The non-stochastic multi-armed bandit problem[*]

Peter Auer
Institute for Theoretical Computer Science
Graz University of Technology
A-8010 Graz (Austria)
pauer@igi.tu-graz.ac.at

Nicolò Cesa-Bianchi
Department of Computer Science
Università di Milano
I-20135 Milano (Italy)
cesabian@dsi.unimi.it

Yoav Freund **Robert E. Schapire**
AT&T Labs
180 Park Avenue
Florham Park, NJ 07932-0971
{yoav, schapire}@research.att.com

November 18, 2001

Abstract

In the multi-armed bandit problem, a gambler must decide which arm of K non-identical slot machines to play in a sequence of trials so as to maximize his reward. This classical problem has received much attention because of the simple model it provides of the trade-off between exploration (trying out each arm to find the best one) and exploitation (playing the arm believed to give the best payoff). Past solutions for the bandit problem have almost always relied on assumptions about the statistics of the slot machines.

In this work, we make no statistical assumptions whatsoever about the nature of the process generating the payoffs of the slot machines. We give a solution to the bandit problem in which an adversary, rather than a well-behaved stochastic process, has complete control over the payoffs. In a sequence of T plays, we prove that the per-round payoff of our algorithm approaches that of the best arm at the rate $O\big(T^{-1/2}\big)$. We show by a matching lower bound that this is best possible.

We also prove that our algorithm approaches the per-round payoff of *any* set of strategies at a similar rate: if the best strategy is chosen from a pool of N strategies then our algorithm approaches the per-round payoff of the strategy at the rate $O\big((\log N)^{1/2}T^{-1/2}\big)$. Finally, we apply our results to the problem of playing an unknown repeated matrix game. We show that our algorithm approaches the minimax payoff of the unknown game at the rate $O\big(T^{-1/2}\big)$.

Keywords: adversarial bandit problem, unknown matrix games
AMS subject classification: 68Q32 68T05 91A20

[*] An early extended abstract of this paper appeared in the proceedings of the *36th Annual Symposium on Foundations of Computer Science*, pages 322–331, 1995.

1 Introduction

In the multi-armed bandit problem, originally proposed by Robbins [19], a gambler must choose which of K slot machines to play. At each time step, he pulls the arm of one of the machines and receives a reward or payoff (possibly zero or negative). The gambler's purpose is to maximize his return, i.e. the sum of the rewards he receives over a sequence of pulls. In this model, each arm is assumed to deliver rewards that are independently drawn from a fixed and unknown distribution. As reward distributions differ from arm to arm, the goal is to find the arm with the highest expected payoff as early as possible, and then to keep gambling using that best arm.

The problem is a paradigmatic example of the trade-off between exploration and exploitation. On the one hand, if the gambler plays exclusively on the machine that he thinks is best ("exploitation"), he may fail to discover that one of the other arms actually has a higher expected payoff. On the other hand, if he spends too much time trying out all the machines and gathering statistics ("exploration"), he may fail to play the best arm often enough to get a high return.

The gambler's performance is typically measured in terms of "regret". This is the difference between the expected return of the optimal strategy (pulling consistently the best arm) and the gambler's expected return. Lai and Robbins proved that the gambler's regret over T pulls can be made, for $T \to \infty$, as small as $O(\ln T)$. Furthermore, they prove that this bound is optimal in the following sense: it does not exist a strategy for the gambler with a better asymptotic performance.

Though this formulation of the bandit problem allows an elegant statistical treatment of the exploration-exploitation trade-off, it may not be adequate to model certain environments. As a motivating example, consider the task of repeatedly choosing a route for transmitting packets between two points in a communication network. To cast this scenario within the bandit problem, suppose there is a only a fixed number of possible routes and the transmission cost is reported back to the sender. Now, it is likely that the costs associated with each route cannot be modeled by a stationary distribution, so a more sophisticated set of statistical assumptions would be required. In general, it may be difficult or impossible to determine the right statistical assumptions for a given domain, and some domains may exhibit dependencies to an extent that no such assumptions are appropriate.

