Sleeping Experts and Bandits with Stochastic Action Availability and Adversarial Rewards

Varun Kanade[*]
Georgia Tech
varunk@cc.gatech.edu

Brendan McMahan
Google Inc.
mcmahan@google.com

Brent Bryan
Google Inc.
brent@google.com

Abstract

We consider the problem of selecting actions in order to maximize rewards chosen by an adversary, where the set of actions available on any given round is selected stochastically. We present the first polynomial-time no-regret algorithm for this setting. In the full-observation (experts) version of the problem, we present an exponential-weights algorithm that achieves regret $\mathcal{O}(\sqrt{T \log n})$, which is the best possible. For the bandit setting (where the algorithm only observes the reward of the action selected), we present a no-regret algorithm based on follow-the-perturbed-leader. This algorithm runs in polynomial time, unlike the EXP4 algorithm which can also be applied to this setting. Our algorithm has the interesting interpretation of solving a geometric experts problem where the actual embedding is never explicitly constructed. We argue that this adversarial-reward, stochastic-availability formulation is important in practice, as assuming stationary stochastic rewards is unrealistic in many domains.

1 Introduction

Online algorithms for selecting actions in order to maximize a reward or minimize a prediction loss have been extensively studied; Cesa-Bianchi and Lugosi (2006) provides a thorough introduction. However, in many practical domains not all actions are available at each timestep. Ads in an online auction may be made temporarily unavailable in order to limit the rate of depletion of an advertiser's budget, certain caches or servers in a computer system may be periodically unreachable, world financial markets are closed during certain hours of the day, known construction projects or congestion may limit the selection of driving routes, etc.

Given a world where actions are sometimes unavailable, it is natural to seek algorithms that perform almost as well as the best post-hoc ranking of actions, where the highest-ranked available action is always played. Kleinberg et al. (2008) introduced this notion of regret and gave efficient algorithms for several variations of the problem with stochastic rewards. They also point out that the EXP4 algorithm (Auer et al., 2003) achieves no-regret in the adversarial rewards setting, but unfortunately runs in time exponential in the total number of actions. Since the stochastic rewards assumption does not hold in many interesting domains, this leaves open the natural question of finding efficient algorithms for limited action availability problems where rewards are chosen by an adversary. In this paper we present such a model of adversarial rewards, under the assumption that action availability is stochastic and independent of the rewards. We also provide experimental evidence that even if EXP4 could be implemented efficiently, its performance on problems that actually have stochastic action availability will be worse than the algorithms we propose.

Notation and Formal Model: Assume a fixed set $\mathcal{A}$ of possible actions, indexed by integers $1, \ldots, n$. An algorithm selects one action per round over a sequence of rounds (indexed by t), each of which proceeds as follows:

Step 1: An adversary or randomness selects the set $\mathcal{A}^t \subseteq \mathcal{A}$ of actions available on round t, and a reward vector $r \in \mathbb{R}^n$. We describe four models for this selection process below. For simplicity, we assume this vector assigns reward $r[a]$ to all actions in $\mathcal{A}$, even those not available on this round.

[*]Part of work completed while visiting Google.

Appearing in Proceedings of the 12^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2009, Clearwater Beach, Florida, USA. Volume 5 of JMLR: W&CP 5.

Step 2: The algorithm observes the set $\mathcal{A}^t$ (but not the rewards r), selects an action $\grave{a} \in \mathcal{A}^t$ to play, and receives reward $r[\grave{a}]$. In the *bandit setting*, the algorithm only observes this single reward; in the *full information (or experts) setting* the full reward vector r is observed.

We compare our performance to the best *action list* in hindsight; an action list is an ordering (permutation) on the set of actions. Given the best action list, the optimal strategy is to play the highest-ranked action that is available. We will be interested in minimizing regret with respect to the best action list. Let σ be an action list on $\mathcal{A}$. We abuse notation slightly and treat σ as a function on subsets of $\mathcal{A}$ such that $\sigma(\mathcal{A}^t)$ is the action in $\mathcal{A}^t$ ranked highest by σ. For example, suppose $\sigma = (2, 3, 1)$, then $\sigma(\{1, 3\}) = 3$ and $\sigma(\{1, 2, 3\}) = 2$, etc.

We consider 4 different models for step 1:
Stochastic availability, stochastic rewards: The set $\mathcal{A}^t$ is selected by sampling from a fixed joint distribution, $\Pr_{\text{avail}}$, on subsets of $\mathcal{A}$, which is independent of time and rewards. Rewards $r[a]$ are chosen from fixed distributions $\Pr_a$ that are independent of time and action availability.
Adversarial availability, stochastic rewards: First an adversary chooses the set $\mathcal{A}^t$, and then rewards $r[a]$ are sampled from fixed distributions $\Pr_a$ (which are independent of $\mathcal{A}^t$).
Stochastic availability, adversarial rewards: First, an adversary chooses the reward vector r, and then the set $\mathcal{A}^t$ is drawn from $\Pr_{\text{avail}}$. $\Pr_{\text{avail}}$ can be viewed as a joint distribution of n random variables X_a for each $a \in \mathcal{A}$. Each X_a takes on value 1 when $a \in \mathcal{A}^t$ and 0 otherwise. The algorithm we present works against an adaptive adversary.
Adversarial availability, adversarial rewards: A single adversary selects both $\mathcal{A}^t$ and r at the same time.

Notions of regret differ slightly between these settings. In particular, in the adversarial/adversarial case, one must consider regret against the best action list with respect to the sets $\mathcal{A}^t$ actually selected by the adversary. However, in the stochastic availability case, a more natural notion of regret is to compare ourselves to the *expected* performance of the best action list. Letting $S_{\mathcal{A}}$ be the set of all permutations on $\mathcal{A}$, we define the regret of an algorithm in the stochastic availability setting as:

$$\Re(\text{Alg}) = \max_{\sigma \in S_{\mathcal{A}}} \mathbb{E}_{\mathcal{A}^t} \left[\sum_{t=1}^{T} r^t[\sigma(\mathcal{A}^t)] \right] - \sum_{t=1}^{T} r^t[\grave{a}^t],$$

where $\grave{a}^t$ are the actions taken by the algorithm. The order of the max and expectation is important: we are competing with the best action list chosen with knowledge of all T reward vectors and the distribution $\Pr_{\text{avail}}$, but without knowing which $\mathcal{A}^t$ will actually be available on each round.

Our algorithms are randomized, and so $\Re(\text{Alg})$ is a random variable; one can consider both high-probability bounds on $\Re(\text{Alg})$ and bounds on expectation; in this paper we present bounds on $\mathbb{E}[\Re(\text{Alg})]$, where the expectation is over the random choices of the algorithm.

Related Work: For the first two models, many solutions have been proposed. For the setting when all actions are available and rewards stochastic, there is large body of work starting with (Lai and Robbins, 1985). Even-Dar et al. (2002) gives an algorithm that is optimal in terms of number of exploration steps. Their algorithm works by exploring actions for the first few time steps, and then exploiting the best action for the remaining time steps. This approach is not applicable in the sleeping experts/bandit setting, since some actions may be sleeping throughout the exploration phase. Kleinberg et al. (2008) proposes algorithm AUER and prove that this algorithm is information theoretically (almost) optimal. These algorithms work when the availability is adversarial as well. While the analysis presented in these papers is slightly different, it is straightforward to show these algorithms satisfy the bounds shown in Table 1.

In this paper, we are interested in rewards that are chosen by an adversary. There are two types of adversaries commonly considered in the bandit setting. An *oblivious adversary* knows the strategy of the algorithm that selects the actions, but not the sequence of random choices (if any) made by the algorithm. On the other hand, an *adaptive adversary* gets to observe the random choices made by the algorithm on the previous rounds in addition to knowing the strategy of the algorithm. The algorithm we present works against an adaptive adversary.

