On Interruptible Pure Exploration in Multi-Armed Bandits*

Alexander Shleyfman
Technion - Israel Institute of Technology,
Haifa, Israel

Antonín Komenda
Dept. of Computer Science,
Faculty of Electrical Engineering,
Czech Technical University,
Prague, Czech Republic

Carmel Domshlak
Technion - Israel Institute of Technology,
Haifa, Israel

Abstract

Interruptible pure exploration in multi-armed bandits (MABs) is a key component of Monte-Carlo tree search algorithms for sequential decision problems. We introduce Discriminative Bucketing (DB), a novel family of strategies for pure exploration in MABs, which allows for adapting recent advances in non-interruptible strategies to the interruptible setting, while guaranteeing exponential-rate performance improvement over time. Our experimental evaluation demonstrates that the corresponding instances of DB favorably compete both with the currently popular strategies UCB1 and ε-Greedy, as well as with the conservative uniform sampling.

Introduction

The decision setting known as Multi-Armed Bandit (MAB) captures the structure of some of the most fundamental problems in the field of decision making (Robbins 1952). In particular, in the stochastic Multi-Armed Bandit (MAB) setting, a player is given K actions $A := \{a_1, ..., a_K\}$. Each action $a_i \in A$ is associated with a reward, which is a random variable bounded in the interval $[0, 1]$ with expectation μ_i. The highest expected reward among the actions is denoted by μ^*, and the (for simplicity assumed to be single) action having this expectation is denoted by a^*. The player can make some T samples of the actions, but he does not know the expectations of the action rewards.

One of the canonical decision problems in the context of MABs is the problem of identifying an as good as possible action from A. In this problem, the player is not judged by the quality of the rewards collected by his samples, but only by the quality of the action which he recommends at the end of the process. With the recent rise of Monte-Carlo tree search (MCTS) algorithms for online planning of sequential decisions, this MAB problem receives a substantial attention in AI literature (Browne et al. 2012). This is because the nodes of a state-space Monte-Carlo search tree can be seen as a hierarchy of MAB players, and thus, having more effective sampling strategies for such pure exploration in MABs allows for devising more effective MCTS algorithms (e.g., see the analysis/discussion by Feldman and Domshlak, 2014).

Sampling strategies for pure exploration in MABs can be roughly divided into contract strategies and interruptible strategies. In the contract setting, the player is allowed exactly T samples up front, and thus his sampling strategy should aim at optimizing the quality of the recommendation that can be guaranteed within these T samples. In the interruptible setting, the player has no knowledge of T and can be interrupted at any time. Thus, the player here should optimize the rate at which his recommendation improves over time.

As of recent, certain contract strategies for pure exploration in MABs have been shown achieving guarantees that are optimal up to logarithmic factors (Audibert, Bubeck, and Munos 2010; Karnin, Koren, and Somekh 2013). A characteristic property of these strategies is that they incrementally zoom in on smaller and smaller subsets of actions. Unfortunately, contract strategies are not generally applicable in the context of MCTS because the players corresponding to the internal nodes of the search tree cannot be meaningfully preallocated with sampling budgets. In contrast, among the interruptible strategies (that all fit the context of MCTS), works on MCTS typically adopt the UCB1 strategy, which incorporates a soft zooming-in mechanism, but guarantees only polynomial-rate performance improvement over time (Bubeck, Munos, and Stoltz 2011). At the same time, a simple uniform sampling guarantees an exponential-rate performance improvement over time (Bubeck, Munos, and Stoltz 2011), yet it appears to be less popular in practice because it completely ignores the information collected about the actions over time.

In this work we introduce and study Discriminative Bucketing (DB), a family of interruptible strategies for pure exploration in MABs that allows for a flexible zooming-in on subsets of actions while guaranteeing exponential-rate performance improvement over time. We show that DB straightforwardly generalizes both uniform and ε-greedy sampling, but also, and more importantly, that certain instances of DB can be seen as interruptible variants of the state-of-the-art, action rejecting contract strategies. Our ex-

[*] This work is partially supported by and carried out at the Technion-Microsoft Electronic Commerce Research Center, as well as partially supported by the Air Force Office of Scientific Research, USAF, under grant number FA8655-12-1-2096.

perimental evaluation demonstrates that these novel interruptible strategies for pure exploration in MABs favorably compete both with the currently popular strategies UCB1 and ε-Greedy, as well as with the conservative Uniform sampling.