To provide a framework where one could model scenarios like the one sketched above, we present the adversarial bandit problem, a variant of the bandit problem in which *no* statistical assumptions are made about the generation of rewards. We only assume that each slot machine is initially assigned an arbitrary and unknown sequence of rewards, one for each time step, chosen from a bounded real interval. Each time the gambler pulls the arm of a slot-machine he receives the corresponding reward from the sequence assigned to that slot-machine. To measure the gambler's performance in this setting we replace the notion of (statistical) regret with that of worst-case regret. Given any sequence $(j_1, \ldots, j_T)$ of pulls, where $T > 0$ is an arbitrary time horizon and each j_t is the index of an arm, the worst-case regret of a gambler for this sequence of pulls is the difference between the return the gambler would have had by pulling arms $j_1, \ldots, j_T$ and the actual gambler's return, where both returns are determined by the initial assignment of rewards. It is easy to see that, in this model, the gambler cannot keep his regret small (say, sublinear in T) for *all* sequences of pulls and with respect to the worst-case assignment of rewards to the arms. Thus, to make the problem feasible, we allow the regret to depend on the "hardness" of the sequence of pulls for which it is measured, where the hardness of a sequence is roughly the number of times one has to change the slot machine currently being played in order to pull the arms in the order given by the sequence. This trick allows us to effectively control the worst-case regret *simultaneously* for all sequences of pulls, even though (as one should expect) our regret bounds become trivial when the hardness of the sequence $(j_1, \ldots, j_T)$ we compete against gets too close to T.

As a remark, note that a deterministic bandit problem was also considered by Gittins [9] and Ishikida and Varaiya [13]. However, their version of the bandit problem is very different from ours: they assume that the player can compute ahead of time exactly what payoffs will be received from each arm, and their problem is thus one of optimization, rather than exploration and exploitation.

Our most general result is a very efficient, randomized player algorithm whose expected regret for any sequence of pulls is[1] $O(S\sqrt{KT\ln(KT)})$, where S is the hardness of the sequence (see Theorem 8.1 and Corollaries 8.2, 8.4). Note that this bound holds simultaneously for all sequences of pulls, for any assignments of rewards to the arms, and uniformly over the time horizon T. If the gambler is willing to impose an upper bound S on the hardness of the sequences of pulls for which he wants to measure his regret, an improved bound $O(\sqrt{SKT\ln(KT)})$ on the expected regret for these sequences can be proven (see Corollaries 8.3 and 8.5).

With the purpose of establishing connections with certain results in game theory, we also look at a special case of the worst-case regret, which we call "weak regret". Given a time horizon T, call "best arm" the arm that has the highest return (sum of assigned rewards) up to time T with respect to the initial assignment of rewards. The gambler's weak regret is the difference between the return of this best arm and the actual gambler's return. In the paper we introduce a randomized player algorithm, tailored to this notion of regret, whose expected weak regret is $O(\sqrt{KG_{\max}\ln K})$, where $G_{\max}$ is the return of the best arm — see Theorem 4.1 in Section 4. As before, this bound holds for any assignments of rewards to the arms and uniformly over the choice of the time horizon T. Using a more complex player algorithm, we also prove that the weak regret is $O(\sqrt{KT\ln(KT/\delta)})$ with probability at least $1-\delta$ over the algorithm's randomization, for any fixed $\delta > 0$, see Theorems 6.3 and 6.4 in Section 6. This also implies that, asymptotically for $T \to \infty$ and K constant, the weak regret is $O(\sqrt{T(\ln T)^{1+\varepsilon}})$ with probability 1 for any fixed $\varepsilon > 0$, see Corollary 6.5.

Our worst-case bounds may appear weaker than the bounds proved using statistical assumptions, such as those shown by Lai and Robbins [14] of the form $O(\ln T)$. However, when comparing our results to those in the statistics literature, it is important to point out an important difference in the asymptotic quantification. In the work of Lai and Robbins the assumption is that the distribution of rewards that is associated with each arm is *fixed* as the total number of iterations T increases to infinity. In contrast, our bounds hold for any finite T, and, by the generality of our model, these bounds are applicable when the payoffs are randomly (or adversarially) chosen in a manner that does depend on T. It is this quantification order, and not the adversarial nature of our framework, which is the cause for the apparent gap. We prove this point in Theorem 5.1 where we show that, for *any* player algorithm for the K-armed bandit problem and for any T, there exists a set of K reward distributions such that the expected weak regret of the algorithm when playing on these arms for T time steps is $\Omega(\sqrt{KT})$.

