in K': \Delta_i > \lambda} \left( \Delta_i + \frac{16 \log(T)}{\Delta_i} \min \left\{ \frac{1}{(1-\epsilon)^2}, 4 \right\} + 2\Delta_i \log \left( \frac{4}{\Delta_i} \right) d_{\text{max}} \right)$$

$$+ \frac{16}{T} \max_{i \in K'': \Delta_i < \lambda} \Delta_i T,$$

where $K'' = \{i \in K|\Delta_i > 0\}$ and $K' = \{i \in K|\Delta_i > \lambda\}$.  

12
In the above expression, the leading order terms have \(\log T\) dependence. From the expression, we can also see that the upper bound gets tighter as \(\epsilon\) decreases, and this is due to bucketing. When the arms are very similar to each other, then the likelihood of arms getting eliminated within a phase decreases, and we recover the performance guarantee of running only a phase-based strategy (more details on this variant are provided at the end of this section). The impairment effect leads to an additive factor that is proportional to \(d_{\text{max}}\).

We can also get the following instance independent bound using a standard reduction. This matches the performance of several algorithms including UCB1, UCB-REVISITED and SE. The bound is a factor \(\sqrt{\log T}\) off from the best possible, which is achieved by the MOSS algorithm [Audibert and Bubeck, 2009].

**Corollary 3.1.** For all \(T > K\), the expected (pseudo-)regret of Phased-SE is upper bounded as:

\[
R_T \leq O \left( \sqrt{KT \log T} + \sqrt{K^3 \log^3(T) d_{\text{max}}} \right).
\]

To get this result, we start with the regret upper bound given by the Theorem 3.1 and optimize the parameter \(\lambda\) and use \(\lambda = \sqrt{K \log(T) / T}\). We additionally note that \(\log(1 / \hat{\Delta}_m) \leq O(\log T)\) due to halving of the confidence estimate in each phase. The second term is the penalty due to impairment, and varies linearly with the difficulty level of the problem, i.e., \(O(d_{\text{max}})\). Because \(T\) is in the denominator for the term involving \(d_{\text{max}}\), the effect of impairment decreases quickly as the horizon length increases.

**Proof of Theorem 3.1:** The following proof bounds regret by considering four mutually exclusive and exhaustive scenarios. In a phase, let \(B_*\) denote the bucket which would contain the optimal arm \(*\). For each sub-optimal arm \(i\), let \(m_i = \min\{m | \hat{\Delta}_m < \Delta_i / 2\}\) be the first phase where \(\hat{\Delta}_m < \Delta_i / 2\). In terms of our algorithm, \(m_i\) should be the first phase where arm \(i\) gets eliminated with high probability. By the definition of \(\hat{\Delta}_m\) and \(m_i\), we have the following inequalities:

\[
2^{m_i} = \frac{1}{\hat{\Delta}_m} \leq \frac{4}{\Delta_i} \leq \frac{1}{\hat{\Delta}_{m+1}} = 2^{m_i+1}.
\]

We upper bound the regret by bounding the expected number of times a sub-optimal arm, \(i\) is pulled. Cases (a) and (c) bound the probabilities when we grossly when the reward estimate deviate from true mean by large amounts. Case (b) is for the situation when the arm is pulled exactly \(n_{m_i}\) times. Finally, Case (d) accounts for regret for the nearly optimal arms.

**Case (a)** Arm \(i\) is not deleted in phase \(m_i\) with the optimal arm \(*) in the set \(K_{m_i}\).

The phase \(m_i\) is characterized by: \(\sqrt{\log(T) / n_{m_i}} \leq \Delta_i / 2 \leq \Delta_i / 4\). If \(i \in B_*\), we analyze by upper bounding the probabilities at the end of the phase. If the following inequalities hold:

\[
\overline{\mu}_i \leq \mu_i + \sqrt{\frac{\log(T)}{n_{m_i}}},
\]

This matches the performance of several algorithms including UCB1, UCB-REVISITED and SE. The bound is a factor \(\sqrt{\log T}\) off from the best possible, which is achieved by the MOSS algorithm [Audibert and Bubeck, 2009].

**Corollary 3.1.** For all \(T > K\), the expected (pseudo-)regret of Phased-SE is upper bounded as:

\[
R_T \leq O \left( \sqrt{KT \log T} + \sqrt{K^3 \log^3(T) d_{\text{max}}} \right).
\]

To get this result, we start with the regret upper bound given by the Theorem 3.1 and optimize the parameter \(\lambda\) and use \(\lambda = \sqrt{K \log(T) / T}\). We additionally note that \(\log(1 / \hat{\Delta}_m) \leq O(\log T)\) due to halving of the confidence estimate in each phase. The second term is the penalty due to impairment, and varies linearly with the difficulty level of the problem, i.e., \(O(d_{\text{max}})\). Because \(T\) is in the denominator for the term involving \(d_{\text{max}}\), the effect of impairment decreases quickly as the horizon length increases.

**Proof of Theorem 3.1:** The following proof bounds regret by considering four mutually exclusive and exhaustive scenarios. In a phase, let \(B_*\) denote the bucket which would contain the optimal arm \(*\). For each sub-optimal arm \(i\), let \(m_i = \min\{m | \hat{\Delta}_m < \Delta_i / 2\}\) be the first phase where \(\hat{\Delta}_m < \Delta_i / 2\). In terms of our algorithm, \(m_i\) should be the first phase where arm \(i\) gets eliminated with high probability. By the definition of \(\hat{\Delta}_m\) and \(m_i\), we have the following inequalities:

\[
2^{m_i} = \frac{1}{\hat{\Delta}_m} \leq \frac{4}{\Delta_i} \leq \frac{1}{\hat{\Delta}_{m+1}} = 2^{m_i+1}.
\]

We upper bound the regret by bounding the expected number of times a sub-optimal arm, \(i\) is pulled. Cases (a) and (c) bound the probabilities when we grossly when the reward estimate deviate from true mean by large amounts. Case (b) is for the situation when the arm is pulled exactly \(n_{m_i}\) times. Finally, Case (d) accounts for regret for the nearly optimal arms.

**Case (a)** Arm \(i\) is not deleted in phase \(m_i\) with the optimal arm \(*) in the set \(K_{m_i}\).

The phase \(m_i\) is characterized by: \(\sqrt{\log(T) / n_{m_i}} \leq \Delta_i / 2 \leq \Delta_i / 4\). If \(i \in B_*\), we analyze by upper bounding the probabilities at the end of the phase. If the following inequalities hold:

\[
\overline{\mu}_i \leq \mu_i + \sqrt{\frac{\log(T)}{n_{m_i}}},
\]

This matches the performance of several algorithms including UCB1, UCB-REVISITED and SE. The bound is a factor \(\sqrt{\log T}\) off from the best possible, which is achieved by the MOSS algorithm [Audibert and Bubeck, 2009].

**Corollary 3.1.** For all \(T > K\), the expected (pseudo-)regret of Phased-SE is upper bounded as:

\[
R_T \leq O \left( \sqrt{KT \log T} + \sqrt{K^3 \log^3(T) d_{\text{max}}} \right).
\]

To get this result, we start with the regret upper bound given by the Theorem 3.1 and optimize the parameter \(\lambda\) and use \(\lambda = \sqrt{K \log(T) / T}\). We additionally note that \(\log(1 / \hat{\Delta}_m) \leq O(\log T)\) due to halving of the confidence estimate in each phase. The second term is the penalty due to impairment, and varies linearly with the difficulty level of the problem, i.e., \(O(d_{\text{max}})\). Because \(T\) is in the denominator for the term involving \(d_{\text{max}}\), the effect of impairment decreases quickly as the horizon length increases.

