Decoupling Exploration and Exploitation in Multi-Armed Bandits

Orly Avner ORLYKA@TX.TECHNION.AC.IL
Shie Mannor SHIE@EE.TECHNION.AC.IL
Department of Electrical Engineering, Technion

Ohad Shamir OHADSH@MICROSOFT.COM
Microsoft Research New England

Abstract

We consider a multi-armed bandit problem where the decision maker can explore and exploit different arms at every round. The exploited arm adds to the decision maker's cumulative reward (without necessarily observing the reward) while the explored arm reveals its value. We devise algorithms for this setup and show that the dependence on the number of arms, k, can be much better than the standard $\sqrt{k}$ dependence, depending on the behavior of the arms' reward sequences. For the important case of piecewise stationary stochastic bandits, we show a significant improvement over existing algorithms. Our algorithms are based on a non-uniform sampling policy, which we show is essential to the success of *any* algorithm in the adversarial setup. Finally, we show some simulation results on an ultra-wide band channel selection inspired setting indicating the applicability of our algorithms.

1. Introduction

Multi-armed bandits have long been a canonical framework for studying online learning under partial information constraints. In this framework, a learner has to repeatedly obtain rewards by choosing from a fixed set of k actions (arms), and gets to see only the reward of the chosen action. The goal of the learner is to minimize regret, namely the difference between her own cumulative reward and the cumulative reward of the best single action in hindsight. We focus here on algorithms suited for adversarial settings, which have reasonable regret even without any stochastic assumptions on the reward generating process.

A central theme in multi-armed bandits is the *exploration-exploitation tradeoff*: The learner must choose highly-rewarding actions most of the time in order to minimize regret, but also needs to do some exploration in order to determine which actions to choose. Ultimately, the tradeoff comes from the assumption that the learner is constrained to observe only the reward of the action she picked.

While being a compelling and widely applicable framework, there exist several realistic bandit-like settings, which do not correspond to this fundamental assumption. For example, in ultra-wide band (UWB) communications, the decision maker, also called the "secondary," has to decide in which channel to transmit and in what way. There are typically many possible channels (i.e., frequency bands) and several transmission methods (power, code used, modulation, etc.; see (Oppermann et al., 2004)). In some UWB devices, the secondary can sense a different channel (or channels) than the one it currently uses for transmission. In fact, in some settings, the secondary cannot sense the channel it is currently transmitting in because of interference. The UWB environment is extremely noisy since it potentially contains many other sources, called "primaries." Some of these sources are sources whose behavior (which channel they use, for how long, and in which power level) can be very hard to predict as they represent a mobile device using WiMAX, WiFI or some other communication protocol. It is therefore sensible to model the behavior of primaries as an adversarial process or a piecewise stationary process. We should mention that UWB networks are highly complex, with many issues such as power constraints and multi-agency that have been considered in the multi-armed bandit framework (Liu & Zhao, 2010; Avner & Mannor, 2011; Lai et al., 2008), but the decoupling of

Appearing in *Proceedings of the 29th International Conference on Machine Learning*, Edinburgh, Scotland, UK, 2012.

sensing and transmission has not been considered to the best of our knowledge. More abstractly, our work relates to any bandit-like setting, where we are free to query the environment for some additional partial information, irrespective of our actual actions.

In such settings, the assumption that the learner can only observe the reward of the action she picked is an unnecessary constraint, and one might hope that removing this constraint and constructing suitable algorithms would allow better performance. We emphasize that this is far from obvious: In this paper, we will mostly focus on the case where the learner may query just a single action, so in some sense the learner gets the same "amount of information" per round as the standard bandit setting (i.e., the reward of a single action out of k actions overall). The goal of this paper is to devise algorithms for this setting, and analyze theoretically and empirically whether the hope for improved performance is indeed justified. We emphasize that our results and techniques naturally generalize to cases where more than one action can be queried, and cases where the reward of the selected action is always revealed (see Sec. 7).

Specifically, our contributions are the following:

- We present a "decoupled" multi-armed bandit algorithm, which is suited to our setting. The algorithm is based on a certain querying distribution, which is adaptive and depends on the distribution by which the actions are actually picked. We show a "data-dependent" regret guarantee for the algorithm, which is never worse than that of standard bandit algorithms, and can be much better (in terms of dependence on the number of actions k), depending on how the actions' rewards behave.
- We prove that in certain settings (in particular, piecewise stochastic rewards), the decoupling assumption allows us to devise algorithms with significantly better performance than *any* possible standard bandit algorithm.
- Our algorithms are based on a certain adaptive querying distribution, in contrast to previous works in the stochastic case where the querying distribution was uniform. We show that in some sense, such an adaptive policy is *necessary* in an adversarial setting, in order to get performance improvements compared to standard bandit algorithms.
- We perform a preliminary experimental study, corroborating our theoretical findings and indicating that our algorithmic approach indeed leads to improved results, compared to standard approaches.