Pure Exploration in MABs

Originally, the MAB environment was mostly studied in the setting of "learning while acting", in which the player actually *experiences* the T sampled rewards (Robbins 1952; Lai and Robbins 1985). In this setting, the player is interested in maximizing his expected cumulative reward, or, conversely, in minimizing his *cumulative regret* from not sampling T times the best action a^* (as if μ^* is revealed to the player at the end of the process). Various strategies for action sampling in "learning while acting" have been proposed over the years. In particular, it is now known that the player can achieve a (close to optimal) logarithmic regret by adopting the deterministic UCB1 strategy (Auer, Cesa-Bianchi, and Fischer 2002), in which, taking advice from the Hoeffding's tail inequality (Hoeffding 1963), the player at time $t + 1$ samples the action

$$\operatorname*{argmax}_{a_i} \left[\widehat{\mu}_i + \sqrt{\frac{2 \log t}{t_i}} \right],$$

where t_i is the number of times the action a_i was sampled so far and $\widehat{\mu}_i$ is the current empirical mean of a_i.

In contrast, in the setting of pure exploration that is of interest here, the player only *observes* the T sampled rewards, and is then asked to output a recommendation, formed by a probability distribution over the action. Here, the player is only interested in identifying the best action, or, in more flexible terms, in minimizing his *simple regret* from not recommending a^* (as if suffering the difference between μ^* and the expected reward of the recommended action). As of recent, devising good sampling strategies for pure exploration in MABs is drawing an increasing attention, in particular because of their key role in devising efficient Monte-Carlo tree search (MCTS) algorithms for online planning in sequential decision problems such as Markov decision processes, multi-player turn games, etc. (Browne et al. 2012; Tolpin and Shimony 2012; Feldman and Domshlak 2014).

Adopting terminology from the literature on anytime algorithms (Zilberstein 1993), sampling strategies that assume knowledge of the sampling budget T are henceforth called *contract* strategies, while strategies that make no such assumptions are called *interruptible*. Considering interruptible strategies, Bubeck, Munos, and Stoltz (2011) showed that uniform sampling of the actions achieves exponential-rate reduction of simple regret over time, while UCB1 achieves only polynomial-rate reduction of this measure. Subsequently, Tolpin and Shimony (2012) showed that the popular ε-Greedy strategy, sampling the empirically best action with probability ε, while, with probability $1 - \varepsilon$, sampling an action drawn from A uniformly at random, also achieves exponential-rate reduction of simple regret over time for any $\frac{1}{\varepsilon} = O(1)$. The attractiveness of the ε-Greedy's convergence-rate guaranty relatively to that of Uniform depends much on the fit between the action set A and the choice of ε, which, of course, is not known a priori.

While interruptible strategies can obviously be applied in the contract setting as well, specialized algorithms for the contract setting have been developed to exploit the knowledge of the sample budget T.[1] In particular, Audibert, Bubeck, and Munos (2010) introduced the Successive Reject (SReject) strategy, in which the T samples are divided into $K - 1$ successive epochs, within each epoch the actions are sampled evenly, and after each epoch, the empirically worst action is ruled out. Audibert et al. proved the optimality of SReject up to logarithmic factors, and demonstrated its effectiveness in simulations. Recently, Karnin et al. (2013) examined another strategy, called Sequential Halving (SHalve), which also gradually rules actions out but does it differently from SReject: the T samples are split evenly across $\log_2 K$ epochs, within each epoch the actions are sampled evenly, and at the end of each epoch, the empirically worst half of the actions are ruled out. Karnin et al. proved the optimality of SHalve up to doubly-logarithmic factors, and demonstrated its competitiveness with SReject in simulations.

Returning now to the interruptible strategies, note that the formal superiority of Uniform/ε-Greedy over UCB1 is not entirely unexpected: The exploitative sampling of UCB1 has no direct motivation in the purely explorative setting, while Uniform dedicates no samples to exploitation at all. A radical conclusion from this result of Bubeck et al. (2011) would be that, unless the player is given some extra information about the action set A, the less his sampling strategy exploits the empirical knowledge, the better. Note, however, that the (close to) optimal contract strategies SReject and SHalve do exploit the empirical knowledge collected over time by gradually ruling out empirically worst actions and focusing the sampling on the more promising candidates. Furthermore, UCB1 actually outperforms Uniform, both formally and empirically, under restrictive sample allowances (Bubeck, Munos, and Stoltz 2011; Feldman and Domshlak 2014). Thus, after all, strategies for the interruptible setting probably should somehow take the empirical knowledge into account, and the question is, of course, how. This is precisely the question we consider in what comes next.