**Proof of Theorem 3.1:** The following proof bounds regret by considering four mutually exclusive and exhaustive scenarios. In a phase, let \(B_*\) denote the bucket which would contain the optimal arm \(*\). For each sub-optimal arm \(i\), let \(m_i = \min\{m | \hat{\Delta}_m < \Delta_i / 2\}\) be the first phase where \(\hat{\Delta}_m < \Delta_i / 2\). In terms of our algorithm, \(m_i\) should be the first phase where arm \(i\) gets eliminated with high probability. By the definition of \(\hat{\Delta}_m\) and \(m_i\), we have the following inequalities:

\[
2^{m_i} = \frac{1}{\hat{\Delta}_m} \leq \frac{4}{\Delta_i} \leq \frac{1}{\hat{\Delta}_{m+1}} = 2^{m_i+1}.
\]

We upper bound the regret by bounding the expected number of times a sub-optimal arm, \(i\) is pulled. Cases (a) and (c) bound the probabilities when we grossly when the reward estimate deviate from true mean by large amounts. Case (b) is for the situation when the arm is pulled exactly \(n_{m_i}\) times. Finally, Case (d) accounts for regret for the nearly optimal arms.

**Case (a)** Arm \(i\) is not deleted in phase \(m_i\) with the optimal arm \(*) in the set \(K_{m_i}\).

The phase \(m_i\) is characterized by: \(\sqrt{\log(T) / n_{m_i}} \leq \Delta_i / 2 \leq \Delta_i / 4\). If \(i \in B_*\), we analyze by upper bounding the probabilities at the end of the phase. If the following inequalities hold:

\[
\overline{\mu}_i \leq \mu_i + \sqrt{\frac{\log(T)}{n_{m_i}}},
\]

13
and
\[ \overline{\mu}_* \geq \mu_* - \sqrt{\frac{\log(T)}{n_{m_i}}} \] (4)
then the end of phase elimination condition of the Algorithm [1] are satisfied, as:
\[ \overline{\mu} + \sqrt{\frac{\log(T)}{n_{m_i}}} \leq \mu_* + 2 \sqrt{\frac{\log(T)}{n_{m_i}}} < \mu_* + \Delta_i \] and
\[ \overline{\mu} + \sqrt{\frac{\log(T)}{n_{m_i}}} \leq \mu_* - \sqrt{\frac{\log(T)}{n_{m_i}}} \leq \mu_* + \Delta_i - 2 \sqrt{\frac{\log(T)}{n_{m_i}}} \]
On the contrary, if the arm \( i \) did not get eliminated at the end of the phase \( m_i \), then (3) and (4) cannot hold. We can then bound the inverse of the events in (3) and (4) by Hoeffding’s bounds as:
\[ P(\overline{\mu}_i > \mu_i + \sqrt{\frac{\log(T)}{n_{m_i}}}) \leq \frac{1}{T^2} \] (5)
Similarly:
\[ P(\overline{\mu}_* < \mu_* + \sqrt{\frac{\log(T)}{n_{m_i}}}) \leq \frac{1}{T^2} \] (6)
By Union bound, the probability of arm \( i \) not being eliminated in the phase \( m_i \) is upper bounded by \( \frac{2}{T^2} \).
Let \( i \notin B_* \) and \( j \) be the locally arm optimal in that bucket. \( \frac{\Delta_i - \Delta_j}{2} > \Delta_{m_i} \), then the arm \( i \) should be eliminated inside the bucket. By the definition of \( \epsilon \), we know \( \frac{\Delta_i - \Delta_j}{2} > \frac{\Delta_i (1 - \epsilon)}{2} \).
Similar to above, the probability of the arm not getting eliminated inside the bucket but at the end of the phase is upper bounded by \( \frac{2}{T^2} \). This event is subsumed by the event: arm not getting eliminated at the end of the bucket. Hence, \( \forall j \in K' \), the expected regret conditioned on the events of this case is given by:
\[ R_T \leq \sum_{i \in K'} \frac{2}{T^2} (T \Delta_i) \leq \sum_{i \in K'} \frac{2 \Delta_i}{T} \]
**Case (b)** Arm \( i \) is eliminated at the phase \( m_i \) or during the phase \( m_i \) with the optimal arm \( * \in K_{m_i} \).
This case marks the crucial benefit in exploration-exploitation due to use of buckets. If \( i \in B_* \), then it would be same as if arms in \( B_* \) were been pulled by SE (Algorithm [2]) since the beginning of the horizon. So, from Lemma 3.1
\[ \mathbb{E}[n_{t,i}] \leq 1 + \frac{16 \log(T)}{\Delta_i^2} + \frac{2}{T} \]
If \( i \notin B_* \), then in terms of \( \epsilon \) a similar bound would hold:
\[ \mathbb{E}[n_{t,i}] \leq 1 + \frac{16 \log(T)}{\Delta_i^2 (1 - \epsilon)^2} + \frac{2}{T} \]
On the contrary, with bucket size= 1 and \( t \) = time at the end of phase \( m_i \), \( n_{m_i} = n_{t,i} \leq \left( 1 + \frac{64 \log(T)}{\Delta_i^2} + 2 \log(4/\Delta_i) d_{\max} \right) \). Using this we can combine both the bounds as:

\[
\mathbb{E}[n_{t,i}] \leq 1 + \frac{16 \log(T)}{\Delta_i^2} \min \left\{ \frac{1}{(1-\epsilon)^2}, 4 \right\} + 2 \log(4/\Delta_i) d + \frac{2}{T}.
\]

Hence the expected regret conditioned on the events of this case is given by:

\[
R_T \leq \sum_{i \in K'} \left( \Delta_i + \frac{16 \log(T)}{\Delta_i} \min \left\{ \frac{1}{(1-\epsilon)^2}, 4 \right\} + 2 \Delta_i \log(4/\Delta_i) d + \frac{2\Delta_i}{T} \right),
\]

and \( \epsilon < 1/2 \) offers clear advantage in terms of regret bounds when using a bucketed strategy.

**Case (c) Optimal arm \( * \) deleted by some sub-optimal \( i \) in the set \( K''' \).**

Now, we consider the case when the last of all the optimal arms (in case there more than one), \( * \) is eliminated by some sub optimal arm \( i \) in \( K''' = \{ i \in K | \Delta_i > 0 \} \) in some round \( m_\ast \) (Overloading the definition of \( m_\ast \), we imply that \( m_\ast \) is any round where \( * \) is eliminated). As elimination of the optimal arm can be induced by larger number of arms, the whole set \( K''' \), at the end of the phase as compared to during a phase, we only need to analyze at the end of a phase for upper bounds. This is similar in spirit to the event in Case (a), as now as well (3) and (4) cannot hold together with the elimination condition of the Algorithm 1.

Hence the probability of this happening is again upper bounded by \( \frac{2}{T^2} \) by similar arguments.