The proofs of our theorems are provided in the appendix of the full version (Avner et al., 2012).

Related Work. The idea of decoupling exploration and exploitation has appeared in a few previous works, but in different settings and contexts. For example, (Yu & Mannor, 2009) discuss a setting where the learner is allowed to query an additional action in a multi-armed bandit setting, but the focus there was on algorithms for stochastic bandits, as opposed to adversarial bandits as we do here. (Agarwal et al., 2010) study a bandit setting with (one or more) queries per round. However, they focus on the problem of bandit convex optimization, which is much more general than ours, and exploration and exploitation remains coupled in their framework. A different line of work ((Even-Dar et al., 2006; Audibert et al., 2010; Bubeck et al., 2011)) considers multi-armed bandits in a stochastic setting, where the goal is to identify the best action by performing pure exploration. While this work also conceptually "decouples" exploration and exploitation, the goal and setting are quite different than ours.

2. Problem Setting

We use $[k]$ as shorthand for $\{1, \ldots, k\}$. Bold-face letters represent vectors, and $\mathbf{1}_A$ represents the indicator function for an event A. We use the standard big-Oh notation $\mathcal{O}(\cdot)$ to hide constants, and $\tilde{\mathcal{O}}(\cdot)$ to hide constants and logarithmic factors. For a distribution vector $\mathbf{p}$ on the k-simplex, we use the notation

$$\|\mathbf{p}\|_{1/2} = \left(\sum_{j=1}^{k} \sqrt{p_j} \right)^2$$

to describe the '$\ell_{1/2}$'-norm of the distribution. It is straightforward to show that for a distribution vector, this quantity is always in $[1, k]$. In particular, it is k for the uniform distribution, and gets smaller the more non-uniform the distribution is, attaining the value of 1 when $\mathbf{p}$ is a unit vector.

Our setting is a variant of the standard adversarial multi-armed bandit framework, focusing (for simplicity) on an oblivious adversary and a fixed horizon. In this setting, we have a fixed set of $k > 1$ actions and a fixed known number of rounds T. Each action i at each round t has an unknown associated reward $g_i(t) \in [0, 1]$. At each round, a learner chooses one of the actions i_t, and obtains the associated reward $g_{i_t}(t)$. The basic goal in this setting is to minimize the

regret with respect to the best single action in hindsight, namely

$$\max_i \sum_{t=1}^{T} g_i(t) - \sum_{t=1}^{T} g_{i_t}(t).$$

Unless specified otherwise, we make no assumptions on how the rewards $g_i(t)$ are generated (other than boundedness), and they might even be generated adversarially by an agent with full knowledge of our algorithm. However, we assume that the rewards are fixed in advance and do not depend on the learner's (possibly random) choices in previous rounds.

In standard multi-armed bandits, at the end of each round, the learner only gets to know the reward $g_{i_t}(t)$ of the action i_t which was actually picked, but not the reward of other actions. Instead, in this paper we focus on a different setting, where the learner, after choosing an action i_t, may *query* a single action j_t and get to see its associated reward $g_{j_t}(t)$. This setting is a (slight) relaxation of the standard bandit setting, since we can always query $j_t = i_t$. However, here it is possible to query an action different than i_t. We emphasize that the regret is still measured with respect to the chosen actions i_t, and the querying only has informational value. In order to compare our results with those obtainable in the standard setting, we will use the term *standard bandit algorithm* to refer to algorithms which are not free to query rewards, and are limited to receiving the reward of the chosen action. A typical example is the EXP3.P (Auer et al., 2002), with a $\tilde{\mathcal{O}}(\sqrt{kT})$ regret upper bound, holding with high probability, or the Implicitly Normalized Forecaster of (Audibert & Bubeck, 2009) with $\mathcal{O}(\sqrt{kT})$ regret.

An interesting variant of our setting is when the learner gets to query more than one action, or gets to see $g_{i_t}(t)$ on top of $g_{j_t}(t)$. Such variants are further discussed in Sec. 7.