Discriminative Bucketing

The key conceptual difference between the (close to) optimal strategies for pure exploration in the interruptible and the contract settings appears to be that the latter incrementally shrink the set of candidates for recommendation. Unfortunately, if we are not ready to give up on the eventual

[1] The other way around, the player can be constrained with the required quality of the recommendation, and then he should minimize the number of samples required to stand by the requested quality guarantees. This setup gave rise to a family of PAC (probably approximately correct) algorithms for MABs (Even-Dar, Mannor, and Mansour 2002; Mannor and Tsitsiklis 2011), but these are less relevant to our work here.

```
**Discriminative Bucketing** (DB)
Parameters: A probability distribution $\mathcal{P}$ over the buckets $[\kappa]$;
            A bucketing function $\mathbf{b} : [K] \times [0,1]^K \to [\kappa]$, which, given a set of empirical means for actions in $A$,
                maps each action to the nesting-wise first bucket containing it;
While not interrupted, for each iteration $t = 1, 2, \ldots$
    If $t \leq K$, then sample $a_t$, initialize $\widehat{\mu}_t$ with the sampled reward, and proceed to the next round.
    Otherwise,
        – select bucket $B_j$ with probability $\mathcal{P}(j)$;
        – select an action $a_i$ from the bucket $B_j$, either by selecting one of the least sampled actions in $B_j$,
              or by selecting an action from $B_j$ uniformly at random;
        – sample $a_i$ and update $\widehat{\mu}_i$ with the sampled reward.

Recommend $a \in \operatorname{argmax}_{a_i} \widehat{\mu}_i$, i.e., recommend one of the empirically best actions (ties broken is some way).
```

Figure 1: The Discriminative Bucketing sampling scheme

convergence of the simple regret to zero, then ruling actions out over time cannot be used in the interruptible setting: No matter how many times each action has been sampled so far, even the currently empirically worst action still can be recommended at the end, if T will turn out to be large enough to allow a sufficient exploration of that action. At the same time, in contrast to Uniform, both ε-Greedy and UCB1 appear to *dynamically discriminate* actions based on their empirical means, biasing exploration towards the currently more promising actions. In terms of convergence-rate guarantees, that bias of UCB1 appears to be too strong, resulting in only a polynomial-rate reduction of simple regret over time. In contrast, the bias of ε-Greedy is only constant-multiplicative, keeping it close enough to Uniform to preserve exponential-rate reduction over time. It is not clear, however, what precisely justifies ε-Greedy's equal discrimination of all but the empirically best action in the context of pure exploration.

We now introduce Discriminative Bucketing (DB), a parametrized family of interruptible strategies for simple regret minimization in MABs, which allows for a flexible dynamic discrimination of actions while guaranteeing exponential-rate reduction of simple regret over time. In particular, we show that DB straightforwardly generalizes both Uniform and ε-Greedy. More importantly, certain instances of DB can be seen as interruptible variants of the action rejecting contract strategies, and later we demonstrate the effectiveness of these instances of DB in simulations.

The DB strategy scheme is depicted in Figure 1; there and henceforth, $[n]$ for $n \in \mathbb{N}$ denotes the set $\{1, \ldots, n\}$. As any other interruptible strategy for MABs, at each iteration DB samples an action a_i from A and updates its empirical mean $\widehat{\mu}_i$. Once interrupted, DB recommends one of the actions with the highest empirical mean. The selection of the actions for sampling is based on a set of $\kappa \leq K$ *nested buckets* of actions

$$B_1 \subset B_2 \subset \cdots \subset B_{\kappa-1} \subset B_\kappa = A,$$

and a positive probability distribution $\mathcal{P}$ over these buckets.

While the content of all buckets but the "all-inclusive bucket" B_κ can change between the iterations, the sizes of the buckets, as well as the probability distribution $\mathcal{P}$ over them, remain fixed. At each iteration, the actions are redistributed among the buckets based on their current empirical means $\widehat{M} = \{\widehat{\mu}_1, \ldots, \widehat{\mu}_K\}$. Recalling that our buckets are nested, let $a_{(1)}, \ldots, a_{(K)}$ be a relabeling of the actions in a non-decreasing order of their empirical means. For $i \in [K]$, we have $a_{(i)} \in B_j$ iff $i \leq |B_j|$. In Figure 1, this dynamic assignment of actions to buckets is captured by a function $\mathbf{b}$ that, given empirical means of K actions, maps a given action to the index of the first bucket that contains it; since the buckets are nested, if $a_i \in B_j$, then $a_i \in B_l$ for all $l \geq j$.