The optimal arm, \( * \) belonged to \( K_{m_\ast} \) for all the sub-optimal arms \( s \) with \( m_\ast < m_s \). Therefore arm \( i \), which causes elimination of the optimal arm \( * \), should satisfy \( m_i \geq m_\ast \). Therefore the regret is upper bounded by:

\[
R_T \leq \sum_{m_\ast=0}^{\max_{j \in K'} m_j} \sum_{i \in K'' : m_i \leq m_\ast} \frac{2}{T^2} T \max_{j \in K'' : m_j \geq m_\ast} \Delta_i,
\]

\[
\leq \sum_{m_\ast=0}^{\max_{j \in K'} m_j} \sum_{i \in K'' : m_i \leq m_\ast} \frac{2}{T^4} \tilde{\Delta}_{m_\ast},
\]

\[
\leq \sum_{i \in K''} \sum_{m_\ast \geq 0} \frac{8}{T} \theta^{-m_\ast}.
\]

\[
R_T \leq \sum_{i \in K''} \frac{16}{T}.
\]

**Case (d) Arm \( i \in K'' \) and \( \notin K' \).**

Here, we account for the difference in the sets \( K'' \) and \( K' \). Following gives the upper bound on the regret conditioned on this case:

\[
R_T \leq \max_{i \in K'' : \Delta_i < \lambda} \Delta_i T.
\]

As all the four cases are mutually exclusive and exhaustive, we thus, get the desired regret upper bound.
3.2 Learning without Bucketing

When the bucket size is set to 1, then Phased-SE (Algorithm 1) is equivalent to UCB-Revisited [Auer and Ortner, 2010] (which is also the approach we take in Section 4, see Algorithm 3). In this case, the choice of \( n_m \) is given by (8).

**Lemma 3.2.** The expected (pseudo-)regret of Algorithm 3 when \( d_{\text{max}} \) is known is upper bounded as:

\[
R_T \leq \sum_{i \in K: \Delta_i > \lambda} \left( \Delta_i + \frac{64 \log(T)}{\Delta_i} + 2 \Delta_i \log \left( \frac{4}{\Delta_i} \right) d_{\text{max}} \right) + \sum_{i \in K: \Delta_i > \lambda} \frac{2 \Delta_i}{T} + \sum_{i \in K: \Delta_i < \lambda} \frac{16}{T} \Delta_i T,
\]

(7)

where \( K'' = \{ i \in K | \Delta_i > 0 \} \) and \( K' = \{ i \in K | \Delta_i > \lambda \} \).

**Proof.** The proof is along the lines of the proof for UCB-Revisited in [Auer and Ortner, 2010]. In particular, we redefine the algorithm parameter \( n_m \) as below to account for impairment:

\[
n_m = \left\lceil \frac{4 \log(T)}{\Delta_m^2} \right\rceil + m d_{\text{max}}.
\]

(8)

The extra \( m d_{\text{max}} \) term allows the algorithm to play each active arm an extra \( d_{\text{max}} \) times in each phase, overcoming the impairment effect. As a consequence, in any phase \( m \), \( \bar{X}_{m,j} - \mu_j \leq \Delta_m / 2 \) \( \forall j \in K_m \) with high probability. \( \square \)

When compared to the regret bound of Phased-SE in Theorem 3.1 we can see that the above bound will be worse for input instances whose corresponding \( \epsilon \) is small. And this non-trivial difference is due to the use of buckets, each running SE.

4 Known Expected Impairment

Just like in the previous section and also similar to the algorithm designs of [Auer and Ortner, 2010], [Karnin et al., 2013], [Valko et al., 2014] and [Pike-Burke et al., 2018], our algorithm also works in phases. In each phase, every arm in the set of active arms is played consecutively (unlike Section 3 where there are played in a round-robin fashion inside buckets, here the bucket size can be considered as 1). Again we argue that repetition ensures that the stochastic impairment is limited. As in these previous works, the rewards collected for each arm are used to update the estimate of the mean rewards. Between two consecutive phases, arms with low estimated mean rewards are eliminated (unlike Section 3 where they can be eliminated in any round). The threshold chosen to eliminate the arms is updated sequentially.

As more number of phases are executed, the estimated mean rewards for the active arms get closer to their true means and the threshold is suitably adjusted. Because of repeat plays
of the same arm for a carefully chosen number of rounds, the effect of reward impairment is mitigated. Similar to previous works, the algorithm has access to the horizon length $T$. This information can be relaxed by adopting the doubling trick as mentioned in Section 3. Finally, note that in Pike-Burke et al. [2018], the algorithm has a bridge period to address the issue of aggregation of delayed rewards. In Auer and Ortner [2010], the algorithm chooses a different length for each phase, compared to our choice. They also have a different function of $\tilde{\Delta}_m$ in the arm elimination step.

**Notation:** Let $m$ index phases. Let $T_j(m)$ refer to the collection of times when the $j^{th}$ arm is played up to phase $m$. As in previous section, $J_t$ is the arm pulled at time $t$, $X_t$ is the reward accrued at time $t$ and $R_{t,j}$ is the rewards observed on playing arm $j$ at time $t$. The sequence of parameters \( \{n_m | m = 0, 1, 2, ...\} \) determine the number of consecutive rounds each active arm is played in phase $m$, where active arms belong to the set $\mathcal{K}_m$. The estimated mean reward for arm $j$ at the end of phase $m$ is denoted by $\bar{X}_{m,j}$.

**Input:** A set of arms $\mathcal{K}$, time horizon $T$, and parameters $\{n_m | m = 0, 1, 2, ...\}$.

**Initialization:** phase index $m = 1$, $\mathcal{K}_m = \mathcal{K}$, $\Delta_1 = 1$, $T_j(m) = \phi \ \forall j \in \mathcal{K}$, and time index $t = 1$.

**while** $t \leq T$ **do**

| Play arms: |
| --- |
| **for** each active arm $j$ in $\mathcal{K}_m$ **do** |
| Set $T_j(m) = T_j(m-1)$ if $m > 1$. |
| Play $j$ for $n_m - n_{m-1}$ consecutive rounds. In each round $t$: |
| if $\left( \sum_{k=\max(t-N,0)}^{t} I[J_k = j] \right) \geq d_{t,j}$ **then** |
| Observe reward $X_{t,j}$ and add $t$ to $T_j(m)$. |
| **end** |
| Eliminate Suboptimal Arms: |
| **for** each active arm $j$ in $\mathcal{K}_m$ **do** |
| $\bar{X}_{m,j} = \frac{1}{n_m} \sum_{t \in T_j(m)} X_{t,j}$. |
| **end** |
| Construct $\mathcal{K}_{m+1}$ by eliminating arms $j$ in $\mathcal{K}_m$ for which $\bar{X}_{m,j} + \Delta_m/2 < \max_{j' \in \mathcal{K}_m} \bar{X}_{m,j'} - \Delta_m/2$. |

**Update the confidence bound:**

- Set $\Delta_{m+1} = \frac{\Delta_m}{2}$.
- Increment phase index $m$ by 1.

**Algorithm 3: UCB-Revisited+**
4.1 Analysis

Unlike the result in Section 3, here we perform a more careful analysis to choose the values of the sequence \{n_m|m = 0, 1, 2\ldots\}. This yields the following regret upper bound, where the dependence on the expected impairment value \(E[d]\) is sublinear. In contrast, the regret upper bound for Phased-SE depends linearly in \(d_{\text{max}}\).