3. Basic Algorithm and Results

In analyzing our "decoupled" setting, perhaps the first question one might ask is whether one can *always* get improved regret performance, compared to the standard bandit setting. Namely, that for any reward assignment, the attainable regret will always be significantly smaller. Unfortunately, this is not the case: It can be shown that there exists an adversarial strategy such that the regret of standard bandit algorithms is $\tilde{\Theta}(\sqrt{kT})$, whereas the regret of any "decoupled" algorithm will be[1] $\Omega(\sqrt{kT})$. Therefore, one cannot hope to always obtain better performance. However, as we will soon show, this can be obtained under certain realistic conditions on the actions' rewards.

We now turn to present our first algorithm (Algorithm 1 below) and the associated regret analysis. The algorithm is rather similar in structure to standard bandit algorithms, picking actions at random in each round t according to a weighted distribution $\mathbf{p}(t)$ which is updated multiplicatively. The main difference is in determining how to query the reward. Here, the queried action is picked at random, according to a query distribution $\mathbf{q}(t)$ which is based on but not identical to $\mathbf{p}(t)$. More particularly, the queried action j_t is chosen with probability

$$q_{j_t}(t) = \frac{\sqrt{p_{j_t}(t)}}{\sum_{j=1}^{k} \sqrt{p_j(t)}}. \tag{1}$$

Roughly speaking, this distribution can be seen as a "geometric average" between $\mathbf{p}(t)$ and a uniform distribution over the k actions. See Algorithm 1 for the precise pseudocode.

Algorithm 1 Decoupled MAB Algorithm

Input: Step size parameter $\mu \in [1, k]$, confidence parameter $\delta \in (0, 1)$
Let $\eta = 1/\sqrt{\mu T}$, $\beta = 2\eta\sqrt{6\log(3k/\delta)}$ and $\gamma = \eta^2(1+\beta)^2k^2$
$\forall\, j \in [k]$ let $w_j(1) = 1$.
for $t = 1, \ldots, T$ **do**
  $\forall\, j \in [k]$, let $p_j(t) = (1-\gamma)\frac{w_j(t)}{\sum_{l=1}^{k} w_l(t)} + \frac{\gamma}{k}$
  Choose action i_t with probability $p_{i_t}(t)$
  Query reward $g_{j_t}(t)$ with probability
  $$q_{j_t}(t) = \frac{\sqrt{p_{j_t}(t)}}{\sum_j \sqrt{p_j(t)}}$$
  $\forall\, j \in [k]$, let $\tilde{g}_j(t) = \frac{1}{q_j(t)}\left(g_j(t)\mathbf{1}_{j_t=j} + \beta\right)$
  $\forall\, j \in [k]$, let $w_j(t+1) = w_j(t)\exp(\eta\tilde{g}_j(t))$
end for

Readers familiar with bandit algorithms might notice the existence of the common "exploration component" γ/k in the definition of $p_j(t)$. In standard bandit algorithm, this is used to force the algorithm to explore all arms to some extent. In our setting, exploration is performed via the separate query distribution $q_j(t)$, and in fact, this γ/k term can be inserted into the $q_j(t)$ definition instead. While this would be more aesthetically pleasing , it also seems to make our proofs

[1] One simply needs to consider the strategy used to obtain the $\Omega(\sqrt{kT})$ regret lower bound in the standard bandit setting (Auer et al., 2002). The lower bound proof can be shown to apply to a "decoupled" algorithm as well. Intuitively, this is because the hardness for the learner stems from distinguishing slightly different distributions based on at most T samples, which has nothing to do with the coupling constraint.

and results more complicated, without substantially improving performance. Therefore, we will stick with this formulation.

Before discussing the formal theoretical results, we would like to briefly explain the intuition behind this querying distribution. Most bandit algorithms (including ours) build upon a standard multiplicative updates approach, which updates the distribution $\mathbf{p}(t)$ multiplicatively based on each action's rewards. In the bandit setting, we only get partial information on the rewards, and therefore resort to multiplicative updates based on an unbiased estimate of them. The key quantity which controls the regret is the variance of these estimates, in expectation over the action distribution $\mathbf{p}(t)$. In our case, this quantity turns out to be on the order of $\sum_{j=1}^{k} p_j(t)/q_j(t)$. Now, standard bandit algorithms, which may not query at will, are essentially constrained to have $q_j(t) = p_j(t)$, leading to an expected variance of k and hence the k in their $\tilde{\mathcal{O}}(\sqrt{kT})$ regret bound. However, in our case, we are free to pick the querying distribution $\mathbf{q}(t)$ as we wish. It is not hard to verify that $\sum_{j=1}^{k} p_j(t)/q_j(t)$ is minimized by choosing $\mathbf{q}(t)$ as in Eq. (1), with the value of $\|\mathbf{p}(t)\|_{1/2}$. Thus, roughly speaking, instead of dependence on k, we get a dependence on $\frac{1}{T}\sum_{t=1}^{T}\|\mathbf{p}(t)\|_{1/2}$, as will be seen shortly.