Table 1 shows the best known regret bounds for the four settings discussed above. When rewards are stochastic, an adversary cannot gain by controlling availability, and the algorithms mentioned above work in either case. As observed by Kleinberg et al. (2008), in the case when an adversary decides rewards and availability EXP4 gives low regret, but it is not efficient because it involves keeping track of all $n!$ action lists.

The problem of online decision making where some experts are unavailable has been considered previously; see for example (Freund et al., 1997; Blum and Mansour, 2007). The notions of regret used are different from the one considered in this work, and not directly

Reward	**Action Availability**	Bound	Algorithm	Reference
Stochastic	Stochastic	$\mathcal{O}(\sqrt{nT\log(T)})$	AUER	(Kleinberg et al., 2008)
Stochastic	Adversarial	$\mathcal{O}(\sqrt{nT\log(T)})$	AUER	(Kleinberg et al., 2008)
Adversarial	Stochastic	$\mathcal{O}(n^{\frac{4}{5}}T^{\frac{4}{5}}\log(T))$	Current work	
Adversarial	Adversarial	$\mathcal{O}(n\sqrt{T\log(n)})$	EXP4	(Auer et al., 2003)

Table 1: Limited action availability models in the bandit setting. Note that EXP4 does not run in polynomial time per round.

Parameter $\epsilon > 0$
for $t = 1, \ldots, T$
  Draw vector $Z^t \in [0, 1/\epsilon]^n$
    uniformly at random;
  Let $R^t = r^1 + \ldots + r^{t-1}$;
  Let $\sigma^t = \text{sort}(R^t + Z^t)$;
  Choose action $\grave{a}^t = \sigma^t(\mathcal{A}^t)$
  Get reward $r^t[\grave{a}^t]$ and observe r^t;

Figure 1: Algorithm: Sleeping Follow the Perturbed Leader (SFPL)

comparable.

A Motivating Example We consider the problem of selecting ads to display alongside a search result as a motivating domain. The revenue model of most search companies today is *pay per click*. Thus an important aspect of ad selection is estimating correct *click through rates* for a given advertisement. We consider a simplified model so that we can focus on partial availability. In particular, only a single ad is shown for each search, and then we observe whether that ad was clicked (in which case we get a positive reward) or not (reward 0). Thus, we have formulated a multiarmed bandit problem. Our choice of arms is a large pool of advertisements, only a subset of which are available on each round.

We believe the stochastic/adversarial model is particularly appropriate for this and other real-world domains. Ads can be unavailable for many reasons that are independent of the reward we would receive for showing it. For example, ad distributors could randomly consider an ad unavailable to avoid depleting an advertiser's budget too quickly, or because the ad is not relevant at the particular time or geographic location of the query. It is worth clarifying that in practice we do not expect the rewards (which are influenced primarily by whether or not a user clicks) to be adversarial. However, using an algorithm that is robust to such adversaries means that we can avoid making the strong (and doubtless incorrect) assumption that the reward for each action comes from a fixed distribution that is independent of time.

2 Algorithms and Analysis

We begin by considering the **full information setting**, where the full r^t vector is revealed to the algorithm (even the rewards of the actions that were not available). We will then use the first algorithm introduced here as a subroutine in an algorithm for the bandit setting.

We present the Sleeping Follow the Perturbed Leader (SFPL_ϵ) algorithm in Figure 1. The algorithm takes a parameter ϵ which determines the magnitude of perturbations. We use the definitions of Z^t, R^t and σ^t from Figure 1. Let $\text{sort}(v)$ return a permutation of indices of vector v, so that the permutation indexes v in descending order; for example, if $v = (0.1, 0.7, 0.4)$, then $\text{sort}(v) = (2, 3, 1)$. SFPL_ϵ will play an action list that results from sorting the actions based on perturbed cumulative rewards, $\sigma^t = \text{sort}(R^t + Z^t)$.

We relate the performance of SFPL_ϵ to the performance of a geometric experts algorithm on a hypothetical geometric optimization problem. A permutation σ represents the following strategy: On each round play the first available action (according to σ). Since the availability is decided by a joint distribution independently on each round, on a particular round t, the expected reward for a fixed σ is

$$\sum_{\mathcal{A}^t \in \mathcal{P}(\mathcal{A})} \Pr_{\text{avail}}(\mathcal{A}^t)\ r^t[\sigma(\mathcal{A}^t)] = \sum_{a \in \mathcal{A}} \Pr_{\text{avail}}[\sigma(\mathcal{A}^t) = a] r^t[a]$$

where $\mathcal{P}(\mathcal{A})$ is the powerset of $\mathcal{A}$. The probabilities are with respect to the randomness in the choice of $\mathcal{A}^t$. If k is the index of a in σ, then $\Pr_{\text{avail}}[\sigma(\mathcal{A}^t) = a]$ is equal to $\Pr_{\text{avail}}[X_{\sigma_1} = 0, \ldots, X_{\sigma_{k-1}} = 0, X_{\sigma_k} = 1]$ that is, the marginal probability that all actions ranked higher than a are unavailable and a is available. The quantity $\sum_{a=1}^{n} \Pr[\sigma(\mathcal{A}^t) = a] r^t[a]$ looks very much like a dot product, which suggests a geometric optimization problem; we now define such a problem. Let $\ell : S_\mathcal{A} \to \mathbb{R}^n$ be the function such that $\ell(\sigma)[a] = \Pr_{\mathcal{A}^t}[\sigma(\mathcal{A}^t) = a]$. In this manner the set of action lists defines a subset L in $\mathbb{R}^n$. For example, let $n = 3$, and suppose that each action is available

independently at random with probability 1/2. For the action list $\sigma = (2, 3, 1)$ the vector $\ell(\sigma) \in \mathbb{R}^3$ is $(\frac{1}{8}, \frac{1}{2}, \frac{1}{4})$. If we choose action list σ to play on round t our expected reward is exactly $\ell(\sigma) \cdot r^t$. Throughout this paper, the *corresponding geometric problem* refers to the geometric online optimization problem with the feasible set $L = \{\ell(\sigma) \mid \sigma \in S_{\mathcal{A}}\}$. We use *follow the perturbed leader* (Kalai and Vempala, 2005; Hannan, 1957) with parameter ϵ (FPL_ϵ) to solve this geometric problem. At time step t, FPL_ϵ, picks a random vector $Z^t \in [0, 1/\epsilon]^n$, and finds $x \in L$ such that $x \cdot (R^t + Z^t)$ is maximized. We couple the randomness of SFPL_ϵ and FPL_ϵ so that they draw the same random vector Z^t. If SFPL_ϵ picks σ^t at time t, $\ell(\sigma^t) \cdot (R^t + Z^t) = \max_{x \in L} x \cdot (R^t + Z^t)$. This relatively simple observation reveals essential structure induced by the stochastic availability model, and so it is worth stating the result formally (proof appears in a full version):

Lemma 1. *Fix an arbitrary distribution* $\Pr_{\text{avail}}$ *on the possible* $\mathcal{A}^t$ *and a vector* $v \in \mathbb{R}^n$*, and consider the action list* $\sigma = \text{sort}(v)$*. Then* $\ell(\sigma) \cdot v = \max_{x \in L} x \cdot v$*, where* ℓ *is defined with respect to* $\Pr_{\text{avail}}$*.*

An important corollary is that the post-hoc optimal action list σ^* will always be $\text{sort}(R^{T+1})$, the action list obtained by sorting actions according to their total cumulative reward. Importantly, this action list will be optimal for *any* availability distribution $\Pr_{\text{avail}}$.