**Theorem 4.1.** The expected (pseudo-)regret of UCB-Revisited+ (Algorithm 3) is bounded as:

\[
R_T \leq \sum_{i \in K'} \left( \Delta_i + \frac{64 \log(T)}{\Delta_i} + \frac{64 \log(T)}{3} + 32 \sqrt{\log \left( \frac{4}{\Delta_i} \right) E[d] \log(T)} \right) + \sum_{i \in K': \Delta_i > \lambda} \frac{2\Delta_i}{T} + \sum_{i \in K'} \frac{32}{T} + \max_{i \in K'': \Delta_i < \lambda} \Delta_i T,
\]

where \(K'' = \{i \in K | \Delta_i > 0\}\) and \(K' = \{i \in K | \Delta_i > \lambda\}\).

Similar to the Corollary 3.1, an instance independent bound for this setting is shown below.

**Corollary 4.1.** For all \(T \geq K\) and choosing \(\lambda = \sqrt{\frac{K \log(T)}{T}}\) the expected (pseudo-)regret of UCB-Revisited+ is upper bounded as:

\[
R_T \leq O\left( \sqrt{KT \log T} + K \sqrt{\log^2 T E[d]} \right).
\]

The key difference with Corollary 3.1 is that in above there is milder dependence on the impairment parameter (\(\sqrt{E[d]}\) vs \(d_{\text{max}}\)) but with a worse time dependence in the second term. However, here Algorithm 3 is doing a significantly harder job as only \(E[d]\) is available to it. The proof of the result in Theorem 4.1 is along the lines of an analysis for a phase based algorithm called ODAAF that was designed to address delays in reward accrual in Pike-Burke et al. [2018]. It poses very different challenges, e.g. rewards due to multiple arms may get accrued collectively, at a given time instance in the delay model (see Section 1 for a detailed comparison of these two models). In impairment setting learning occurs by repeating actions appropriate number of times. Even though the philosophy of (10) is very similar to (13) in Pike-Burke et al. [2018], we obtain different order of regret bounds.

There are two key aspect to the proof: (a) identifying an appropriate \(n_m\) that can work with stochastic impairment effects, (b) showing that this \(n_m\) swiftly eliminates the sub-optimal arms in the execution of UCB-Revisited+ (Algorithm 3). We first provide several key lemmas that will be useful to prove the main claim, which we defer to until after Lemma 4.5.

We first start with two concentration inequalities concerning certain classes of stochastic processes below:
Lemma 4.1. Generalized Bernstein inequality for Martingales (Theorem 1.6 in Freedman 1975, Theorem 10 in Pike-Burke et al. 2018) Let \( \{Y_k\}_{k=0}^\infty \) be a real valued martingale with respect to the filtration, \( \{\mathcal{F}_k\}_{k=0}^\infty \) with increments \( \{Z_k\}_{k=1}^\infty \), implying \( \mathbb{E}[Z_k|\mathcal{F}_{k-1}] = 0 \) and \( Z_k = Y_k - Y_{k-1} \) for \( k = 1, 2, 3, \ldots \). Given the martingale difference sequence is uniformly upper bounded as, \( Z_k \leq b \) for \( k = 1, 2, 3, \ldots \). Define the predictable variation process \( W_k = \sum_{j=1}^k \mathbb{E}[Z_j^2|\mathcal{F}_{j-1}] \) for \( k = 1, 2, \ldots \). Then for all \( \alpha \geq 0 \), \( \sigma^2 \geq 0 \), the following probability is bounded:

\[
\mathbb{P}\left( \exists k : Y_k \geq \alpha \text{ and } W_k \leq \sigma^2 \right) \leq \exp\left(-\frac{\alpha^2/2}{\sigma^2 + \alpha \eta / 3}\right).
\]

Lemma 4.2. Concentration of bounded random variables (Lemma A.1 in Szita and Szepesvári 2011, Lemma 11 in Pike-Burke et al. 2018) Fix the positive integers \( m, n \) and let \( a, c \in \mathbb{R} \). Let \( \mathcal{F} = \{\mathcal{F}_t\}_{t=0}^\infty \) be a filtration, \( \{\rho_t\}_{t=1,2,\ldots,n} \) be \( \{0,1\} \)-valued and \( \mathcal{F}_{t-1} \)-measurable random variables and \( \{Z_t\}_{t=1,2,\ldots,n} \) be \( \mathcal{F}_t \)-measurable \( \mathbb{R} \)-valued random variables, satisfying \( \mathbb{E}[Z_t|\mathcal{F}_{t-1}] = 0 \), \( Z_t \in [a, a + c] \) and \( \sum_{s=1}^n \rho_s \leq m \) with probability one. Then for any \( \eta > 0 \):

\[
\mathbb{P}\left( \sum_{t=1}^n \rho_t Z_t \geq \eta \right) \leq \exp\left(-\frac{2\eta^2}{c^2 m}\right).
\]

Now, we setup some additional notation. For any active arm \( j \) and phase \( m \), let \( S_{m,j} \) denote the time in this phase when the algorithm starts playing this arm. Similarly let \( U_{m,j} \) denote the time in this phase when the algorithm stops playing this arm. All the phases are distinct and non-overlapping. Except for an initial few plays, reward accrual is instantaneous by construction. We define a filtration \( \{\mathcal{G}_s\}_{s=0}^\infty \) by setting \( \mathcal{G}_0 = \Omega, \phi \) and defining \( \mathcal{G}_t \) to be the \( \sigma \)-algebra over \( (X_1, X_2, \ldots, X_t, J_1, J_2, \ldots, J_t, d_{1,t,j}, \ldots, d_{t,j}, R_{1,j}, R_{2,j}, \ldots, R_{t,j}) \).

In the next lemma, we give a constructive proof for the choice of \( n_m \) such that the estimated mean reward for an arm \( j \) is close to its true mean within a predefined error range with high probability.

Lemma 4.3. There exists a positive \( n_m \) for which the estimates \( \overline{X}_{m,j} \), calculated by the Algorithm \( \mathcal{B} \) for the active arm \( j \) ( \( j \in K_m \) ) and phase \( m \), satisfy the following with probability \( \geq 1 - \frac{2}{m^2} \):

\[
\overline{X}_{m,j} - \mu_j \leq \hat{\Delta}_m/2.
\]

Proof. Using the above notation, it follows that for each arm \( j \):

\[
\sum_{i=1}^m \sum_{t=S_{i,j}}^{U_{i,j}} (X_t - \mu_j) \leq \sum_{i=1}^m \sum_{t=S_{i,j}}^{U_{i,j}} (R_{t,j} - \mu_j) - \sum_{i=1}^m \sum_{t=S_{i,j}}^{U_{i,j}} R_{t,j} \mathbb{I}\{t \leq S_{i,j} + d_{t,j}\}. \tag{10}
\]

Define \( A_{t,j} := R_{t,j} \mathbb{I}\{t \leq S_{i,j} + d_{t,j}\} \) and also \( M_t := \sum_{i=m}^m A_{i,t} \mathbb{I}\{S_{i,j} \leq t \leq U_{i,j}\} \). We rewrite (10) in terms of \( M_t \) as:

\[
\sum_{i=1}^m \sum_{t=S_{i,j}}^{U_{i,j}} (X_t - \mu_j) \leq \sum_{i=1}^m \sum_{t=S_{i,j}}^{U_{i,j}} (R_{t,j} - \mu_j) - \sum_{t=1} U_{m,j} M_t,
\]
\[
\sum_{i=1}^{m} \sum_{t=S_{i,j}} U_{i,j} (X_t - \mu_j) \leq \sum_{i=1}^{m} \sum_{t=S_{i,j}} (R_{i,t} - \mu_j) + \sum_{t=1}^{U_{m,j}} (E[M_t | G_{t-1}] - M_t) - \sum_{t=1}^{E[M_t | G_{t-1}]} (E[M_t | G_{t-1}]).\quad (11)
\]