So far we have considered notions of regret that compare the return of the gambler to the return of a sequence of pulls or to the return of the best arm. A further notion of regret which

[1] Though in this introduction we use the compact asymptotic notation, our bounds are proven for each finite T and almost always with explicit constants.

we explore is the regret for the best strategy in a given set of strategies that are available to the gambler. The notion of "strategy" generalizes that of "sequence of pulls": at each time step a strategy gives a recommendation, in the form of a probability distribution over the K arms, as to which arm to play next. Given an assignment of rewards to the arms and a set of N strategies for the gambler, call "best strategy" the strategy that yields the highest return with respect to this assignment. Then the regret for the best strategy is the difference between the return of this best strategy and the actual gambler's return. Using a randomized player that combines the choices of the N strategies (in the same vein as the algorithms for "prediction with expert advice" from [3]), we show that the expected regret for the best strategy is $O(\sqrt{KT \ln N})$ — see Theorem 7.1. Note that the dependence on the number of strategies is only logarithmic, and therefore the bound is quite reasonable even when the player is combining a very large number of strategies.

The adversarial bandit problem is closely related to the problem of learning to play an unknown N-person finite game, where the same game is played repeatedly by N players. A desirable property for a player is Hannan-consistency, which is similar to saying (in our bandit framework) that the weak regret per time step of the player converges to 0 with probability 1. Examples of Hannan-consistent player strategies have been provided by several authors in the past (see [18] for a survey of these results). By applying (slight extensions of) Theorems 6.3 and 6.4, we can prove provide an example of a simple Hannan-consistent player whose convergence rate is optimal up to logarithmic factors.

Our player algorithms are based in part on an algorithm presented by Freund and Schapire [6, 7], which in turn is a variant of Littlestone and Warmuth's [15] weighted majority algorithm, and Vovk's [20] aggregating strategies. In the setting analyzed by Freund and Schapire the player scores on each pull the reward of the chosen arm, but gains access to the rewards associated with *all* of the arms (not just the one that was chosen).

2 Notation and terminology

An *adversarial bandit problem* is specified by the number K of possible actions, where each action is denoted by an integer $1 \leq i \leq K$, and by an *assignment of rewards*, i.e. an infinite sequence $\boldsymbol{x}(1), \boldsymbol{x}(2), \ldots$ of vectors $\boldsymbol{x}(t) = (x_1(t), \ldots, x_K(t))$, where $x_i(t) \in [0,1]$ denotes the reward obtained if action i is chosen at time step (also called "trial") t. (Even though throughout the paper we will assume that all rewards belong to the $[0,1]$ interval, the generalization of our results to rewards in $[a,b]$ for arbitrary $a < b$ is straightforward.) We assume that the player knows the number K of actions. Furthermore, after each trial t, we assume the player only knows the rewards $x_{i_1}(1), \ldots, x_{i_t}(t)$ of the previously chosen actions $i_1, \ldots, i_t$. In this respect, we can view the player algorithm as a sequence $I_1, I_2, \ldots$, where each I_t is a mapping from the set $(\{1, \ldots, K\} \times [0,1])^{t-1}$ of action indices and previous rewards to the set of action indices.

For any reward assignment and for any $T > 0$, let

$$G_A(T) \stackrel{\text{def}}{=} \sum_{t=1}^{T} x_{i_t}(t)$$

be the *return at time horizon* T of algorithm A choosing actions $i_1, i_2, \ldots$. In what follows, we will write G_A instead of $G_A(T)$ whenever the value of T is clear from the context.

Our measure of performance for a player algorithm is the *worst-case regret*, and in this paper we explore variants of the notion of regret. Given any time horizon $T > 0$ and any sequence of actions $(j_1, \ldots, j_T)$, the (worst-case) regret of algorithm A for $(j_1, \ldots, j_T)$ is the difference

$$G_{(j_1,\ldots,j_T)} - G_A(T) \tag{1}$$

where

$$G_{(j_1,\ldots,j_T)} \stackrel{\text{def}}{=} \sum_{t=1}^{T} x_{j_t}(t)$$

is the return, at time horizon T, obtained by choosing actions $j_1, \ldots, j_T$. Hence, the regret (1) measures how much the player lost (or gained, depending on the sign of the difference) by following strategy A instead of choosing actions $j_1, \ldots, j_T$. A special case of this is the regret of A for the best single action (which we will call *weak regret* for short), defined by

$$G_{\max}(T) - G_A(T)$$

where

$$G_{\max}(T) \stackrel{\text{def}}{=} \max_j \sum_{t=1}^{T} x_j(t)$$

is the return of the single globally best action at time horizon T. As before, we will write $G_{\max}$ instead of $G_{\max}(T)$ whenever the value of T is clear from the context.