Unlike most algorithms for geometric experts problems, FPL_ϵ only requires an oracle to return a point in the feasible region that maximizes the dot product with $R^t + Z^t$. This allows us to simulate it without knowing the feasible set L. We state the result about the performance of FPL_ϵ (Theorem 1.1 in (Kalai and Vempala, 2005)) as Lemma 2:

Lemma 2. *Let* ν *be an adversary that selects reward vectors* $r^t \in \mathbb{R}^n$ *as a deterministic function of the algorithm's previous actions* $s^1, \ldots, s^{t-1}$*. If* S *is the feasible region in* $\mathbb{R}^n$ *and* A*,* D*, and* $\tilde{R}$ *are such that* $A \geq \|r^t\|_1$*,* $D \geq \|s - s'\|_1$*,* $\tilde{R} \geq |s \cdot r^t|$ *for any* $s, s' \in S$ *and all* r^t*, then if* $s^1, \ldots, s^T$ *are points picked by* FPL_ϵ *for* $0 < \epsilon \leq 1$*:*

$$\mathbb{E}\left[\sum_{t=1}^{T} r^t \cdot s^t\right] \geq \mathbb{E}\left[\max_{s \in S} \sum_{t=1}^{T} r^t \cdot s\right] - \epsilon A \tilde{R} T - \frac{D}{\epsilon}$$

The following lemma relates the performance of SFPL_ϵ and FPL_ϵ (on the geometric problem). Let $\mathbf{Z} = (Z^1, \ldots, Z^T)$ denote the random choices made by SFPL_ϵ.

Lemma 3. *Let* ν *be an adversary that selects reward vectors* $r^t \in \mathbb{R}^n$ *as a deterministic function of the algorithm's and environment's previous random choices. Suppose that* $\|r^t\|_1 \leq A$ *and* $|\ell(\sigma) \cdot r^t| \leq \tilde{R}$ *for all* $\sigma \in S_{\mathcal{A}}$ *and for all* t*, the action lists* $\sigma^1, \ldots, \sigma^T$ *played by* SFPL_ϵ *satisfy*

$$\mathbb{E}\left[\sum_{t=1}^{T} r^t[\sigma^t(\mathcal{A}^t)]\right] \geq \max_{\sigma \in S_{\mathcal{A}}} \mathbb{E}\left[\sum_{t=1}^{T} r^t[\sigma(\mathcal{A}^t)]\right] - \epsilon A \tilde{R} T - \frac{2}{\epsilon}$$

Proof. Let $\mathcal{H}^t$ denote the history of random choices made by the algorithm and the environment before (but *not* including) time step t. Let $r^1, \ldots, r^T$ denote the reward sequence chosen by the adversary. Recall $L = \{\ell(\sigma) \mid \sigma \in S_{\mathcal{A}}\}$ is the feasible set of the corresponding geometric optimization problem. Suppose the vectors $r^1, \ldots, r^T$ are passed as reward vectors to FPL_ϵ attempting to solve the geometric problem. We couple the randomness used by SFPL_ϵ and FPL_ϵ, i.e. they draw the same random vector Z^t at time step t. Since the reward vector r^t does not depend on the random subset of available actions at round t, it is clear that:

$$\mathbb{E}_{\mathcal{A}^t}[r^t[\sigma^t(\mathcal{A}^t)] \mid \mathcal{H}^t] = \ell(\sigma^t) \cdot r^t \tag{1}$$

where r^t is a constant given $\mathcal{H}^t$. By Lemma 1, $\ell(\sigma^t)$ maximizes $x \cdot (R^t + Z^t)$ for $x \in L$, and hence FPL_ϵ can pick $\ell(\sigma^t)$ whenever SFPL_ϵ picks σ^t. Note that

$$\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{T} r^t[\sigma^t(\mathcal{A}^t)]\right] &= \sum_{t=1}^{T} \mathbb{E}\left[r^t[\sigma^t(\mathcal{A}^t)]\right] \\
&= \sum_{t=1}^{T} \mathbb{E}_{\mathcal{H}^t}\left[\mathbb{E}_{\mathcal{A}^t}[r^t[\sigma^t(\mathcal{A}^t)] \mid \mathcal{H}^t]\right] \\
&= \sum_{t=1}^{T} \mathbb{E}_{\mathcal{H}^t}\left[\ell(\sigma^t) \cdot r^t\right] = \mathbb{E}\left[\sum_{t=1}^{T} \ell(\sigma^t) \cdot r^t\right],
\end{aligned} \tag{2}$$

using (1). By Jensen's inequality,

$$\begin{aligned}
\mathbb{E}\left[\max_{\sigma \in S_{\mathcal{A}}} \sum_{t=1}^{T} \ell(\sigma) \cdot r^t\right] &\geq \max_{\sigma \in S_{\mathcal{A}}} \mathbb{E}\left[\sum_{t=1}^{T} \ell(\sigma) \cdot r^t\right] \\
&= \max_{\sigma \in S_{\mathcal{A}}} \mathbb{E}\left[\sum_{t=1}^{T} r^t[\sigma(\mathcal{A}^t)]\right],
\end{aligned} \tag{3}$$

where the equality follows from an argument analogous to Equation (2). Finally, for any two vectors $\ell(\sigma), \ell(\sigma') \in L$, it holds that $\|\ell(\sigma) - \ell(\sigma')\|_1 \leq 2$. Using the hypothesis that $\|r^t\|_1 \leq A$ and $|\ell(\sigma) \cdot r^t| \leq \tilde{R}$ for all $\ell(\sigma) \in L$ and for all r^t, applying Lemma 2 to (2) and (3) proves the lemma. □

```
inputs: parameter η
(∀a ∈ A^t)  R^1[a] = 0
for t = 1, ..., T
    w^t[a] = exp(η R^t[a])
    Observe A^t drawn from Pr_avail
    W^t = Σ_{a∈A^t} w^t[a]
    (∀a ∈ A^t)  Let q^t[a] = w^t[a]/W^t
    Sample à from q^t
    Play à, get reward r^t[à]
    Observe full vector r^t
    (∀a ∈ A^t)  R^{t+1}[a] = R^t[a] + r^t[a]
```

Figure 2: Algorithm EWSA.

An Optimal Exponential Weights Algorithm We introduce the EWSA Algorithm, (for *Exponential Weights, Stochastic Availability*); pseudocode is given in Figure 2. EWSA achieves the best-possible regret bounds for the full-information stochastic availability, (oblivious) adversarial reward problem:

Theorem 1. *If* $\eta = \sqrt{(8/T)\log n}$*, Algorithm* EWSA *has*

$$\mathbb{E}[\Re(\mathsf{EWSA})] \leq \sqrt{T \log n}$$

when playing against an oblivious adversary and making full observations of the reward vector r^t *each round.*

Proof. The proof connects the behavior of EWSA to the behavior of an imagined instance of an exponential-weights algorithm EW (say, hedge or weighted majority) on particular fixed-availability problems. In particular, consider a fixed action set $\bar{A} \subseteq \mathcal{A}$, and let $a^* = \operatorname{argmax}_{a \in \bar{A}} R^{T+1}[a]$, the best single action in $\bar{A}$ chosen post-hoc. Standard bounds for EW give

$$\sum_{t=1}^{T} \left(r^t[a^*] - \sum_{a \in \bar{A}} q^t[a] r^t[a]\right) \leq \sqrt{T \log n} \qquad (4)$$

(e.g., Theorem 2.2 of (Cesa-Bianchi and Lugosi, 2006)), where q^t is the distribution played by the exponential-weights algorithm.[1] If $\bar{A}$ is available on round t, EWSA chooses its distribution q^t based only on the cumulative rewards of the actions in $\bar{A}$, and in fact chooses them by *exactly* the same formula as EW (so writing q^t for both is in fact not an abuse of notation). Further, if σ^* is the post-hoc best action list, $\sigma^* = \operatorname{sort}(R^{T+1})$ (as a corollary to Lemma 1), and so

$$a^* = \sigma^*(\bar{A}) \qquad (5)$$

[1] Note that this bound holds for any $\bar{A}$ as long as we fix the reward multiplying parameter η of EW based on n, and not $|\bar{A}|$.