We bound each term individually in (11) from lemmas 4.5 and 4.7 accompanied by a trivial non-negative upper bound for the term 3 in (11), we can write with probability 1 \(-\frac{2}{T^2}\) (from union bound)

\[
\sum_{i=1}^{m} \sum_{t=S_{i,j}} U_{i,j} (X_t - \mu_j) \leq \sqrt{n_m \log(T)} + \frac{2}{3} \log(T) + \sqrt{\frac{4 \log^2(T)}{9}} + 4mE[d] \log(T).
\]

For each active arm \(j \in K_m\),

\[
\frac{1}{n_m} \sum_{t \in T_j(m)} (X_t - \mu_j) \leq \sqrt{\frac{\log(T)}{n_m}} + \frac{2 \log(T)}{3n_m} + \frac{1}{n_m} \sqrt{\frac{4 \log^2(T)}{9}} + 4mE[d] \log(T) = w_m.
\]

Algorithm 3 requires \(w_m \leq \tilde{\Delta}_m/2\) so that the arm elimination condition holds good. This helps to determine appropriate \(n_m\). Let \(z = \sqrt{\frac{4 \log^2(T)}{9}} + 4mE[d] \log(T)\), then, we need to solve the following quadratic inequality:

\[
\frac{\tilde{\Delta}_m}{2} n_m - \sqrt{n_m \log(T)} - \frac{2 \log(T)}{3} - z \geq 0.
\]

The smallest \(n_m\) which satisfies the above is given by:

\[
n_m = \left\lfloor \frac{1}{\tilde{\Delta}_m^2} \left( \sqrt{\frac{\log(T)}{n_m}} + \sqrt{\frac{4 \log(T)}{3n_m} + \frac{4 \tilde{\Delta}_m \log(T)}{3} + 2 \tilde{\Delta}_m z} \right)^2 \right\rfloor.
\]

Using \((a + b)^2 \leq 2(a^2 + b^2)\) and \(x = \lfloor y \rfloor \rightarrow x \leq y + 1:\)

\[
n_m \leq 1 + \frac{1}{\tilde{\Delta}_m^2} \left( 4 \log(T) + \frac{8 \tilde{\Delta}_m \log(T)}{3} + 4 \tilde{\Delta}_m z \right).
\]

We can now substitute \(z\) and use inequality \(\sqrt{a^2 + b^2} \leq (a + b)\), therefore:

\[
n_m \leq 1 + \frac{1}{\tilde{\Delta}_m^2} \left( 4 \log(T) + \frac{16 \tilde{\Delta}_m \log(T)}{3} + 8 \tilde{\Delta}_m \sqrt{mE[d] \log(T)} \right),\]

\[
n_m \leq 1 + \frac{4 \log(T)}{\tilde{\Delta}_m^2} + \frac{16 \log(T)}{3 \tilde{\Delta}_m} + \frac{8 \sqrt{mE[d] \log(T)}}{\tilde{\Delta}_m}.
\]

This completes the proof. \(\square\)
Lemma 4.4. \( Y_s = \sum_{t=1}^{s} \mathbb{E}[M_t|G_{t-1} - M_t] \) for all \( s \geq 1 \) with \( Y_0 = 0 \) is a martingale with respect to the filtration \( \{G_s\}_{s=0}^{\infty} \) with increments \( Z_s = Y_s - Y_{s-1} = \mathbb{E}[M_s|G_{s-1} - M_s] \), satisfying \( \mathbb{E}[Z_s|G_{s-1}] = 0 \), \( Z_s \leq 1 \) for all \( s \geq 1 \).

Proof. To show \( \{Y_s\}_{s=0}^{\infty} \) is a martingale defined on filtration \( \{G_s\}_{s=0}^{\infty} \), we need to show \( Y_s \) is \( \{G_s\} \)-measurable for all \( s \geq 1 \) and \( \mathbb{E}[Y_s|G_{s-1}] = Y_{s-1} \)

By the definition of \( \sigma \)-algebra, \( \{G_s\}_{s=0}^{\infty} \), \( d_{t,j}, R_{t,j} \) are all \( \{G_s\} \)-measurable for \( t \leq s \). Additionally for phases \( i \) where time instance \( t \) lie in phases after \( i \), \( \mathbb{I}\{S_{i,j} \leq t \leq U_{i,j}\} = 0 \) (measurable by \( G_0 \) ). Hence \( \{Y_s\}_{s=0}^{\infty} \) is measurable by \( \{G_s\}_{s=0}^{\infty} \). Now consider the conditional expectation:

\[
\mathbb{E}[Y_s|\{G_{s-1}\}] = \mathbb{E} \left[ \sum_{t=1}^{s} (\mathbb{E}[M_t|G_{t-1}] - M_t)|G_{s-1} \right],
\]

\[
= \mathbb{E} \left[ \sum_{t=1}^{s-1} (\mathbb{E}[M_t|G_{t-1}] - M_t)|G_{s-1}\right] + \mathbb{E} \left[ (\mathbb{E}[M_s|G_{s-1}] - M_s)|G_{s-1}\right],
\]

\[
= \mathbb{E} \left[ \sum_{t=1}^{s-1} (\mathbb{E}[M_t|G_{t-1}] - M_t)|G_{s-1}\right] = Y_{s-1}.
\]

Therefore \( \{Y_s\}_{s=0}^{\infty} \) is a martingale with respect to the filtration \( \{G_s\}_{s=0}^{\infty} \). Clearly, the increment \( Z_s = Y_s - Y_{s-1} = (\mathbb{E}[M_s|G_{s-1}] - M_s) \) and \( \mathbb{E}[Z_s|G_{s-1}] = (\mathbb{E}[M_s|G_{s-1}] - M_s)|G_{s-1}| = 0 \)

Note that for any phase \( i \), \( A_{i,t} \leq 1 \) as reward \( R_{t,i} \) is bounded by 1. Also, for any time \( t \) \( M_t \leq 1 \) as \( \mathbb{I}\{S_{i,j} \leq t \leq U_{i,j}\} \) is 1 for only a particular phase \( i \) thus \( Z_s \leq 1 \) for \( s \geq 1 \). This completes the proof \( \square \)

Lemma 4.5. Using the notations derived before:

\[
\mathbb{P} \left( \sum_{t=1}^{m} \sum_{j=S_{i,j}}^{U_{m,j}} (R_{t,i,j} - \mu_j) \leq \sqrt{\log(T)} \right) > 1 - \frac{1}{T^2}.
\]

Proof. We would apply lemma 4.2 to prove the above. Take \( n = T \), \( \mathcal{F}_t \) as filtration with \( \sigma \)-algebra on \( \{X_1, ..., X_t, R_{1,j}, ..., R_{t,i,j}\}_{j=1,2,...,T} \). \( Z_t = R_{t,i,j} - \mu_j \) and \( \rho_t = \mathbb{I}\{J_t = j, t \leq U_{m,j}\} \). Clearly, \( \forall t \in T_j(m) \), \( \rho_t = 1 \) and by definition of \( T_j(m) \), \( \sum_{t=1}^{T} \rho_t = |T_j(m)| \leq n_m \). So \( \sum_{t \in T_j(m)} (R_{t,i,j} - \mu_j) = \sum_{t=1}^{T} \rho_t (R_{t,i,j} - \mu_j) \) follows.