As our player algorithms will be randomized, fixing a player algorithm defines a probability distribution over the set of all sequences of actions. All the probabilities $\mathbf{P}\{\cdot\}$ and expectations $\mathbf{E}[\cdot]$ considered in this paper will be taken with respect to this distribution.

In what follows, we will prove two kinds of bounds on the performance of a (randomized) player A. The first is a bound on the *expected regret*

$$G_{(j_1,\ldots,j_T)} - \mathbf{E}\left[G_A(T)\right]$$

of A for an arbitrary sequence $(j_1, \ldots, j_T)$ of actions. The second is a confidence bound on the weak regret. This has the form

$$\mathbf{P}\left\{G_{\max}(T) > G_A(T) + \varepsilon\right\} \le \delta$$

and states that, with high probability, the return of A up to time T is not much smaller than that of the globally best action.

Finally, we remark that all of our bounds hold for *any* sequence $\boldsymbol{x}(1), \boldsymbol{x}(2), \ldots$ of reward assignments, and most of them hold *uniformly* over the time horizon T (i.e., they hold for all T without requiring T as input parameter).

3 Upper bounds on the weak regret

In this section we present and analyze our simplest player algorithm, **Exp3** (which stands for "Exponential-weight algorithm for Exploration and Exploitation"). We will show a bound on the

```
Algorithm Exp3
Parameters: Real $\gamma \in (0,1]$
Initialization: $w_i(1) = 1$ for $i = 1, \ldots, K$.

For each $t = 1, 2, \ldots$
    1. Set
           $p_i(t) = (1-\gamma)\frac{w_i(t)}{\sum_{j=1}^{K} w_j(t)} + \frac{\gamma}{K} \qquad i = 1, \ldots, K.$
    2. Draw $i_t$ randomly accordingly to the probabilities $p_1(t), \ldots, p_K(t)$.
    3. Receive reward $x_{i_t}(t) \in [0,1]$.
    4. For $j = 1, \ldots, K$ set
           $\hat{x}_j(t) = \begin{cases} x_j(t)/p_j(t) & \text{if } j = i_t \\ 0 & \text{otherwise,} \end{cases}$
           $w_j(t+1) = w_j(t)\ \exp\left(\gamma \hat{x}_j(t)/K\right)$ .
```

Figure 1: Pseudo-code of algorithm **Exp3** for the weak regret.

expected regret of **Exp3** with respect to the single best action. In the next sections, we will greatly strengthen this result.

The algorithm **Exp3**, described in Figure 1, is a variant of the algorithm **Hedge** introduced by Freund and Schapire [6] for solving a different worst-case sequential allocation problem. On each time step t, **Exp3** draws an action i_t according to the distribution $p_1(t), \ldots, p_K(t)$. This distribution is a mixture of the uniform distribution and a distribution which assigns to each action a probability mass exponential in the estimated cumulative reward for that action. Intuitively, mixing in the uniform distribution is done to make sure that the algorithm tries out all K actions and gets good estimates of the rewards for each. Otherwise, the algorithm might miss a good action because the initial rewards it observes for this action are low and large rewards that occur later are not observed because the action is not selected.

For the drawn action i_t, **Exp3** sets the estimated reward $\hat{x}_{i_t}(t)$ to $x_{i_t}(t)/p_{i_t}(t)$. Dividing the actual gain by the probability that the action was chosen compensates the reward of actions that are unlikely to be chosen. This choice of estimated rewards guarantees that their expectations are equal to the actual rewards for each action; that is, $\mathbf{E}[\hat{x}_j(t) \mid i_1, \ldots, i_{t-1}] = x_j(t)$, where the expectation is taken with respect to the random choice of i_t at trial t given the choices $i_1, \ldots, i_{t-1}$ in the previous $t-1$ trials.

We now give the first main theorem of this paper, which bounds the expected weak regret of algorithm **Exp3**.

Theorem 3.1 *For any $K > 0$ and for any $\gamma \in (0, 1]$,*

$$G_{\max} - \mathbf{E}[G_{\mathbf{Exp3}}] \leq (e-1)\gamma G_{\max} + \frac{K \ln K}{\gamma}$$

holds for any assignment of rewards and for any $T > 0$.