Thus, we conclude that *if* it so happened that $\bar{A}$ was selected as available on every round of the game, the above bound would hold for EWSA. Now, it suffices to show that EWSA's expected regret in the real game can be written as a weighted sum of its regret if each set $\bar{A}$ happened to be fixed for every round. We have,

$$\begin{aligned}
&\mathbb{E}[\Re(\mathsf{EWSA})] \\
&\leq \sum_{t=1}^{T} \sum_{\bar{A} \subseteq \mathcal{A}} \Pr(\bar{A}) \Big(r^t[\sigma^*(\bar{A})] - \sum_{a \in \bar{A}} q^t[a] r^t[a]\Big) \\
&= \sum_{\bar{A} \subseteq \mathcal{A}} \Pr(\bar{A}) \sum_{t=1}^{T} \Big(r^t[\sigma^*(\bar{A})] - \sum_{a \in \bar{A}} q^t[a] r^t[a]\Big)
\end{aligned}$$

and substituting (5) into (4),

$$\leq \sum_{\bar{A} \subseteq \mathcal{A}} \Pr(\bar{A}) \sqrt{T \log n} = \sqrt{T \log n}.$$

□

The standard (full-availability) experts problem is a special case of our stochastic availability setting, and so the lower bound on regret of $\sqrt{T \log n}$ (e.g., Cesa-Bianchi and Lugosi (2006)) for that setting also applies here, showing that the bound of Theorem 1 is essentially the best possible.

Bandit Setting: We now turn to the bandit (partial reward observability) setting. We show that as long as the number of rounds is large enough ($T = \Omega(n^4)$), the bandit version of our algorithm has low regret.

Figure 3 presents a bandit version of SFPL. For convenience, we assume that the number of rounds is $T = T_0 + T_1$; we label the initial rounds $-T_0$ through -1, and the remaining rounds 1 through T_1. Our algorithm uses the first T_0 rounds to construct estimates $\hat{p}[a]$ of the marginal probabilities of availability $p[a] = \Pr_{\text{avail}}(X_a = 1)$ for each action a. At the end of this phase, the algorithm maintains a set of actions $\mathcal{A}_\beta = \{a \in \mathcal{A} \mid \hat{p}[a] \geq \beta\}$, where β is a parameter. While our algorithm will only play actions from this set, it will still get low regret with respect to the best action list over all actions.

During rounds $1, \ldots, T_1$, the algorithm on each round decides whether to explore (with probability γ) or exploit, by setting the variable χ^t. While exploiting, the master algorithm simply follows the advice of the black-box stochastic availability experts algorithm (e.g., SFPL). The reward vector passed down to the black-box algorithm in this case is the zero vector $\mathbf{0}$. When exploring, the master algorithm picks an action $\tilde{a} \in \mathcal{A}_\beta$ uniformly at random. If $\tilde{a} \in \mathcal{A}^t$, it gets reward $b = r^t[\tilde{a}]$, otherwise $b = 0$. The reward vector

Parameters:
$\alpha \geq 1$, $\beta < 1$, $\gamma < 1$
Set $\epsilon = \frac{\beta}{n}\sqrt{\frac{\gamma}{2T_1}}$
for $t = -T_0 \ldots, -1$
  observe available actions $\mathcal{A}^t$
  for $a = 1, \ldots, n$

$$c[a] = c[a] + \begin{cases} 1 & \text{if } a \in \mathcal{A}^t \\ 0 & \text{otherwise} \end{cases}$$

for $a = 1, \ldots, n$
  $\hat{p}[a] = c[a]/T_0$
$\mathcal{A}_\beta = \{a \mid \hat{p}[a] \geq \beta\}$
for $t = 1, \ldots, T_1$
  observe available actions $\mathcal{A}^t$

$$\chi^t = \begin{cases} 1 & \text{with probability } \gamma \\ 0 & \text{otherwise} \end{cases}$$

  $\hat{\sigma}^t = \mathsf{SFPL}_\epsilon(\hat{r}^1, \ldots, \hat{r}^{t-1})$
  if $\chi^t = 1$ // exploration round
    sample $\tilde{a}$ uniformly from $\mathcal{A}_\beta$

$$\grave{a}^t = \begin{cases} \tilde{a} & \text{if } \tilde{a} \in \mathcal{A}^t \\ \hat{\sigma}^t(\mathcal{A}^t) & \text{otherwise} \end{cases}$$

    play $\grave{a}^t$, observe $r^t[\grave{a}^t]$

$$b = \begin{cases} r^t[\grave{a}^t] & \text{if } \tilde{a} \in \mathcal{A}^t \\ 0 & \text{otherwise} \end{cases}$$

    $\hat{r}^t = \alpha \mathbf{1}$
    $\hat{r}^t[\grave{a}^t] \mathrel{+}= \frac{nb}{\gamma \hat{p}[\grave{a}^t]}$
  else // exploit
    play $\grave{a}^t = \hat{\sigma}^t(\mathcal{A}^t)$, observe $r^t[\grave{a}^t]$
    $\hat{r}^t = \mathbf{0}$

Figure 3: Algorithm BSFPL (Bandit SFPL)

passed down is $\alpha\mathbf{1} + \frac{nb}{\gamma\hat{p}[\tilde{a}]}\mathbf{e}_{\tilde{a}}$, where $\mathbf{1}$ is the vector with all ones and $\mathbf{e}_{\tilde{a}}$ is the unit vector with 1 in position $\tilde{a}$. Here $\alpha \geq 1$ is a parameter that causes deliberate overestimation, reasons for which shall be discussed later. On any round, the reward vector that is passed to the black-box algorithm is an almost unbiased estimate of the true reward vector. This algorithm is similar to the McMahan-Blum algorithm (McMahan and Blum, 2004) for the geometric bandit problem. Note that $\hat{p}[a]\mathbf{e}_a$ form an (almost) barycentric spanner for the geometric problem defined earlier. Our analysis is similar in spirit to that of (Dani and Hayes, 2006).

In the rest of the section, we let $\mathbf{r} = (r^1, \ldots, r^{T_1})$ be the reward vectors for rounds $1, \ldots, T_1$, $r^t \in [0,1]^n$, and $\hat{\mathbf{r}} = (\hat{r}^1, \ldots, \hat{r}^{T_1})$ be the vectors that the algorithm passes down to the black-box.

We first sketch the main ideas of the three lemmas required for the proof. While the master algorithm is trying to minimize regret with respect to the best action list, the black-box algorithm is trying to solve the geometric problem with feasible set $L = \{\ell(\sigma) \mid \sigma \in S_{\mathcal{A}}\}$. The first of the three lemmas relates the performance of the master algorithm and the black-box algorithm, stating that the black-box algorithm can't have reward much higher than the master algorithm. The second lemma uses the properties of FPL to show that the black-box algorithm must have low regret. The third lemma shows that the reward of the best strategy for the geometric problem can't be much lower than the reward of the best action list for the original problem. Combining these implies BSFPL has low regret. Details of omitted proofs can be found in the full version.

Although the black-box algorithm is actually solving a sleeping experts problem, it can be used to solve the corresponding geometric problem. We will assume that the black-box algorithm has oracle access to the function ℓ and that when it plays action list $\hat{\sigma}$, it actually plays $\ell(\hat{\sigma})$ to get reward $\ell(\hat{\sigma}) \cdot \hat{r}$. Note that we can do this only because of the unique property of FPL, which requires only an oracle that gives the best point in the feasible set (and does not need to know the set itself!). We assume below that we have access to good estimates for the probabilities $p[a]$, by setting T_0 appropriately later.