Additionally, for any \( 1 \leq t \leq T \), \( \rho_t = \mathbb{I}\{J_t = j, t \leq U_{m,j}\} \) is \( \mathcal{F}_{t-1} \)-measurable. Given all the observations \( X_1, X_2, ..., X_{t-1} \) till \( (t-1) \) we know which phase does \( t \) belong. This is because of the phased the phased nature of Algorithm 3 thus, \( \mathbb{I}\{t \leq U_{m,j}\} \) is determined. Similarly \( J_t = j \) can be determined, establishing \( \rho_t \) is \( \mathcal{F}_{t-1} \)-measurable. \( Z_t \) is \( \mathcal{F}_t \)-measurable by definition. Taking \( a = -\mu_j \) and \( c = 1 \), we apply lemma 4.2 to get the above result. \( \square \)

Lemma 4.6. For any \( t \), \( Z_t = \mathbb{E}[M_t|G_{s-1}] - M_t \), then

\[
\sum_{t=1}^{U_{m,j}} \mathbb{E}[Z_t^2|G_{s-1}] \leq m \mathbb{E}[d].
\]

21
Proof.

\[
U_{m,j} \sum_{t=1}^{n} \mathbb{E}[Z_t^2|G_{s-1}] = \sum_{t=1}^{n} \mathbb{V}[M_t^2|G_{s-1}] \leq \sum_{t=1}^{n} \mathbb{E}[M_t^2|G_{s-1}] = \sum_{t=1}^{n} \mathbb{E} \left[ \left( \sum_{i=1}^{m} A_{i,t} \mathbb{I}\{S_{i,j} \leq t \leq U_{i,j}\} \right)^2 \middle| G_{t-1} \right].
\]

For phase \(i\), the indicator \(\mathbb{I}\{S_{i,j} \leq t \leq U_{i,j}\} = 1\) for atmost once. Therefore:

\[
\sum_{t=1}^{n} \mathbb{E}[Z_t^2|G_{s-1}] \leq \sum_{t=1}^{n} \mathbb{E} \left[ \sum_{i=1}^{m} A_{i,t}^2 \mathbb{I}\{S_{i,j} \leq t \leq U_{i,j}\} \middle| G_{t-1} \right],
\]

\[
= \sum_{i=1}^{m} \sum_{t=S_{i,j}}^{U_{i,j}} \mathbb{E} \left[ \sum_{i=1}^{m} A_{i,t}^2 \mathbb{I}\{S_{i,j} \leq t \leq U_{i,j}\} \middle| G_{t-1} \right],
\]

\[
= \sum_{i=1}^{m} \sum_{t=S_{i,j}}^{U_{i,j}} \mathbb{E} \left[ A_{i,t}^2 \mathbb{I}\{S_{i,j} \leq t \leq U_{i,j}\} \middle| G_{t-1} \right].
\]

As \(S_{i,j}\) and \(U_{i,j}\) are \(G_{t-1}\)-measurable, if \(\mathbb{I}\{S_{i,j} \leq t \leq U_{i,j}\} = 1\), therefore:

\[
\leq \sum_{i=1}^{m} \sum_{t=S_{i,j}}^{U_{i,j}} \mathbb{E}[A_{i,t}^2|G_{t-1}],
\]

\[
= \sum_{i=1}^{m} \sum_{t=S_{i,j}}^{U_{i,j}} \mathbb{E}[R_{i,t,j}^2 \mathbb{I}\{t < S_{i,j} + d_{i,t,j}\} \middle| G_{t-1}],
\]

\[
\leq \sum_{i=1}^{m} \sum_{t=S_{i,j}}^{U_{i,j}} \mathbb{E}[\mathbb{I}\{t < S_{i,j} + d_{i,t,j}\} \middle| G_{t-1}],
\]

\[
= \sum_{i=1}^{m} \sum_{s=0}^{\infty} \sum_{s'=s}^{\infty} \sum_{t=s}^{\infty} \mathbb{E}[\mathbb{I}\{S_{i,j} = s, U_{i,j} = s', t < s + d_{i,t,j}\} \middle| G_{t-1}],
\]

\[
= \sum_{i=1}^{m} \sum_{s=0}^{\infty} \sum_{s'=s}^{\infty} \sum_{t=s}^{\infty} \mathbb{P}(t < s + d_{i,t,j}),
\]

\[
\leq \sum_{i=1}^{m} \sum_{s=0}^{\infty} \sum_{s'=s}^{\infty} \sum_{t=s}^{\infty} \mathbb{P}(l < d),
\]

\[
\leq \sum_{i=1}^{m} \mathbb{E}[d] = m \mathbb{E}[d].
\]

This completes the proof. \(\square\)
Lemma 4.7. Using the notations derived before:

\[
\mathbb{P}\left( \sum_{t=1}^{t_{m,j}} (\mathbb{E}[M_t|G_{t-1}] - M_t) < \frac{2}{3} \log(T) + \sqrt{\frac{4 \log^2(T)}{9} + 4m \mathbb{E}[d] \log(T)} \right) > 1 - \frac{1}{T^2}.
\]

Proof. Using Lemma 4.4, we know \(Y_s = \sum_{t=1}^{s} \mathbb{E}[M_t|G_{t-1}] - M_t\) for all \(s \geq 1\) with \(Y_0 = 0\) is a martingale with respect to the filtration \(\{G_s\}_{s=0}^{\infty}\) with increments \(Z_s = Y_s - Y_{s-1} = \mathbb{E}[M_s|G_{s-1}] - M_s\), satisfying \(\mathbb{E}[Z_s|G_{s-1}] = 0\), \(Z_s \leq 1\) for all \(s \geq 1\). Also from lemma 4.6, \(\sum_{s=1}^{s} \mathbb{E}[Z^2_s|G_{t-1}] \leq m \mathbb{E}[d]\). As lemma 4.1 guarantees the existence of a \(t\) for which the above inequality is satisfied with high probability.

Proof of Theorem 4.1:

Proof. Proceeding as in the proof of Theorem 3.1, we create 4 mutually exclusive and exhaustive cases with very similar interpretation with the difference being in this setting elimination at the end of the phase is only considered. We then bound the expected regret conditioned on the events of these cases. Recall that for each sub-optimal arm \(i\), let \(m_i := \min\{m|\tilde{\Delta}_m < \Delta_i^2\}\), is the first phase where \(\tilde{\Delta}_m < \Delta_i^2\). We bound the expected regret by following cases:

**Case (a) Arm \(i\) is not deleted in phase \(m_i\) with the optimal arm \(*\) in the set \(K_{m_i}\).**

The phase \(m_i\) is characterized by: \(w_{m_i} \leq \Delta_{m_i} \leq \frac{\Delta_i}{2}\). Let \(E := \mathbb{I}\{\bar{\mu}_i \leq \mu_i + w_{m_i}\}\) and \(R := \mathbb{I}\{\bar{\mu}_* \geq \mu_* - w_{m_i}\}\).