The theoretical analysis of our algorithm relies on the following technical quantity: For any algorithm parameter choices μ, δ, and for any $v \in [1, k]$, define

$$P(v, \delta, \mu) = \Pr\left(\frac{1}{T}\sum_{t=1}^{T}\|\mathbf{p}(t)\|_{1/2} > v\right),$$

where the probability is over the algorithm's randomness, run with parameters μ, δ, with respect to the (fixed) reward sequence. The formal result we obtain is the following:

Theorem 1. *Suppose that T is sufficiently large (and thus η and β sufficiently small) so that $(1+\beta)^2 \leq 2$. Then for any $v \in [1, k]$, it holds that with probability at least $1 - \delta - P(v, \delta, \mu)$ that the sequence of rewards $g_{i_1}(1), \ldots, g_{i_T}(T)$ returned by Algorithm 1 satisfies*

$$\max_i \sum_{t=1}^{T} g_i(t) - \sum_{t=1}^{T} g_{i_t}(t) \leq \tilde{\mathcal{O}}\left(\sqrt{\left(\frac{v^2}{\mu} + \mu + v\right)T} + \frac{k^2}{\mu} + \frac{k^2}{T^{3/2}}\right)$$

where the $\tilde{\mathcal{O}}$ notation hides numerical constants and factors logarithmic in k and δ.

At this point, the nature of this result might seem a bit cryptic. We will soon provide more concrete examples, but would like to give a brief general intuition. First of all, if we pick $\mu = v = k$, then $P(v, \delta, \mu) = 0$ always (as $\|\mathbf{p}(t)\|_{1/2} \leq k$), and the bound becomes $\tilde{\mathcal{O}}(\sqrt{kT})$, holding with probability $1 - \delta$, similar to standard multi-armed bandit guarantees. This shows that our algorithm's regret guarantee is *never* worse than that of standard bandit algorithms. However, the theorem also implies that under certain conditions, the resulting bound may be significantly better. For example, if we run the algorithm with $\mu = 1$ and have $v = \mathcal{O}(1)$, then the bound becomes $\tilde{\mathcal{O}}\left(\sqrt{T}\right)$ for sufficiently large T. This bound is meaningful only if $P(\mathcal{O}(1), \delta, 1)$ is reasonably small. This would happen if the distribution vectors $\mathbf{p}(t)$ chosen by the algorithm tend to be highly non-uniform, since it leads to a small value for $\frac{1}{T}\sum_{t=1}^{T}\|\mathbf{p}(t)\|_{1/2}$.

We now turn to provide a concrete scenario, where the bound we obtain is better than those obtained by standard bandit algorithms. Informally, the scenario we discuss assumes that although there are k actions, where k is possibly large, only a small number of them are actually "relevant" and have a performance close to that of the best action in hindsight. Intuitively, such cases would lead to the distribution vectors $\mathbf{p}(t)$ to be non-uniform, which is favorable to our analysis.

Theorem 2. *Suppose that the reward of each action is chosen i.i.d. from a distribution supported on $[0, 1]$. Furthermore, suppose that there exist a subset $G \subset [k]$ of actions and a parameter $\Delta > 0$ (where $|G|, \Delta$ are considered constants independent of k, T), such that the expected reward of any action in G is larger than the expected reward of any action in $[k] \setminus G$ by at least Δ. Then if we run our algorithm with*

$$\mu = k^{\min\left\{1,\ \max\left\{0,\ \frac{4}{3} - \frac{1}{3}\log_k(T)\right\}\right\}},$$

it holds with probability at least $1 - \delta$ that the regret of the algorithm is at most

$$\tilde{\mathcal{O}}\left(\sqrt{k^{\max\{0, \frac{4}{3} - \frac{1}{3}\log_k(T)\}}T}\right),$$

where the $\tilde{\mathcal{O}}$ notation hides numerical constants and factors logarithmic in δ, k.