Given that, at each iteration DB samples the reward of an action selected by

1. sampling the set of buckets according to $\mathcal{P}$, and then
2. either selecting one of the least sampled actions in the selected bucket, or sampling the selected bucket uniformly at random.

Different strategy instances DB($\mathbf{b}, \mathcal{P}$) vary in the structure of their bucket sets (captured by $\mathbf{b}$) and/or in their probability distributions over the buckets. Note that both Uniform and ε-Greedy can be seen as specific instances of DB:

- Uniform corresponds to the (only) instance of DB with $\kappa = 1$, i.e., with a single bucket $B_1 = A$, and
- ε-Greedy is an instance of DB with $\kappa = 2$, $|B_1| = 1$ (and $|B_2| = K$), and $\mathcal{P}(1) = \varepsilon$ (respectively, $\mathcal{P}(2) = 1 - \varepsilon$).

More importantly, however, as we show below, (i) not only these two, but any instance of the DB strategy provides an exponential-rate reduction of simple regret over time, and (ii) the flow of DB allows to adapt the action rejection approach of the state-of-the-art contract strategies to the interruptible setting.

Proposition 1 *Every instance* DB($\mathbf{b}, \mathcal{P}$) *of* DB *ensures that, at any iteration* $t \geq K$, *the expected simple regret associated with the recommended action is upper-bounded by* $\alpha e^{-\beta t}$ *for some (independent of* t*) parameters* $\alpha, \beta > 0$.

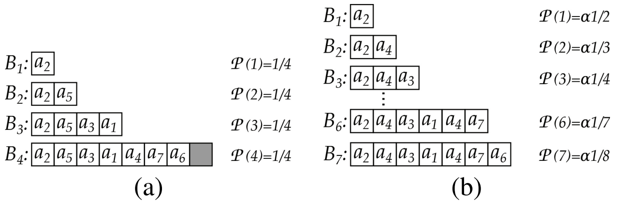


Figure 2: Example of bucketing $K = 7$ actions in (a) $\mathsf{DB}^{\mathsf{SH}}$ and (b) $\mathsf{DB}^{\mathsf{SR}}$

Proof sketch: Since $\mathcal{P}$ is a positive probability distribution and the largest bucket of actions B_κ consists of the entire set of actions A, each action is expected to be sampled by $\mathsf{DB}(\mathbf{b}, \mathcal{P})$ at least $t\frac{\mathcal{P}(\kappa)}{K}$, that is, $\Theta(t)$, times. The claim follows from that, combined with the exponential-rate convergence of simple regret ensured by Uniform (Bubeck, Munos, and Stoltz 2011). ■

We now introduce two novel instances of DB, baptized as $\mathsf{DB}^{\mathsf{SH}}$ and $\mathsf{DB}^{\mathsf{SR}}$, which can be seen as interruptible versions of the SHalve strategy of Karnin et al. (2013) and the SReject strategy of Audibert et al. (2010), respectively. Specifically:

$\mathsf{DB}^{\mathsf{SH}}$**:** While in SHalve, the $\lceil \log_2 K \rceil$ sampling epochs all have the same duration and each epoch rules out *half* of the actions, in $\mathsf{DB}^{\mathsf{SH}}$, the K actions fill $\lceil \log_2 K \rceil + 1$ *exponentially* growing buckets, with $|B_i| = 2^{i-1}$, and the probability distribution over the buckets is uniform, i.e., $\mathcal{P}(i) = \frac{1}{\kappa}$ (see Figure 2a).

$\mathsf{DB}^{\mathsf{SR}}$**:** In SReject, the duration of the sampling epochs grows linearly, with the duration of the epoch i being proportional to $\frac{1}{K-i+1}$, and each epoch rules out a single action. Reflecting that, the K actions in $\mathsf{DB}^{\mathsf{SR}}$ (see Figure 2b) fill K buckets, with $|B_i| = i$, and the probability distribution over the buckets is (linear) $\mathcal{P}(i) = \alpha\frac{1}{i+1}$, with α being the respective normalization factor.