If the events \(E\) and \(R\) hold then the phase-end elimination condition of the Algorithm 3 are satisfied, as:

\(\bar{\mu}_i + w_{m_i} \leq \mu_i + 2w_{m_i} < \mu_i + \Delta_i - 2w_{m_i} \leq \bar{\mu}_* - w_{m_i}\).

From the Lemma 4.3, \(\mathbb{P}(E > 1 - \frac{1}{T^2})\) and \(\mathbb{P}(E > 1 - \frac{1}{T})\) follows. In this case we are interested in conditional regret of \(\{E^c \cup R^c\}\) which by Union bound bears the upper bound of \(\frac{4}{T^2}\), which is given by:

\[
\sum_{i \in K^c} \frac{4}{T^2} T\Delta_i \leq \sum_{i \in K^c} \frac{4\Delta_i}{T}.
\]

**Case (b) Arm \(i\) is eliminated at the phase \(m_i\) with the optimal arm \(*\) \(\in K_{m_i}\).**

By the lemma 4.3 in case the sub-optimal arm \(i\) is eliminated in the phase \(m_i\) then the maximum number of times it is played is given by \(\{12\}\). Additionally, we make use of \(\Delta_i/4 \leq \tilde{\Delta}_m \leq \Delta_i/2\) and \(m_i \leq \log_2 \left( \frac{4}{\Delta_i} \right) < 2 \log \left( \frac{4}{\Delta_i} \right)\). Therefore:

\[
n_{m_i} \leq 1 + \frac{64 \log(T)}{\Delta_i^2} + \frac{64 \log(T)}{3 \Delta_i} + \frac{32 \sqrt{\log \left( \frac{4}{\Delta_i} \right) \mathbb{E}[d] \log(T)}}{\Delta_i}.
\]
\[ R_T \leq \sum_{i \in K'} \Delta_i \left( 1 + \frac{64 \log(T)}{\Delta_i^2} + \frac{64 \log(T)}{3 \Delta_i} + \frac{32 \sqrt{\log \left( \frac{4}{\Delta_i} \right)} \mathbb{E}[d \log(T)]}{\Delta_i} \right), \]

\[ R_T \leq \sum_{i \in K'} \left( \Delta_i + \frac{64 \log(T)}{\Delta_i} + \frac{64 \log(T)}{3} + 32 \sqrt{\log \left( \frac{4}{\Delta_i} \right)} \mathbb{E}[d \log(T)] \right). \]

The following two cases exactly the same as for the proof of Theorem 3.1.

**Case (c) Optimal arm \(*\) deleted by some sub-optimal \(i\) in the set \(K''\).**

Recall the events \(E\) and \(R\) as discussed in **Case (a)**, the event in which \(*\) is eliminated by some suboptimal arm \(i\) in \(K'' = \{i \in K | \Delta_i > 0\}\), is simply \(\{E^c \cup R^c\}\) with role of \(*\) and \(i\) reversed. Hence the probability of this happening is again upper bounded by \(\frac{4}{T^2}\) by similar arguments. Let \(m_*\) imply the round where \(*\) is eliminated.

Following the same argument, as the **Case (c)** of the previous section, arm \(i\), which causes elimination of the optimal arm \(*\), should satisfy \(m_i \geq m_*\). Therefore the regret is upper bounded by:

\[ R_T \leq \max_{j \in K''} \sum_{m_j=0}^{m_*} \frac{4}{T^2} T \max_{j \in K''; m_j \geq m_*} \Delta_j, \]

\[ \leq \sum_{m_*=0}^{\max_{j \in K''} m_j} \sum_{i \in K''; m_i \leq m_*} \frac{4}{T} 4 \hat{\Delta}_{m_*}, \]

\[ \leq \sum_{i \in K''; m_* \geq 0} \frac{16}{T} 2^{-m_*}, \]

\[ \leq \sum_{i \in K''} \frac{32}{T}. \]

**Case (d) Arm \(i \in K''\) and \(\not\in K'\).**

Following gives the upper bound on the regret conditioned on this case:

\[ R_T \leq \max_{i \in K''; \Delta_i < \lambda} \Delta_i T. \]

As all the four cases mutually exclusive and comprehensive, we thus, conclude the proof of the Theorem 4.1.

\[ \square \]

5 Experiments

We run three experiments. In the first, we investigate the arm switching behavior of UCB1. In the second, we investigate the effect of bucketing when there is no impairment present. In
We use a simple setup of $K = 30$ arms and set the reward distributions to be the Bernoulli with randomly chosen biases. The horizon length $T = 5000$. We then run UCB1 under three different configurations. In the first, the bandit instance is run as is and there is an unique optimal arm. In the second, the number of optimal arms is increased to 3, and in the third the number of optimal arms is increased to 7. Figure 2 shows the unnormalized counts of same arm plays in the past 15 plays. This was computed by checking how many times the current arm was also played in the past 15 rounds. As expected, as the number of optimal arms increases, the counts of same arm plays decreases rapidly. This indicates that UCB1 and other related algorithms may perform poorly in settings with impairment.

Next, we compared the performance of Phased-SE (Algorithm 1) against SE and UCB1 when there is no impairment. While our algorithms are expected to do worse than UCB1 and SE, our aim here is to investigate the role of bucket size. Our instance has 20 arms, and we choose a time horizon of 5000 rounds. The distributions associated with each arm follow Bernoulli with a randomly chosen bias. As shown in Figures 3 and 4, the performance of Phased-SE interpolates between that of SE when the bucket size is large (= 20) and UCB-Revisited when the bucket size is small (= 3).

Finally, we show the performance of UCB-Revisited+ (Algorithm 3) for varying levels of impairment. The number of arms in this experiment is 10. Impairment is stochastic and is simulated using the absolute value normal distribution with means $= \{2, 6, 10, 14\}$ and the standard deviation being proportional to the arm index. The fixed impairment parameter is $N = 20$ and the time horizon is 10000. From Figure 5, we can observe that as the cumulative regret increases as $\mathbb{E}[d]$ is increased. It can also be shown that the regret of UCB-Revisited+ grows as $O(\sqrt{\mathbb{E}[d]})$ (omitted).
Figure 3: Cumulative regret as a function of time: performance of Phased-SE for small bucket sizes is closer to the performance of UCB-Revisited.

Figure 4: Cumulative regret as a function of time: performance of Phased-SE for large bucket sizes is closer to the performance of SE.

Figure 5: Performance (cumulative regret) of UCB-Revisited+ as the impairment level is varied.
6 Conclusion

In this work, we have addressed an interesting gap in the stochastic multi-armed bandits literature that models impairment, a phenomena where an algorithm only accrues reward if it has tried an arm multiple times in the recent past. This temporal dependence in collecting rewards is markedly different from extant literature on modeling delayed rewards. Our model of impairment captures both endogenous constraints (such as the algorithm being able to collect a reward only on multiple tries) and exogenous constraints (such as in distributed settings where only a subset of arms can be played for extended periods of time). Under this model, we develop new algorithms that either rotate between feasible arms in a round-robin fashion or play the same arm consecutively and show how their performance is impacted by impairment. Crucial to our algorithm designs is the concept of bucketing, that not only allows a further tradeoff in exploration and exploitation, but also allows us to meet arbitrary impairment restrictions. It also has uses in distributed online learning settings. Some future directions include proving tight lower bounds for the problem class, generalizing impairment to depend on past actions as well as rewards, and extending the bucketing strategy to the scenario where expected impairment is known.