The bound we obtain interpolates between the usual $\tilde{\mathcal{O}}(\sqrt{kT})$ bound obtained using a standard bandit algorithm, and a considerably better $\tilde{\mathcal{O}}(\sqrt{T})$, as T gets larger compared with k. We note that a mathematically equivalent form of the bound is

$$\max\left\{\left(\frac{k}{T}\right)^{2/3}, \left(\frac{1}{T}\right)^{1/2}\right\}T.$$

Namely, the average per-round regret scales down as $(k/T)^{2/3}$, until T is sufficiently large and we switch to

a $(1/T)^{1/2}$ regime. In contrast, the bound for standard bandit algorithms is always of the form $(k/T)^{1/2}$, and the rate of regret decay is significantly slower.

We emphasize that although the setting discussed above is a stochastic one (where the rewards are chosen i.i.d.), our algorithm can cope simultaneously with arbitrary rewards, unlike algorithms designed specifically for stochastic i.i.d. rewards (which do admit better dependence in T, although not necessarily in k).

Finally, we note in practice, the optimal choice of μ depends on the (unknown) rewards, and hence cannot be determined by the learner in advance. However, this can be resolved algorithmically by a standard doubling trick (cf. (Cesa-Bianchi & Lugosi, 2006)), without materially affecting the regret guarantee. Roughly speaking, we can guess an upper bound v on $\frac{1}{T}\sum_{t=1}^{T}\|\mathbf{p}(t)\|_{1/2}$ and pick $\mu = v$, and if the cumulative sum $\sum\|\mathbf{p}(t)\|_{1/2}$ eventually exceeds Tv at some round, then we double v and μ and restart the algorithm.

4. Decoupling Provably Helps in some Adversarial Settings

So far, we have seen how the bounds obtained for our approach are better than the ones known for standard bandit algorithms. However, this doesn't imply that our approach would indeed yield better performance in practice: it might be possible, for instance, that for the setting described in Thm. 2, one can provide a tighter analysis of standard bandit algorithms, and recover a similar result. In this section, we show that there are cases where decoupling provably helps, and our approach can provide performance provably better than any standard bandit algorithm, for information-theoretic reasons. We note that the idea of decoupling has been shown to be helpful in cases reminiscent of the one we will be discussing (Yu & Mannor, 2009), but here we study it in the more general and challenging adversarial setting.

Instead of the plain-vanilla multi-armed bandit setting, we will discuss here a slightly more general setting, where our goal is not to achieve regret with respect to the best single action, but rather to the best sequence of $S > 1$ actions. More specifically, we wish to obtain a regret bound of the form

$$\max_{\substack{1=T_1\le T_2\le\ldots\le T_{S+1}=T\\ i^1,\ldots,i^S\in[k]}}\sum_{s=1}^{S}\sum_{t=T_s+1}^{T_{s+1}} g_{i^s}(t) - \sum_{t=1}^{T} g_{i_t}(t).$$

This setting is well-known in the online learning literature, and has been considered for instance in (Herbster & Warmuth, 1998) for full-information online learning (under the name of "tracking the best expert") and in (Auer et al., 2002) for the bandit setting (under the name of "regret against arbitrary strategies").

This setting is particularly suitable when the best action changes with time. Intuitively, our decoupling approach helps here, since we can exploit much more aggressively while still performing reasonable exploration, which is important for detecting such changes.

The algorithm we use follows the lead of (Auer et al., 2002) and is presented as Algorithm 2. The only difference compared to Algorithm 1 is that the $w_j(t+1)$ parameters are computed differently. This change facilitates more aggressive exploration.

Algorithm 2 Decoupled MAB Algorithm For Switching

Input: Step size parameter $\mu \in [1,k]$, confidence parameter $\delta \in (0,1)$, number of switches S
Let $\eta = \sqrt{S/\mu T}$, $\alpha = 1/T$, $\beta = 2\eta\sqrt{6\log(3k/\delta)}$ and $\gamma = \eta^2(1+\beta)^2k^2$
$\forall\, j \in [k]$ let $w_j(1) = 1$.
for $t = 1,\ldots,T$ **do**
  $\forall\, j \in [k]$, let $p_j(t) = (1-\gamma)\frac{w_j(t)}{\sum_{l=1}^{k} w_l(k)} + \frac{\gamma}{k}$
  Choose action i_t with probability $p_{i_t}(t)$
  Query reward $g_{j_t}(t)$ with probability $q_{j_t}(t) = \frac{\sqrt{p_{j_t}(t)}}{\sum_j \sqrt{p_j(t)}}$
  $\forall\, j \in [k]$, let $\tilde{g}_j(t) = \frac{1}{q_j(t)}\left(g_j(t)\mathbf{1}_{j_t=j} + \beta\right)$
  $\forall\, j \in [k]$, let $w_j(t+1) = w_j(t)\exp(\eta\tilde{g}_j(t)) + \frac{e\alpha}{k}\sum_{i=1}^{T} w_i(t)$
end for