Lemma 4. *Assume that for each action a, it holds that $1 - \xi \leq \frac{\hat{p}[a]}{p[a]} \leq \frac{1}{1-\xi}$ and that $\alpha \geq 1$, then*

$$\mathbb{E}\left[\sum_{t=1}^{T_1} r^t[\grave{a}^t]\right] \geq (1-\xi)\mathbb{E}\left[\sum_{t=1}^{T_1} \ell(\hat{\sigma}^t) \cdot \hat{r}^t\right] - 2\alpha\gamma T_1$$

The next lemma uses the bounds of FPL from Lemma 2 (see Kalai and Vempala (2005)). The proof of this is similar to those in Dani and Hayes (2006).

Lemma 5. *Let $\hat{\mathbf{r}} = (\hat{r}^1, \ldots, \hat{r}^{T_1})$ be a sequence of reward vectors the algorithm passes to black-box SFPL_ϵ, and assume $\hat{p}[a] \geq \beta$ and $\alpha \leq 1/(\beta\gamma)$. Then*

$$\mathbb{E}\left[\sum_{t=1}^{T_1} \ell(\hat{\sigma}^t) \cdot \hat{r}^t\right] \geq \mathbb{E}\left[\max_{\hat{\sigma}} \sum_{t=1}^{T_1} \ell(\hat{\sigma}) \cdot \hat{r}^t\right] - 4\sqrt{2}\frac{n}{\beta}\sqrt{\frac{T_1}{\gamma}}$$

The expectation is over all the random choices of the algorithm and the environment (on which $\hat{r}^t$ may depend).

Lemma 6 shows the total reward of the best strategy on the geometric problem (with $\hat{\mathbf{r}}$ as the reward vectors) is not much lower than the total expected reward of the best action list with the actual reward vectors $r^1, \ldots, r^T$. In order to ensure this, we were required to overestimate the rewards slightly by adding $\alpha\mathbf{1}$.

Lemma 6. *Assuming that $\alpha \geq 1$, $\xi \leq \alpha\gamma \leq 1$, and that all actions a satisfy $\hat{p}[a] \geq \beta$*

$$\mathbb{E}\left[\max_{\hat{\sigma}} \sum_{t=1}^{T_1} \ell(\hat{\sigma}) \cdot \hat{r}^t\right] \geq \mathbb{E}\left[\max_{\sigma} \sum_{t=1}^{T_1} \ell(\sigma) \cdot r^t\right] - \sqrt{2T_1\left(5\alpha^2 + \frac{2n}{\beta} + 16\right)\log\left(\frac{1}{\beta}\right)} - \beta n T_1$$

We can put the lemmas together to get our main result, that BSFPL has low regret.

Theorem 2. *Assume $T = T_0 + T_1 = \Omega(n^4)$. The bandit algorithm* BSFPL *with parameters $\alpha = 1$, $\beta = 2^{\frac{6}{5}} n^{-\frac{1}{5}} T^{-\frac{1}{5}}$, $\gamma = 2^{\frac{1}{5}} n^{\frac{4}{5}} T^{-\frac{1}{5}}$, $\xi = \gamma$, $T_0 = n^{-\frac{6}{5}} T^{\frac{4}{5}} \log(T)$, satisfies the following:*

$$\mathbb{E}\left[\sum_{t=-T_0}^{T_1} r^t[\grave{a}^t]\right] \geq \max_{\sigma} \mathbb{E}\left[\sum_{t=-T_0}^{T_1} r^t[\sigma(r^t)]\right] - \mathcal{O}(n^{\frac{4}{5}} T^{\frac{4}{5}} \log(T)).$$

Proof. With the given settings of the parameters the assumptions of Lemmas 4, 5 and 6 are satisfied; combining their inequalities, we have

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=1}^{T_1} r^t[\grave{a}^t]\right] - \mathbb{E}\left[\max_{\sigma} \sum_{t=1}^{T_1} \ell(\sigma) \cdot r^t\right] \\
&\quad \geq -2\alpha\gamma T_1 - 4\sqrt{2}\frac{n}{\beta}\sqrt{\frac{T_1}{\gamma}} - \xi T_1 \\
&\qquad - \sqrt{2T_1\left(5\alpha^2 + \frac{2n}{\beta} + 16\right)\log\left(\frac{1}{\beta}\right)} \\
&\quad \geq -4 \cdot 2^{\frac{6}{5}} n^{\frac{4}{5}} T^{\frac{4}{5}} - 7 n^{\frac{2}{5}} T^{\frac{3}{5}} \sqrt{\log(nT)} \\
&\quad \geq -\mathcal{O}(n^{\frac{4}{5}} T^{\frac{4}{5}}).
\end{aligned}$$

We now show how to bound estimates of probabilities $\hat{p}[a]$ for all actions. Referring to the steps $-T_0, \ldots, -1$ in Algorithm BSFPL (Figure 3), at time $t = 0$, $c[a]$ is the number of times action a was available during the rounds $-T_0, \ldots, -1$. Also, $p[a]$ is the true marginal probability of availability of action a, hence if $\hat{p}[a] = c[a]/T_0$, using Hoeffding bounds we get

$$\Pr[|\hat{p}[a] - p[a]| \geq \beta\xi] \leq 2\exp(-2\beta^2\xi^2 T_0).$$

Since $T_0 = n^{-\frac{6}{5}} T^{\frac{4}{5}} \log(T)$ and $(\beta\xi)^2 = 4n^{\frac{6}{5}} T^{-\frac{4}{5}}$, with probability at least $1 - \frac{1}{T}$ it holds for all actions $a \in \mathcal{A}$, that $|\hat{p}[a] - p[a]| \leq \beta\xi$. In the case this does not hold (with probability $\frac{1}{T}$), the algorithm can have regret at most T, contributing regret $O(1)$ in expectation. When the estimates of probabilities are good, since $\hat{p}[a] \geq p[a] - \beta\xi$, $\hat{p}[a] < \beta$ implies that $p[a] < \beta + \beta\xi$. So far we've not addressed the issue of actions that have very small available probabilities (less that β). By ignoring all actions a for which $\hat{p}[a] < \beta$, the algorithm would have regret $\mathcal{O}(\beta n T) = \mathcal{O}(n^{\frac{4}{5}} T^{\frac{5}{5}})$. Lastly, it can be easily checked that all actions satisfy:

$$1 - \xi \leq \frac{p[a]}{\hat{p}[a]} \leq \frac{1}{1 - \xi}$$

The T_0 steps for computing probabilities would result in the algorithm forgoing at most $\mathcal{O}(T_0) = \mathcal{O}(n^{\frac{4}{5}} T^{\frac{4}{5}} \log(T))$ reward, and can thus cause at most that much additive regret. Finally, using an argument analogous to the one in Lemma 3:

$$\mathbb{E}\left[\max_{\sigma \in S_{\mathcal{A}}} \sum_{t=-T_0}^{T_1} \ell(\sigma) \cdot r^t\right] \geq \max_{\sigma \in S_{\mathcal{A}}} \mathbb{E}\left[\sum_{t=-T_0}^{T_1} r^t[\sigma(\mathcal{A}^t)]\right].$$

□

3 Experiments

As mentioned earlier, the EXP4 algorithm achieves better regret bounds than BSFPL, but no polynomial-time implementation is known, and so running it for more than a handful of actions is impractical. In this section we show experimentally that on problems that have stochastic availability, BSFPL can actual outperforms EXP4, despite the latter's superior regret bound.

A simple example provides some intuition for this. Consider a problem with three actions $\{a, b, c\}$ with adversarial rewards and availability. The adversary is then free to assign reward vector $r^t = (0.9, 0.6, 0.0)$ whenever $\mathcal{A}^t = \{a, b, c\}$, but set $r^t = (0.0, 0.0, 0.7)$ whenever $\mathcal{A}^t = \{b, c\}$. Hence, the optimal action list is (a, c, b). In this example, however, observations of the rewards on b and c made when a happens to be available are *completely misleading* as to the correct ranking of b and c in the optimal action list. The stochastic availability assumption directly rules out such pathological cases and hence allows algorithms like BSFPL to estimate the performance of each action independently of the context in which the algorithm was available.