References

Rajeev Agrawal, MV Hedge, and Demosthenis Teneketzis. Asymptotically efficient adaptive allocation rules for the multiarmed bandit problem with switching cost. *IEEE Transactions on Automatic Control*, 33(10):899–906, 1988.

Jean-Yves Audibert and Sébastien Bubeck. Minimax policies for adversarial and stochastic bandits. In *Conference on Learning Theory*, pages 217–226, 2009.

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

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

Jeffrey S Banks and Rangarajan K Sundaram. Switching costs and the Gittins index. *Econometrica*, pages 687–694, 1994.

Hans Martin Bosse, Jonathan Mohr, Beate Buss, Markus Krautter, Peter Weyrich, Wolfgang Herzog, Jana Jünger, and Christoph Nikendei. The benefit of repetitive skills training and frequency of expert feedback in the early acquisition of procedural skills. *BMC medical education*, 15(1):22, 2015.
Swapna Buccapatnam, Atilla Eryilmaz, and Ness B Shroff. Stochastic bandits with side observations on networks. ACM SIGMETRICS Performance Evaluation Review, 42(1):289–300, 2014.

Swapna Buccapatnam, Jian Tan, and Li Zhang. Information sharing in distributed stochastic bandits. In IEEE Conference on Computer Communications, pages 2605–2613. IEEE, 2015.

Loc X Bui, Ramesh Johari, and Shie Mannor. Committing bandits. In Neural Information Processing Systems, pages 1557–1565, 2011.

Margaret C Campbell and Kevin Lane Keller. Brand familiarity and advertising repetition effects. Journal of Consumer Research, 30(2):292–304, 2003.

Nicolo Cesa-Bianchi, Claudio Gentile, and Yishay Mansour. Nonstochastic bandits with composite anonymous feedback. In Conference on Learning Theory, pages 750–773, 2018.

Deepayan Chakrabarti, Ravi Kumar, Filip Radlinski, and Eli Upfal. Mortal multi-armed bandits. In Neural Information Processing Systems, pages 273–280, 2009.

Olivier Chapelle and Lihong Li. An empirical evaluation of Thompson sampling. In Neural Information Processing Systems, pages 2249–2257, 2011.

Ofer Dekel, Jian Ding, Tomer Koren, and Yuval Peres. Bandits with switching costs: pow(T,2/3) regret. In ACM Symposium on Theory of Computing, pages 459–467. ACM, 2014.

Robert DeKeyser. Skill acquisition theory. Theories in second language acquisition: An introduction, 97113, 2007.

Eyal Even-Dar, Shie Mannor, and Yishay Mansour. PAC bounds for multi-armed bandit and Markov decision processes. In Conference on Learning Theory, pages 255–270. Springer, 2002.

Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. Journal of Machine Learning Research, 7(1):1079–1105, 2006.

David A Freedman. On tail probabilities for martingales. Annals of Probability, pages 100–118, 1975.

Pratik Gajane, Tanguy Urvoy, and Emilie Kaufmann. Corrupt bandits for privacy preserving input. ArXiv preprint arXiv:1708.05033, 2017.

David Gamarnik, John N Tsitsiklis, and Martin Zubeldia. Delay, memory, and messaging tradeoffs in distributed service systems. Stochastic Systems, 8(1):45–74, 2018.
Aurélien Garivier and Eric Moulines. On upper-confidence bound policies for non-stationary bandit problems. *ArXiv preprint arXiv:0805.3415*, 2008.

Aurélien Garivier and Eric Moulines. On upper-confidence bound policies for switching bandit problems. In *Algorithmic Learning Theory*, pages 174–188. Springer, 2011.

Neha Gupta, Ole-Christoffer Granmo, and Ashok Agrawala. Successive reduction of arms in multi-armed bandits. In *Research and Development in Intelligent Systems XXVIII*, pages 181–194. Springer, 2011.

Delbert Hawkins, Roger J Best, and Kenneth A Coney. *Consumer behavior*. McGraw-Hill Publishing, 2009.

Eshcar Hillel, Zohar S Karnin, Tomer Koren, Ronny Lempel, and Oren Somekh. Distributed exploration in multi-armed bandits. In *Neural Information Processing Systems*, pages 854–862, 2013.

Pooria Joulani, Andras Gyorgy, and Csaba Szepesvári. Online learning under delayed feedback. In *International Conference on Machine Learning*, pages 1453–1461, 2013.

Varun Kanade and Thomas Steinke. Learning hurdles for sleeping experts. *ACM Transactions on Computation Theory*, 6(3):11, 2014.

Sean HK Kang. Spaced repetition promotes efficient and effective learning: Policy implications for instruction. *Policy Insights from the Behavioral and Brain Sciences*, 3(1):12–19, 2016.

Zohar Karnin, Tomer Koren, and Oren Somekh. Almost optimal exploration in multi-armed bandits. In *International Conference on Machine Learning*, pages 1238–1246, 2013.

Robert Kleinberg, Alexandru Niculescu-Mizil, and Yogeshwer Sharma. Regret bounds for sleeping experts and bandits. *Machine Learning*, 80(2-3):245–272, 2010.

Thodoris Lykouris, Vahab Mirrokni, and Renato Paes Leme. Stochastic bandits robust to adversarial corruptions. In *ACM SIGACT Symposium on Theory of Computing*, pages 114–122. ACM, 2018.

Karen A Machleit and R Dale Wilson. Emotional feelings and attitude toward the advertisement: The roles of brand familiarity and repetition. *Journal of Advertising*, 17(3):27–35, 1988.

Odalric-Ambrym Maillard and Rémi Munos. Adaptive bandits: Towards the best history-dependent strategy. In *Artificial Intelligence and Statistics*, pages 570–578, 2011.

Vianney Perchet, Philippe Rigollet, Sylvain Chassang, Erik Snowberg, et al. Batched bandit problems. *Annals of Statistics*, 44(2):660–681, 2016.
Ciara Pike-Burke, Shipra Agrawal, Csaba Szepesvari, and Steffen Grunewalder. Bandits with delayed, aggregated anonymous feedback. In *International Conference on Machine Learning*, pages 4102–4110, 2018.

Virag Shah, Jose Blanchet, and Ramesh Johari. Bandit learning with positive externalities. *ArXiv preprint arXiv:1802.05693*, 2018.

Pannagadatta Shivaswamy and Thorsten Joachims. Multi-armed bandit problems with history. In *Artificial Intelligence and Statistics*, pages 1046–1054, 2012.

Michael R Solomon, Dahren William Dahl, Katherine White, Judith L Zaichkowsky, and Rosemary Polegato. *Consumer behavior: Buying, having, and being*, volume 10. Pearson London, 2014.

Adith Swaminathan and Thorsten Joachims. Counterfactual risk minimization: Learning from logged bandit feedback. In *International Conference on Machine Learning*, pages 814–823, 2015.

István Szita and Csaba Szepesvári. Agnostic KWIK learning and efficient approximate reinforcement learning. In *Conference on Learning Theory*, pages 739–772, 2011.

Michal Valko, Rémi Munos, Branislav Kveton, and Tomáš Kocák. Spectral bandits for smooth graph functions. In *International Conference on Machine Learning*, pages 46–54, 2014.

Kuang Xu and Se-Young Yun. Reinforcement with fading memories. In *ACM Conference on Measurement and Modeling of Computer Systems*, pages 90–92. ACM, 2018.