Note that the general principle behind deriving both $\mathsf{DB}^{\mathsf{SH}}$ and $\mathsf{DB}^{\mathsf{SR}}$ is the same: The number of the sampling epochs in the contract algorithm translates to the number of buckets, the number of actions considered at the sampling epoch i translates to the size of the bucket $\kappa - i + 1$, and the duration of the sampling epoch i translates to the probability $\mathcal{P}(\kappa - i + 1)$ of sampling the bucket $\kappa - i + 1$. Likewise, reflecting the deterministically balanced action selection of SHalve and SReject within each sampling epoch, both $\mathsf{DB}^{\mathsf{SH}}$ and $\mathsf{DB}^{\mathsf{SR}}$ select one of the least sampled actions withing the selected bucket. In the next section, we put $\mathsf{DB}^{\mathsf{SH}}$ and $\mathsf{DB}^{\mathsf{SR}}$ on test in simulation.

Experimental Evaluation

To examine the prospects of $\mathsf{DB}^{\mathsf{SH}}$ and $\mathsf{DB}^{\mathsf{SR}}$, we conducted a few simple experiments and used them to compare the performance of $\mathsf{DB}^{\mathsf{SH}}$ and $\mathsf{DB}^{\mathsf{SR}}$ to the performance of UCB1, Uniform, and ε-Greedy[2].

The first set of seven experiments was borrowed from Audibert et al. (2010). These experiments all use Bernoulli distributions, all have $\mu^* = 0.5$, but correspond to different situations for the gaps between the action expected rewards, differing in the size of the gaps and the distribution of the gaps (clustered in few groups or distributed according to an arithmetic or geometric progression). Specifically:

expA1 A single cluster of bad actions: $K = 20$, $\mu_{2:20} = 0.4$ (meaning for any $j \in \{2, \dots, 20\}$, $\mu_j = 0.4$).

expA2 Two clusters of bad actions: $K = 20$, $\mu_{2:6} = 0.42$, $\mu_{7:20} = 0.38$.

expA3 Geometric progression: $K = 4$, $\mu_i = 0.5 - (0.37)^i$ for $i \in \{2, 3, 4\}$.

expA4 Six actions divided in three clusters: $K = 6$, $\mu_2 = 0.42$, $\mu_{3:4} = 0.4$, $\mu_{5:6} = 0.35$

expA5 Arithmetic progression: $K = 15$, $\mu_i = 0.5 - 0.025i$ for $i \in \{2, \dots, 15\}$

expA6 Two good arms and a large group of bad arms: $K = 20$, $\mu_2 = 0.48$, $\mu_{3:20} = 0.37$