The following theorem, which is proven along similar lines to Thm. 1, shows that in this setting as well, we get the same kind of dependence on the distribution vectors $\mathbf{p}(t)$ as in the standard bandit setting.

Theorem 3. *Suppose that T is sufficiently large (and thus η and β sufficiently small) so that $(1+\beta)^2 \le 2$. Then for any $v \in [1,k]$, it holds that with probability at least $1-\delta-P(v,\delta,\mu)$ that the sequence of rewards $g_{i_1}(1),\ldots,g_{i_T}(T)$ returned by algorithm 2 satisfies the following, simultaneously over all segmentations of $\{1,\ldots,T\}$ to S epochs and a choice of action*

i^s to each epoch:

$$\sum_{s=1}^{S}\sum_{t=T_s+1}^{T_{s+1}} g_{i^s}(t) - \sum_{t=1}^{T} g_{i_t}(t) \leq \tilde{\mathcal{O}}\left(\sqrt{S\left(\frac{v^2}{\mu}+\mu+v\right)T} + \frac{k^2}{\mu} + \frac{k^2}{T^{3/2}}\right).$$

The $\tilde{\mathcal{O}}$ notation hides numerical constants and factors logarithmic in k and δ.

In particular, we can also get a parallel version of Thm. 2, which shows that when there are only a small number of "good" actions (compared to k), the leading term has decaying dependence on k, unlike standard bandit algorithms where the dependence on k is always $\sqrt{k}$.

Theorem 4. *Suppose that the reward of each action is chosen i.i.d. from a distribution supported on $[0,1]$. Furthermore, suppose that at each epoch s, there exists a subset $G^s \subset [k]$ of actions and a parameter $\Delta > 0$ (where $|G^s|, \Delta$ are considered constants independent of k, T), such that the expected reward of any action in G^s is larger than the expected reward of any action in $[k]\setminus G^s$ by at least Δ. Then if we run Algorithm 2 with*

$$\mu = k^{\min\left\{1,\ \max\left\{0,\ \frac{4}{3}-\frac{1}{3}\log_k(T)\right\}\right\}},$$

it holds with probability at least $1-\delta$ that the regret of the algorithm is at most

$$\tilde{\mathcal{O}}\left(\sqrt{Sk^{\max\{0,\frac{4}{3}-\frac{1}{3}\log_k(T)\}}T}\right),$$

where the $\tilde{\mathcal{O}}$ notation hides numerical constants and factors logarithmic in δ and k.

Now, we are ready to present the main negative result of this section, which shows that in the setting of Thm. 2, *any* standard bandit algorithm cannot have a regret better than $\Omega(\sqrt{kT})$, which is significantly worse. For simplicity, we will focus on the case where $S=2$: namely, that we measure regret with respect to a single action from round 1 till some t_0, and then from t_0+1 till T. Moreover, we consider a simple case where $|G^1| = |G^2| = 1$ and $\Delta = 1/5$, so there is just a single action at a time which is significantly better than all the other actions in expectation.

Theorem 5. *Suppose that $T \geq Ck$ for some sufficiently large universal constant C. Then in the setting of Thm. 2, there exists a randomized reward assignment (with $|G^1| = |G^2| = 1$ and $\Delta = 1/5$), such that for any standard bandit algorithm, its expected regret (over the rewards assignment and the algorithm's randomness) is at least $0.007\sqrt{(k-1)T}$.*

The constant 0.007 is rather arbitrary and is not the tightest possible.

We note that a related $\Omega(\sqrt{T})$ lower bound has been obtained in (Garivier & Moulines, 2011). However, their result does not apply to the case $S=2$ and more importantly, does not quantify a dependence on k. It is interesting to note that unlike the standard lower bound proof for standard bandits (Auer et al., 2002), we obtain here an $\Omega(\sqrt{kT})$ regret even when $\Delta > 0$ is fixed and doesn't decay with T.