We use a very simple experimental setup to demonstrate this in practice. We consider a problem with 5 actions, each of which is available on a given round with an (independent) probability of 0.5. Rewards at $t = 0$ are chosen uniformly from $[0, 1]$, and after that point evolve via a random walk with additive perturbations chosen from a normal distribution of mean 0 and $\sigma = 0.02$. This corresponds to an oblivious adversary, which makes cross-algorithm comparisons fair. In practice, this data is "almost" stochastic, and hence algorithms like AUER and ϵ-greedy actually perform

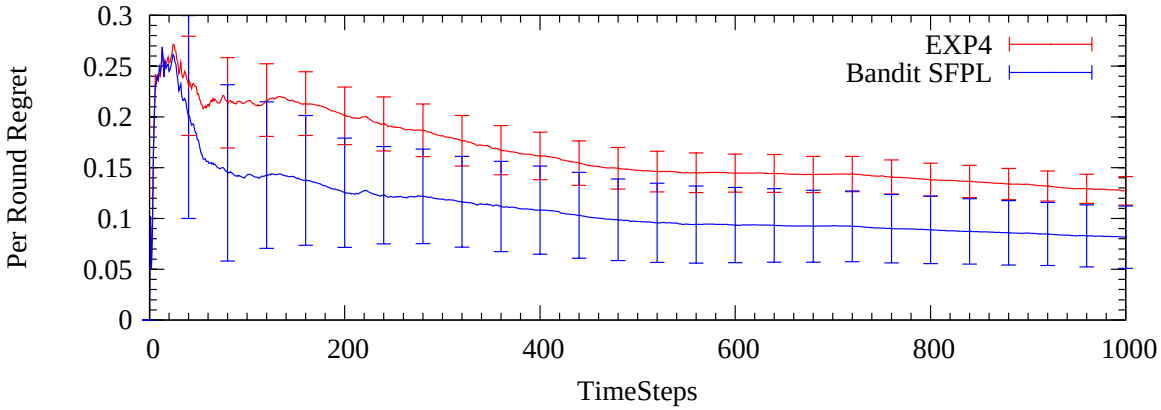

Figure 4: Average per-round regret of EXP4 and BSFPL on a fixed synthetic dataset.

quite well when appropriately tuned; however, because we believe that for real-world data the stochastic rewards assumption is unrealistic, we do not include a direct comparison to such algorithms.

Figure 4 compares EXP4 and BSFPL on a representative 1000 timestep dataset sampled from the above model. Both the available action set and the rewards were fixed. We then performed 200 runs of each algorithm. Data points correspond to the mean per-round regret measured after t timesteps; error bars represent the the variance introduced by the internal randomness of each algorithm. However, we ran this same experiment for many data sets generated as described above, and the results were very similar.

4 Conclusions

We have introduced the first polynomial-time no-regret algorithms for the stochastic availability, adversarial reward problem. The EWSA algorithm achieves essentially the best possible regret in the full-observation setting; the BSFPL algorithm for the bandit setting does not have a matching lower bound, but runs in polynomial time per round (unlike EXP4) and also performs better in practice on at least some datasets. The bounds proved for BSFPL may not be optimal; in particular, it may be possible to get improved bounds using recent results Abernethy et al. (2008).

Our work leaves open several interesting questions. We conjecture that the EWSA algorithm can be extended to the bandit setting, likely yielding better real-world performance and tighter bounds than BSFPL; however, proving regret bounds for such a generalization will likely require new proof techniques. It should also be possible to extend this work to limited action availability in the geometric setting, allowing one to address applications like shortest path problems where certain edges are stochastically unavailable.

References

J. Abernethy, E. Hazan, and A. Rakhlin. Competing in the dark: An efficient algorithm for bandit linear optimization. In *COLT*. Springer, 2008.

P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal of Computing*, 32(1):48–77, 2003.

A. Blum and Y. Mansour. From external to internal regret. *Journal of Machine Learning Research*, 8:1307–1324, 2007.

N. Cesa-Bianchi and G. Lugosi. *Prediction, Learning and Games*. Cambridge University Press, New York, NY, USA, 2006.

F. Chung and L. Lu. Concentration inequalities and martingale inequalities : A survey. *Internet Mathematics*, 3 (1):79–127, 2006.

V. Dani and T. P. Hayes. Robbing the bandit: Less regret in online geometric optimization against an adaptive adversary. In *SODA*, pages 937–943, New York, NY, USA, 2006. ACM.

E. Even-Dar, S. Mannor, and Y. Mansour. Pac bounds for multi-armed bandit and markov decision processes. pages 255–270, 2002.

Y. Freund, R. E. Schapire, Y. Singer, and M. K. Warmuth. Using and combining predictors that specialize. In *STOC*, New York, NY, USA, 1997. ACM.

J. Hannan. Approximation to bayes risk in repeated plays. *Contributions to the Theory of Games*, 3:97–139, 1957.

A. Kalai and S. Vempala. Efficient algorithms for online decision problems. *J. Comput. Syst. Sci.*, 71(3):291–307, 2005.

R. D. Kleinberg, A. Niculescu-Mizil, and Y. Sharma. Regret bounds for sleeping experts and bandits. In *COLT*. Springer, 2008.

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

H. B. McMahan and A. Blum. Online geometric optimization in the bandit setting against an adaptive adversary. In *COLT*, pages 109–123. Springer, 2004.

A Omitted Proofs

Lemma 1. *Fix an arbitrary distribution* $\Pr_{\text{avail}}$ *on the possible* $\mathcal{A}^t$ *and a vector* $v \in \mathbb{R}^n$*, and consider the action list* $\sigma = \text{sort}(v)$*. Then* $\ell(\sigma) \cdot v = \max_{x \in L} x \cdot v$*, where* ℓ *is defined with respect to* $\Pr_{\text{avail}}$*.*

Proof. Suppose the contrary. Consider the set of $\{\tau \mid \ell(\tau) \cdot v = \max_{x \in L} x \cdot v\}$; from this set, choose τ to be an element that has a maximum-length prefix shared with σ . Let $k \geq 1$ be the lowest index such that $\sigma_k \neq \tau_k$. Then, $\tau_l = \sigma_k$ for some $l > k$.

Consider τ', the permutation obtained by swapping τ_{l-1} and τ_l. Since $\sigma_k = \tau_l$, and for all indices smaller than k the two permutations agree, it must be that the element at τ_{l-1} is ranked lower than τ_l in σ. Hence, by definition of σ, $v[\tau_l] \geq v[\tau_{l-1}]$. Recall the interpretation of $\ell(\tau)[a]$ as the probability that the action a is played, if we play according to strategy of action list τ. The only difference between strategies τ and τ' is that when both τ_l and τ_{l-1} are available and none of the actions ranked higher than them (in τ) are available, τ would play τ_{l-1}, and τ' would play τ_l. Since $v[\tau_l] \geq v[\tau_{l-1}]$, it must be the case that $\ell(\tau') \cdot v \geq \ell(\tau) \cdot v$. This procedure can then be repeated until action τ_l is in position k, contradicting the choice of τ as having the maximum common prefix with σ. □

Lemma 4. *Assume that for each action* a*, it holds that* $1 - \xi \leq \frac{\hat{p}[a]}{p[a]} \leq \frac{1}{1-\xi}$ *and that* $\alpha \geq 1$*, then*

$$\mathbb{E}\left[\sum_{t=1}^{T_1} r^t[\grave{a}^t]\right] \geq (1-\xi)\mathbb{E}\left[\sum_{t=1}^{T_1} \ell(\hat{\sigma}^t) \cdot \hat{r}^t\right] - 2\alpha\gamma T_1$$