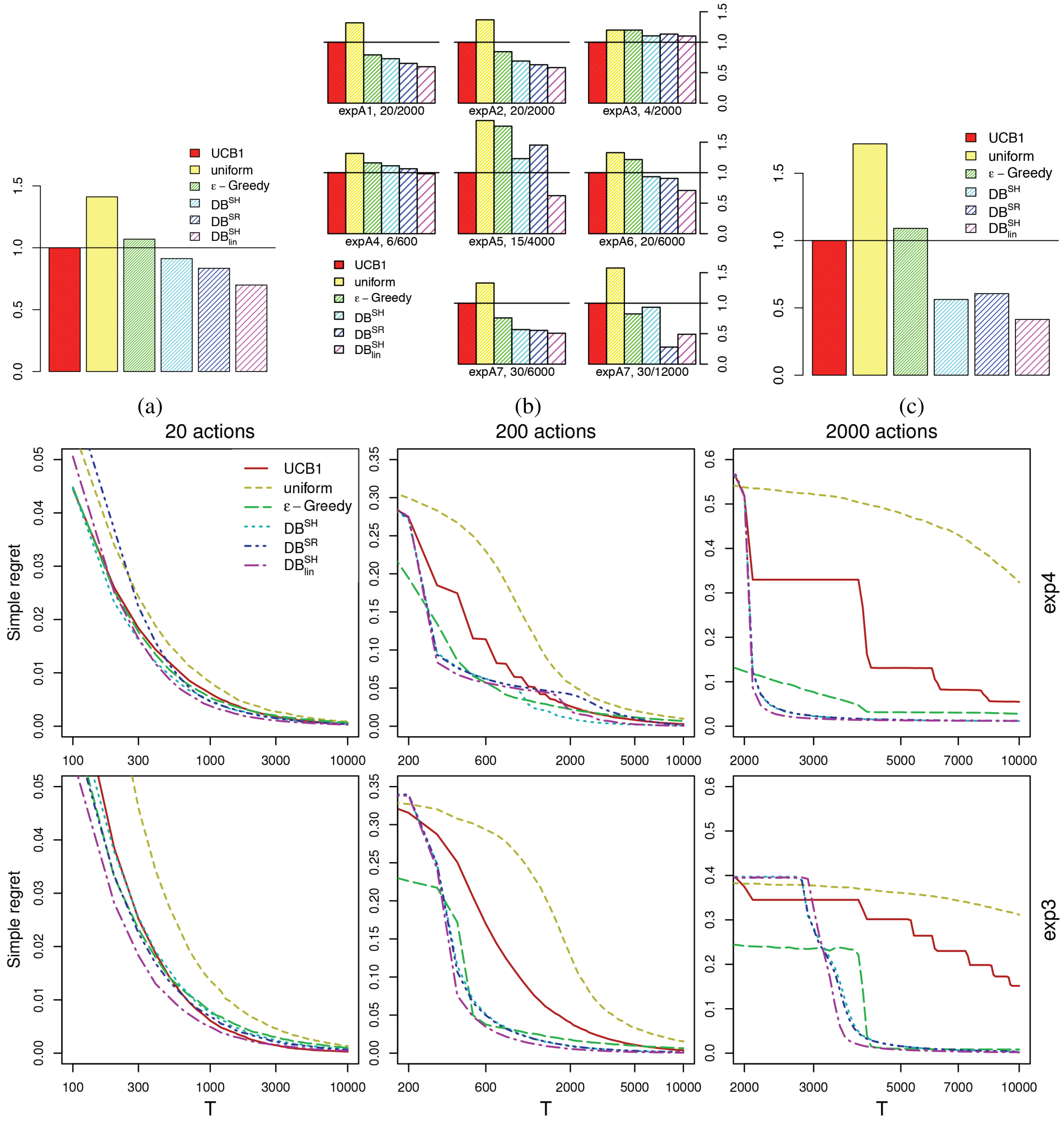
expA7 Three groups of bad arms: $K = 30$, $\mu_{2:6} = 0.45$,$\mu_{7:20} = 0.43$, $\mu_{21:30} = 0.38$

Each experiment was repeated 10000 times for a specific (but, of course, unknown to the algorithms) number of iterations T that was suggested by Audibert et al. (2010) for examining SReject in the respective "situation of gaps", with the last experiment being examined under two different values of T.

Figures 3a and 3b depict the performance of all the aforementioned sampling strategies (plus an additional strategy, $\mathsf{DB}^{\mathsf{SH}}_{\mathrm{lin}}$, that we discuss below) on the eight ($7 \times 1 + 1 \times 2$) experiments from Audibert et al. (2010) in terms of the relative achievements of these strategies with respect to UCB1, which is taken as a baseline. That is, if the simple regret achieved by a strategy A is lower (= better) than that achieved by UCB1, then A is scored $1 + \frac{(\mathsf{A}-\mathsf{UCB1})}{\mathsf{UCB1}}$, otherwise it is scored $1 + \frac{(\mathsf{A}-\mathsf{UCB1})}{\mathsf{A}}$. Figure 3a depicts the scores averaged over all the eight experiments, while Figure 3b depicts the scores per experiment, with labels K/T under the bar charts in Figure 3b capturing the number of actions K and the number of iterations T in the respective experiments. One can notice that

1. Uniform in these experiments was consistently the worst performer, probably suggesting that the situations of gaps examined by Audibert et al. (2010) are not confusing enough to justify the most conservative, uniform exploration of the actions.
2. Both $\mathsf{DB}^{\mathsf{SH}}$ and $\mathsf{DB}^{\mathsf{SR}}$ almost consistently outperformed ε-Greedy, with the sole exception being $\mathsf{DB}^{\mathsf{SH}}$ on **expA7** with $T = 12000$.
3. While in experiments with relatively small number of actions, namely **expA3**, **expA4**, and **expA5**, UCB1's simple

[2] Following Tolpin and Shimony (2012), we used ε-Greedy with $\varepsilon = 1/2$.



(a)

(b)

(c)

(d)

Figure 3: Results of the simulations: (a) Summary and (b) per-experiment results on the MAB setups by Audibert et al. (2010), (c) summary of the results on our additional experiments on MABs with both small and large numbers of actions, and (d) a detailed snapshot of the results summarized in (c).

regret was lower than this achieved by the DB instances, in experiments with relatively high number of actions, the picture was the opposite.