5. The Necessity of a Non-Uniform Querying Distribution

The theoretical results above demonstrated the efficacy of our approach, compared to standard bandit algorithms. However, the exact form of our querying distribution (querying action i with probability proportional to $\sqrt{p_j(t)}$) might still seem a bit mysterious. For example, maybe one can obtain similar results just by querying actions uniformly at random? Indeed, this is what has been done in some other online learning scenarios where queries were allowed (e.g., (Yu & Mannor, 2009; Agarwal et al., 2010)). However, we show below that in the adversarial setting, an adaptive and non-uniform querying distribution is indeed necessary to obtain regret bounds better than $\sqrt{kT}$. For simplicity, we return to our basic setting, where our goal is to compete with just the best single fixed action in hindsight.

Theorem 6. *Consider any online algorithm over $k > 2$ actions and horizon T, which queries the actions based on a fixed distribution. Then there exists a strategy for the adversary conforming to the setting described in Thm. 2, for which the algorithm's regret is at least $c\sqrt{kT}$ for some universal constant c.*

A proof sketch is presented in the appendix of the full version. The intuition of the proof is that if the querying distribution is fixed, and there are only a small number of "good" actions, then we spend too much time querying irrelevant actions, and this hurts our regret performance.

6. Experiments

We compare the decoupled approach with common multi-armed bandit algorithms in a simulated adversarial setting. Our user chooses between k communication channels, where sensing and transmission can be decoupled. In other words, she may choose a certain channel for transmission while sensing (i.e., querying) a different, seemingly less attractive, channel.

We simulate a heavily loaded UWB environment with a single, alternating, channel which is fit for transmission. The rewards of $k-1$ channels are drawn from alternating uniform and truncated Gaussian distributions with random parameters, yielding adversarial rewards in the range $[0,6]$. The remaining channel yields stochastic rewards drawn from a truncated Gaussian distribution bounded in the same range but with a mean drawn from $[3,6]$. The identity of the better channel and its distribution parameters are re-drawn at exponentially distributed switching times.

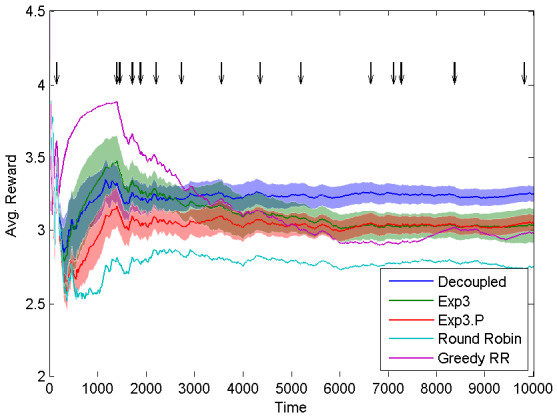

Figure 1. Average reward for different algorithms over time. Shaded areas around plots represent the standard deviation over repetitions.

Figure 1 displays the results of a scenario with $k=10$ channels, comparing the average reward acquired by the different algorithms over $T=10,000$ rounds. We implemented Algorithm 1, Exp3 (Auer et al., 2002), Exp3.P (Auer et al., 2002), a simple round robin policy (which just cycles through the arms in a fixed order) and a "greedy" decoupled form of round robin, which performs uniform queries and picks actions greedily based on the highest empirical average reward. The black arrows indicate rounds in which the identity of the stochastic arm and its distribution parameters were re-drawn. The results are averaged over 50 repetitions of a specific realization of rewards. Although we have tested our algorithm's performance on several realizations of switching times and rewards with very good results, we display a single realization of these for the sake of clarity.

Figure 2 displays the dynamics of channel selection for two of the $k=10$ channels. The thick plots represent the number of times a channel was chosen over time, and the thin plots represent the number of times it was queried. The dashed plots represent a channel which was drawn as the better channel during some periods, resulting in a relatively high average reward,

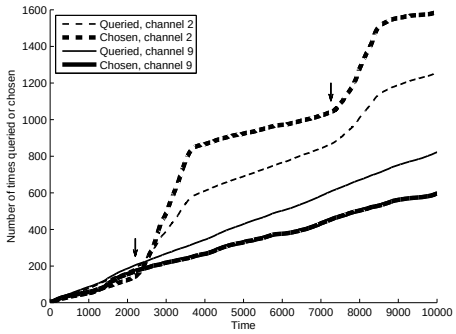

Figure 2. Number of times channels were chosen and queried over time, for two of $k=10$ arms. Arrows mark times in which channel 2 was drawn as the better channel.

while the solid plots represent a channel with a low average reward. The increased flexibility of the decoupled approach is evident from the graph, as well as the adaptive, nonlinear sampling policy.