Proof. Let $\mathcal{H}^t$ denote the history of all random choices made prior to (but not including) round t. This includes the random choices made by the algorithm and the random choices of the environment. Then for a fixed action j:

$$\begin{aligned}\mathbb{E}\left[\hat{r}^t[j] | \mathcal{H}^t\right] &= \gamma\alpha + r^t[j]\frac{p[j]}{\hat{p}[j]} \\ &\leq \frac{1}{1-\xi} r^t[j] + \gamma\alpha \qquad (6)\end{aligned}$$

Given $\mathcal{H}^t$, r^t is fixed (we assume our adversary is a deterministic function on the algorithms passed choices). Further, $\ell(\hat{\sigma}^t)$ depends only on the internal randomization of the black-box algorithm, whereas $\hat{r}^t$ depends on the independent random bits used by the algorithm to decide exploration and the random bits of the environment. Thus, we get the following:

$$\begin{aligned}&\mathbb{E}\left[\ell(\hat{\sigma}^t) \cdot \hat{r}^t \mid \mathcal{H}^t\right] \\ &\quad= \mathbb{E}\left[\ell(\hat{\sigma}^t) \mid \mathcal{H}^t\right] \cdot \mathbb{E}\left[\hat{r}^t \mid \mathcal{H}^t\right] \\ &\quad\leq \mathbb{E}\left[\ell(\hat{\sigma}^t) \mid \mathcal{H}^t\right] \cdot \left(\frac{1}{1-\xi} r^t + \gamma\alpha\mathbf{1}\right) \\ &\quad= \frac{1}{1-\xi}\mathbb{E}\left[\ell(\hat{\sigma}^t) \cdot r^t \mid \mathcal{H}^t\right] + \gamma\alpha\end{aligned}$$

because $\ell(\hat{\sigma}^t) \cdot \mathbf{1} = 1$, and so

$$\begin{aligned}&\mathbb{E}\left[\ell(\hat{\sigma}^t) \cdot r^t \mid \mathcal{H}^t\right] \\ &\quad\geq (1-\xi)\mathbb{E}\left[\hat{r}^t[\hat{\sigma}^t(\mathcal{A}^t)] | \mathcal{H}^t\right] - \gamma\alpha\end{aligned}$$

Since on $(1-\gamma)$ fraction of the rounds $\grave{a}^t = \hat{\sigma}^t(\mathcal{A}^t)$ we have,

$$\begin{aligned}&\mathbb{E}\left[r^t[\grave{a}^t] \mid \mathcal{H}^t\right] \\ &\quad\geq (1-\gamma)\mathbb{E}\left[r^t[\hat{\sigma}^t(\mathcal{A}^t)] \mid \mathcal{H}^t\right] \\ &\quad= (1-\gamma)\mathbb{E}\left[\ell(\hat{\sigma}^t) \cdot r^t \mid \mathcal{H}^t\right] \qquad (7)\end{aligned}$$

since $\|\ell(\hat{\sigma}^t)\|_1 = 1$ and $\|r^t\|_\infty \leq 1$ we have,

$$\begin{aligned}&\geq \mathbb{E}\left[\ell(\hat{\sigma}^t) \cdot r^t \mid \mathcal{H}^t\right] - \gamma \\ &\geq (1-\xi)\mathbb{E}\left[\ell(\hat{\sigma}^t) \cdot \hat{r}^t \mid \mathcal{H}^t\right] - \gamma\alpha - \gamma \\ &\geq (1-\xi)\mathbb{E}\left[\ell(\hat{\sigma}^t) \cdot \hat{r}^t \mid \mathcal{H}^t\right] - 2\alpha\gamma. \qquad (8)\end{aligned}$$

Here (7) holds because r^t does not depend upon the choices of the environment on round t (since the adversary can't see these). In step (8) we use the fact that $\alpha \geq 1$. Finally, observe

$$\begin{aligned}\mathbb{E}\left[\sum_{t=1}^{T_1} r^t[\grave{a}^t]\right] &= \sum_{t=1}^{T_1} \mathbb{E}\left[r^t[\grave{a}^t]\right] \\ &= \sum_{t=1}^{T_1} \mathbb{E}_{\mathcal{H}^t}\left[\mathbb{E}[r^t[\grave{a}^t] \mid \mathcal{H}^t]\right]\end{aligned}$$

and so the result follows by an application of Equation (8). □

The following proof uses ideas which are similar to those in (Dani and Hayes, 2006) (see Proof of (2)).

Lemma 5. *Let* $\hat{\mathbf{r}} = (\hat{r}^1, \ldots, \hat{r}^{T_1})$ *be a sequence of reward vectors the algorithm passes to black-box* SFPL_ϵ*, and assume* $\hat{p}[a] \geq \beta$ *and* $\alpha \leq 1/(\beta\gamma)$*. Then*

$$\mathbb{E}\left[\sum_{t=1}^{T_1} \ell(\hat{\sigma}^t) \cdot \hat{r}^t\right] \geq \mathbb{E}\left[\max_{\hat{\sigma}} \sum_{t=1}^{T_1} \ell(\hat{\sigma}) \cdot \hat{r}^t\right] - 4\sqrt{2}\frac{n}{\beta}\sqrt{\frac{T_1}{\gamma}}$$

The expectation is over all the random choices of the algorithm and the environment (on which $\hat{r}^t$ *may depend).*

Proof. For a fixed sequence $\hat{\mathbf{r}} = (\hat{r}^1, \ldots, \hat{r}^{T_1})$ define regret($\hat{\mathbf{r}}$) by

$$\max_{\hat{\sigma}} \sum_{t=1}^{T_1} \ell(\hat{\sigma}) \cdot \hat{r}^t - \mathbb{E}\left[\sum_{t=1}^{T_1} \ell(\hat{\sigma}^t) \cdot \hat{r}^t\right]$$

In the equation above, the expectation is only over the internal randomizations of the black-box algorithm and not over the random choices of the master algorithm or the environment on which $\hat{\mathbf{r}}$ may depend.

Let l denote the (random) number of non-zero vectors among $(\hat{r}^1, \ldots, \hat{r}^{T_1})$, we have $\mathbb{E}[\text{regret}(\hat{\mathbf{r}})] = \mathbb{E}[\mathbb{E}[\text{regret}(\hat{\mathbf{r}})|l]]$ by Fubini's theorem. Consider a fixed value of l, and suppose that only l of $\hat{\mathbf{r}} = (\hat{r}^1, \ldots, \hat{r}^{T_1})$ are non-zero. Let $\mathcal{S}_l$ denote the set of sequences $\mathbf{z} = (z^1, \ldots, z^{T_1})$ with at most l non-zero vectors, and satisfying $\|z^t\|_1 \le n\alpha + \frac{n}{\gamma\beta}$ and $\|z^t\|_\infty \le \alpha + \frac{n}{\gamma\beta}$. Given a sequence $\mathbf{z}$, let nonzero($\mathbf{z}$) denote the subsequence of vectors which are not zero. Observe that $\hat{r}^t$ satisfies these conditions, hence $\hat{\mathbf{r}} = (\hat{r}^1, \ldots, \hat{r}^{T_1}) \in \mathcal{S}_l$. Since the choices made by the black-box sleeping experts algorithm depend only on past reward vectors, and not its past actions, an adaptive adversary has no advantage over *some* oblivious adversary. Thus we get

$$\begin{aligned}\mathbb{E}[\text{regret}(\hat{\mathbf{r}}) \mid l] &\le \max_{\mathbf{z}\in\mathcal{S}_l} \mathbb{E}[\text{regret}(\mathbf{z})] \\ &= \max_{\mathbf{z}\in\mathcal{S}_l} \mathbb{E}[\text{regret}(\text{nonzero}(\mathbf{z}))]\end{aligned}$$