4. On average across the experiments, while UCB1 outperformed Uniform and ε-Greedy, $\mathsf{DB}^{\mathsf{SH}}$ and $\mathsf{DB}^{\mathsf{SR}}$ outperformed UCB1, reducing the simple regret achieved by the latter by 9% and 17%, respectively.

Interestingly, the best performer in this set of experiments was a strategy referred to in Figure 3 as $\mathsf{DB}^{\mathsf{SH}}_{\mathrm{lin}}$. This instance of DB borrows from both $\mathsf{DB}^{\mathsf{SH}}$ and $\mathsf{DB}^{\mathsf{SR}}$, yet its contract-setting counterpart has not been considered in the literature. Specifically, similarly to $\mathsf{DB}^{\mathsf{SH}}$, $\mathsf{DB}^{\mathsf{SH}}_{\mathrm{lin}}$ uses $\lceil \log_2 K \rceil + 1$ exponentially growing buckets, with $|B_i| = 2^{i-1}$. However, similarly to $\mathsf{DB}^{\mathsf{SR}}$, the probability distribution over the buckets is linear, with $\mathcal{P}(i) = \alpha \frac{1}{i+1}$. Informally, $\mathsf{DB}^{\mathsf{SH}}_{\mathrm{lin}}$ can be seen as combining the (relative) "aggressiveness" of discrimination of the $\mathsf{DB}^{\mathsf{SH}}$'s bucketing structure with the, again, relative, "aggressiveness" of discrimination of the $\mathsf{DB}^{\mathsf{SR}}$'s preference structure over buckets.

As it can be seen in Figure 3b, $\mathsf{DB}^{\mathsf{SH}}_{\mathrm{lin}}$ almost consistently outperformed UCB1 across the experiments, with the average reduction of the simple regret being 30%. In fact, $\mathsf{DB}^{\mathsf{SH}}_{\mathrm{lin}}$ almost consistently outperformed all the other strategies, across all the experiments, with the only exceptions being $\mathsf{DB}^{\mathsf{SR}}$ on, again, **expA7** with $T = 12000$, and UCB1 on **expA7**.

Now, to examine the effect of the growing number of actions on the relative performance of the different strategies, as well as to expand the evaluation beyond Bernoulli distributions, we conducted four large sets of experiments, each consisting of 10000 instances for each number of actions $K \in \{20, 200, 2000\}$. All the instances were generated randomly as described below; in what follows, the operation of drawing a sample from a uniform distribution over the interval $[x, y]$ is denoted by $\sim \mathcal{U}[x, y]$.

exp1 Bernoulli: For $i \in [K]$, the action a_i is set to a Bernoulli random variable $\mathrm{Ber}(p_i)$ with parameter $p_i \sim \mathcal{U}[0, 1]$.

exp2 Bernoulli with random reward: For $i \in [K]$, the action a_i is set to $r_i \cdot \mathrm{Ber}(p_i)$, where with $r_i \sim \mathcal{U}[0, 1]$ and $p_i \sim \mathcal{U}[0, 1]$.

exp3 Composition of two Bernoulli distributions: For $i \in [K]$, the action a_i corresponds to a probability distribution (p_i^1, p_i^2, p_i^3) over rewards $(1, 0.5, 0)$, with the normalized parameters p_i^1, p_i^2, p_i^3 being first drawn uniformly at random.

exp4 Bernoulli with heavy noise: the action a_i corresponds to a probability distribution $(p_i^1, \ldots, p_i^{s_i})$ over $(r_i^1, \ldots, r_i^{s_i})$ with the number of outcomes s_i being drawn uniformly from $\{2, \ldots, 200\}$, normalized distribution parameters $p_i^j \sim \mathcal{U}[0, 1]$, a good outcome $r_i^1 \sim \mathcal{U}[0.5, 1]$, and, for $j > 1$, noise outcomes $r_i^j \sim \mathcal{U}[0, 0.05]$.