Comments: We implement Algorithm 1 and not Algorithm 2 since the number of switches is unknown a-priori. Also, the rewards are in the range $[0,6]$ in order to keep all implemented algorithms on a similar scale, without violating the boundedness assumption.

7. Discussion

In this paper, we analyzed if and how one can benefit in settings where exploration and exploitation can be "decoupled:" namely, that one can query for rewards independently of the action actually picked. We developed some algorithms for this setting, and showed that these can indeed lead to improved results, compared to the standard bandit setting, under certain conditions. We also performed some experiments that corroborate our theoretical findings.

For simplicity, we focused on the case where only a single reward may be queried. If $c>1$ queries are allowed, it is not hard to show parallel guarantees to those in this paper, where the dependence on k is replaced by dependence on k/c. Algorithmically, one simply needs to repeatedly sample from the query distribution c times, instead of a single time. We conjecture that similar lower bounds can be obtained as well. Interestingly, it seems that being allowed to see the reward of the action actually picked, on top of the queried reward, does not result in significantly improved regret guarantees (other than better constants).

Several open questions remain. First, our results do not apply when the rewards are chosen by an adaptive adversary (namely, that the rewards are not fixed

in advance but may be chosen individually at each round, based on the algorithm's behavior in previous rounds). This is not just for technical reasons, but also because data and algorithm dependent quantities like $P(v, \delta, \mu)$ do not make much sense if the rewards are not considered as fixed quantities.

A second open question concerns the possible correlation between sensing and exploration. In some applications it is plausible that the choice of which arm to exploit affects the quality of the sample of the arm that is explored. For instance, in the UWB sensing example discussed in the introduction transmitting and receiving in the same channel is much less preferred than sensing in another channel because of interference in the same frequency band. It would be interesting to model such dependence and take it into account in the learning process.

Finally, it remains to extend other bandit-related algorithms, such as EXP4 (Auer et al., 2002), to our setting, and study the advantage of decoupling in other adversarial online learning problems.

Acknowledgements.

This research was partially supported by the CORNET consortium (http://www.cornet.org.il/).

References

Agarwal, A., Dekel, O., and Xiao, L. Optimal algorithms for online convex optimization with multi-point bandit feedback. In *COLT*, 2010.

Audibert, J.-Y. and Bubeck, S. Minimax policies for adversarial and stochastic bandits. In *COLT*, 2009.

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

Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. The nonstochastic multiarmed bandit problem. *SIAM J. Comput.*, 32(1):48–77, 2002.

Avner, O. and Mannor, S. Stochastic bandits with pathwise constraints. In *50th IEEE Conference on Decision and Control*, 2011.

Avner, O., Mannor, S., and Shamir, O. Decoupling exploration and exploitation in multi-armed bandits. arXiv:1205.2874v1 [cs.LG], 2012.

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

Cesa-Bianchi, N. and Lugosi, G. *Prediction, learning, and games*. Cambridge University Press, 2006.

Even-Dar, E., Mannor, S., and Mansour, Y. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of Machine Learning Research*, 7:1079–1105, 2006.

Freedman, D.A. On tail probabilities for martingales. *Annals of Probability*, 3:100–118, 1975.

Garivier, A. and Moulines, E. On upper-confidence bound policies for switching bandit problems. In *ALT*, 2011.

Herbster, M. and Warmuth, M. K. Tracking the best expert. *Machine Learning*, 32(2):151–178, 1998.

Lai, L., Jiang, H., and Poor, H. V. Medium access in cognitive radio networks: A competitive multi-armed bandit framework. In *Proc. Asilomar Conference on Signals, Systems, and Computers*, pp. 98–102, 2008.

Liu, K. and Zhao, Q. Distributed learning in multi-armed bandit with multiple players. *IEEE Transactions on Signal Processing*, 58(11):5667 –5681, nov. 2010.

Oppermann, I., Hamalainen, M., and Iinatti, J. *UWB Theory and Application*. Wiley, 2004.

Yu, J. Y. and Mannor, S. Piecewise-stationary bandit problems with side observations. In *ICML*, 2009.