The last step is a result from (Dani and Hayes, 2006, Observation 4.1). Recall that $\|z^t\|_1 \le n\alpha + \frac{n}{\gamma\beta}$ and since $\|\ell(\sigma)\|_1 \le 1$, $|\ell(\sigma) \cdot z^t| \le \|z^t\|_\infty \le \alpha + \frac{n}{\gamma\beta}$, using Lemma 2 we get

$$\begin{aligned}\mathbb{E}[\text{regret}(\text{nonzero}(\mathbf{z}))] &\le \epsilon n^2 \left(\alpha + \frac{1}{\beta\gamma}\right)^2 l + \frac{2}{\epsilon} \\ &\le \frac{4\epsilon n^2}{\gamma^2\beta^2} l + \frac{2}{\epsilon} \\ \mathbb{E}[\text{regret}(\hat{\mathbf{r}})] &= \mathbb{E}[\mathbb{E}[\text{regret}(\hat{\mathbf{r}}) \mid l]] \\ &\le \frac{4\epsilon n^2}{\gamma^2\beta^2}\mathbb{E}[l] + \frac{2}{\epsilon} \\ &\le \frac{4\epsilon n^2 T_1}{\gamma\beta^2} + \frac{2}{\epsilon} \qquad (9)\end{aligned}$$

Substituting $\epsilon = \frac{\beta}{n}\sqrt{\frac{\gamma}{2T_1}}$ we get the required result. □

The proof of Lemma 6 requires some submartingale inequalities which we cite from (Chung and Lu, 2006) (see Theorem 7.3).

Lemma 7. *Let $Z_0, \ldots, Z_n$ be a sequence of random variable dependent upon random events $\mathcal{F}_0 = \emptyset, \ldots, \mathcal{F}_n$, such that for $1 \le i \le n$,*

(i) $\mathbb{E}[Z_i \mid \mathcal{F}_i] \le 0$

(ii) $\mathsf{Var}[Z_i \mid \mathcal{F}_i] \le \sigma^2$

(iii) $Z_i - \mathbb{E}[Z_i \mid \mathcal{F}_i] \le a$

Then for any $\lambda > 0$,

$$\Pr\Big[\sum_{i=1}^{n} Z_i \ge Z_0 + \lambda\Big] \le \exp\Big(-\frac{\lambda^2}{2n(\sigma^2 + a^2)}\Big).$$

Lemma 6. *Assuming that $\alpha \ge 1$, $\xi \le \alpha\gamma \le 1$, and that all actions a satisfy $\hat{p}[a] \ge \beta$*

$$\begin{aligned}\mathbb{E}\left[\max_{\hat{\sigma}} \sum_{t=1}^{T_1} \ell(\hat{\sigma}) \cdot \hat{r}^t\right] &\ge \mathbb{E}\left[\max_{\sigma} \sum_{t=1}^{T_1} \ell(\sigma) \cdot r^t\right] \\ &- \sqrt{2T_1\left(5\alpha^2 + \frac{2n}{\beta} + 16\right)\log\left(\frac{1}{\beta}\right)} - \beta n T_1\end{aligned}$$

Proof. Consider,

$$\begin{aligned}&\max_{\sigma} \sum_{t=1}^{T_1} \ell(\sigma) \cdot r^t - \max_{\hat{\sigma}} \sum_{t=1}^{T_1} \ell(\hat{\sigma}) \cdot \hat{r}^t \\ &\le \max_{\sigma} \sum_{t=1}^{T_1} (\ell(\sigma) \cdot (r^t - \hat{r}^t)) \\ &\le \max_{a\in A} \sum_{t=1}^{T_1} (r^t[a] - \hat{r}^t[a]) \qquad (10)\end{aligned}$$

Where the step (10) holds because $\|\ell(\sigma)\|_1 = 1$. Let $Z^t[a] = r^t[a] - \hat{r}^t[a]$. Let $\mathcal{H}^t$ denote the history of all random choices made by the algorithm or the outcomes of the environment before time t; ; we will apply Lemma 7 to Z^t and filter $\mathcal{H}^t$. For condition (i), note that with respect to the sequence of histories $\mathcal{H}_0, \ldots, \mathcal{H}_{T_1}$, the sequence $Z^t[a]$ satisfies:

$$\begin{aligned}\mathbb{E}\left[Z^t[a] \mid \mathcal{H}^t\right] &= r^t[a] - \gamma\alpha - r^t[a]\frac{p[a]}{\hat{p}[a]} \\ &\le r^t[a](1 - (1 - \xi)) - \gamma\alpha \\ &\le r^t[a]\xi - \gamma\alpha \\ &\le \xi - \gamma\alpha \\ &\le 0\end{aligned}$$

Define $Y_t = (Z^t[a] \mid \mathcal{H}^t) - r^t[a]$, and note that $\mathsf{Var}[Z^t[a] \mid \mathcal{H}^t] = \mathsf{Var}[Y_t]$ because once $\mathcal{H}^t$ is fixed,

r^t is a constant. So, for condition (ii),

$$\begin{aligned}
\mathsf{Var}[Y_t] &= \alpha^2\gamma(1-\frac{p[a]}{n}) + \left(\alpha + \frac{nr^t[a]}{\gamma\hat{p}[a]}\right)^2 \frac{\gamma p[a]}{n} \\
&\quad - \left(\alpha\gamma + r^t[a]\frac{p[a]}{\hat{p}[a]}\right)^2 \\
&\leq \alpha^2\gamma + \frac{n(r^t[a])^2}{\hat{p}[a]}\frac{p[a]}{\hat{p}[a]} + 2\alpha(1-\gamma)r^t[a]\frac{p[a]}{\hat{p}[a]} \\
&\leq \alpha^2\gamma + 2\frac{n}{\beta} + 4\alpha \\
&\leq 5\alpha^2 + 2\frac{n}{\beta} \qquad (11)
\end{aligned}$$

Finally, for condition (iii),

$$\begin{aligned}
\mathbb{E}\left[Z^t[a] \mid \mathcal{H}^t\right] &= r^t[a] - \gamma\alpha - r^t[a]\frac{p[a]}{\hat{p}[a]} \\
&\geq r^t[a](1 - \frac{1}{1-\xi}) - \gamma\alpha \\
&\geq -2\xi r^t[a] - \gamma\alpha
\end{aligned}$$

and so,

$$Z^t[a] - \mathbb{E}\left[Z^t[a] \mid \mathcal{H}^t\right] \leq r^t[a] + 2\xi r^t[a] + \gamma\alpha \leq 4$$

Let $\lambda = \sqrt{2T_1\left(5\alpha^2 + \frac{2n}{\beta} + 16\right)\log\left(\frac{1}{\beta}\right)}$, then we have using Lemma 7:

$$\begin{aligned}
\Pr[\sum_{t=1}^{T_1} Z^t[a] \geq \lambda] &\leq \exp(-\frac{\lambda^2}{2T_1(5\alpha^2 + \frac{2n}{\beta} + 16)}) \\
&\leq \beta.
\end{aligned}$$

Hence with probability $1 - n\beta$ it holds that

$$\max_{a\in\mathcal{A}} \sum_{t=1}^{T_1} (r^t[a] - \hat{r}^t[a]) \leq \sqrt{2T_1\left(5\alpha^2 + \frac{2n}{\beta} + 16\right)}.$$

Finally,

$$\begin{aligned}
&\mathbb{E}\left[\max_{\sigma} \sum_{t=1}^{T_1} \ell(\sigma)\cdot r^t\right] - \mathbb{E}\left[\max_{\hat{\sigma}} \sum_{t=1}^{T_1} \ell(\hat{\sigma}^t)\cdot \hat{r}^t\right] \\
&\quad \leq (1-n\beta)\sqrt{2T_1\left(5\alpha^2 + \frac{2n}{\beta} + 16\right)} + n\beta T_1.
\end{aligned}$$

□