Figure 3c summarizes the average performance of the different strategies relatively to the baseline UCB1 after 10000 samples. The performance is averaged over all the MAB instances from all the four experiments. Interestingly, the picture here is qualitatively very similar to the summary of our results in the experimental setups of Audibert et al. (2010), with the advantage of the contract-inspired DB instances being stratified here even further: The average reduction of simple regret by $\mathsf{DB}^{\mathsf{SH}}$, $\mathsf{DB}^{\mathsf{SR}}$, and $\mathsf{DB}^{\mathsf{SH}}_{\mathrm{lin}}$ relatively to UCB1 was 44%, 40%, and 59%, respectively. A closer inspection of the simulation results reveals that the relative advantage of these three DB instances grows sharply with the increase in the number of actions. Figure 3d depicts the evolution of the performance of the different strategies over time (= samples) in the **exp3** and **exp4** experiments, averaged separately over the instances with 20, 200, and 2000 actions.[3] It is easy to see from this illustration that, while the qualitative difference between all the strategies on MABs with 20 actions was not that substantial and they all home in rather quickly on some very good action candidates, when the number of actions grows, the advantage of the contract-inspired instances of DB becomes very substantial: On MABs with 2000 actions, the average reduction of simple regret by $\mathsf{DB}^{\mathsf{SH}}$, $\mathsf{DB}^{\mathsf{SR}}$, and $\mathsf{DB}^{\mathsf{SH}}_{\mathrm{lin}}$ relatively to UCB1 was 87%, 87%, and 90%, respectively.

Summary

We showed how the principle of incremental action rejection from state-of-the-art contract algorithms for pure exploration in MABs can be adapted to interruptible pure exploration via a notion of dynamic bucketing (DB). The resulting interruptible sampling strategies were show guaranteeing exponential-rate performance improvement over time, while favorably competing with the popular UCB1 strategy, as well as with the more conservative ε and uniform sampling. We see this work as a step towards a comprehensive understanding of what is the "right thing to do" when one needs an interruptible mechanism for pure exploration in MABs.

Given the results of our comparative experimentation, a few questions in particular call for our attention. First, the exponential bounds offered by a simple proof strategy we used above are inherently loose. Thus, it is possible that stronger formal claims on the performance of various instances of DB could be derived, offering a better differentiation between them. Second, given the superb performance of $\mathsf{DB}^{\mathsf{SH}}_{\mathrm{lin}}$ in our experiments, it will be interesting to see whether and/or when its contact counterpart will be formally competitive with the currently leading contract strategies SReject and SHalve. Finally, from a practical perspective, a natural next step would be to examine the effectiveness of DB within the MCTS-based solvers for Markov decision processes and the MCTS-based solvers for adversarial turn games with large branching factors such as Go, or even Starcraft.

References

Audibert, J.-Y.; Bubeck, S.; and Munos, R. 2010. Best arm identification in multi-armed bandits. In *COLT*, 41–53.

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

Browne, C.; Powley, E. J.; Whitehouse, D.; Lucas, S. M.; Cowling, P. I.; Rohlfshagen, P.; Tavener, S.; Perez, D.; Samothrakis, S.; and Colton, S. 2012. A survey of Monte-Carlo tree search methods. *IEEE Trans. on Comp. Intell. and AI in Games* 143.

Bubeck, S.; Munos, R.; and Stoltz, G. 2011. Pure exploration in finitely-armed and continuous-armed bandits. *Theor. Comput. Sci.* 412(19):1832–1852.

Even-Dar, E.; Mannor, S.; and Mansour, Y. 2002. PAC bounds for multi-armed bandit and Markov decision processes. In *COLT*, 255–270.

[3] Due to space consideration, **exp3** and **exp4** have been selected for this detailed illustration as the two experiments that, after 10000 iterations, exhibited the largest, respectively the smallest, difference in simple regret between $\mathsf{DB}^{\mathsf{SH}}$ and UCB1.

Feldman, Z., and Domshlak, C. 2014. On MABs and separation of concerns in Monte-Carlo planning for MDPs. In *ICAPS*.

Hoeffding, W. 1963. Probability inequalities for sums of bounded random variables. *J. American Stat. Ass.* 58(301):13–30.

Karnin, Z. S.; Koren, T.; and Somekh, O. 2013. Almost optimal exploration in multi-armed bandits. In *ICML*, 1238–1246.

Lai, T. L., and Robbins, H. 1985. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics* 6(1):4–22.

Mannor, S., and Tsitsiklis, J. N. 2011. Mean-variance optimization in Markov decision processes. In *ICML*, 177–184.

Robbins, H. 1952. Some aspects of the sequential design of experiments. *Bull. Amer. Math. Soc.* 58(5):527–535.

Tolpin, D., and Shimony, S. E. 2012. MCTS based on simple regret. In *AAAI*.

Zilberstein, S. 1993. *Operational Rationality through Compilation of Anytime Algorithms*. Ph.D. Dissertation, University of California at Berkeley.