To understand this theorem, it is helpful to consider a simpler bound which can be obtained by an appropriate choice of the parameter γ.

Corollary 3.2 *For any $T > 0$, assume that $g \geq G_{\max}$ and that algorithm* **Exp3** *is run with input parameter*

$$\gamma = \min\left\{1, \sqrt{\frac{K \ln K}{(e-1)g}}\right\}.$$

Then

$$G_{\max} - \mathbf{E}[G_{\mathbf{Exp3}}] \leq 2\sqrt{e-1}\sqrt{gK \ln K} \leq 2.63\sqrt{gK \ln K}$$

holds for any assignment of rewards.

Proof. If $g \leq (K \ln K)/(e-1)$, then the bound is trivial since the expected regret cannot be more than g. Otherwise, by Theorem 3.1, the expected regret is at most

$$(e-1)\gamma G_{\max} + \frac{K \ln K}{\gamma} = 2\sqrt{e-1}\sqrt{gK \ln K}$$

as desired. □

To apply Corollary 3.2, it is necessary that an upper bound g on $G_{\max}(T)$ be available for tuning γ. For example, if the time horizon T is known then, since no action can have payoff greater than 1 on any trial, we can use $g = T$ as an upper bound. In Section 4, we give a technique that does not require prior knowledge of such an upper bound, yielding a result which holds uniformly over T.

If the rewards $x_i(t)$ are in the range $[a, b]$, $a < b$, then **Exp3** can be used after the rewards have been translated and rescaled to the range $[0, 1]$. Applying Corollary 3.2 with $g = T$ gives the bound $(b-a)2\sqrt{e-1}\sqrt{TK \ln K})$ on the regret. For instance, this is applicable to a standard loss model where the "rewards" fall in the range $[-1, 0]$.

Proof of Theorem 3.1. Here (and also throughout the paper without explicit mention) we use the following simple facts, which are immediately derived from the definitions,

$$\begin{aligned}
\hat{x}_i(t) &\leq 1/p_i(t) \leq K/\gamma & (2)\\
\sum_{i=1}^{K} p_i(t)\hat{x}_i(t) &= p_{i_t}(t)\frac{x_{i_t}(t)}{p_{i_t}(t)} = x_{i_t}(t) & (3)\\
\sum_{i=1}^{K} p_i(t)\hat{x}_i(t)^2 &= p_{i_t}(t)\frac{x_{i_t}(t)}{p_{i_t}(t)}\hat{x}_{i_t}(t) \leq \hat{x}_{i_t}(t) = \sum_{i=1}^{K} \hat{x}_i(t)\,. & (4)
\end{aligned}$$

Let $W_t = w_1(t) + \ldots + w_K(t)$. For all sequences $i_1, \ldots, i_T$ of actions drawn by $\mathbf{Exp3}$,

$$\begin{aligned}
\frac{W_{t+1}}{W_t} &= \sum_{i=1}^{K} \frac{w_i(t+1)}{W_t} \\
&= \sum_{i=1}^{K} \frac{w_i(t)}{W_t} \exp\left(\frac{\gamma}{K}\hat{x}_i(t)\right) \\
&= \sum_{i=1}^{K} \frac{p_i(t) - \gamma/K}{1-\gamma} \exp\left(\frac{\gamma}{K}\hat{x}_i(t)\right) \qquad (5) \\
&\le \sum_{i=1}^{K} \frac{p_i(t) - \gamma/K}{1-\gamma} \left[1 + \frac{\gamma}{K}\hat{x}_i(t) + (e-2)\left(\frac{\gamma}{K}\hat{x}_i(t)\right)^2\right] \qquad (6) \\
&\le 1 + \frac{\gamma/K}{1-\gamma} \sum_{i=1}^{K} p_i(t)\hat{x}_i(t) + \frac{(e-2)(\gamma/K)^2}{1-\gamma} \sum_{i=1}^{K} p_i(t)\hat{x}_i(t)^2 \qquad (7) \\
&\le 1 + \frac{\gamma/K}{1-\gamma} x_{i_t}(t) + \frac{(e-2)(\gamma/K)^2}{1-\gamma} \sum_{i=1}^{K} \hat{x}_i(t). \qquad (8)
\end{aligned}$$

Eq. (5) uses the definition of $p_i(t)$ in Figure 1. Eq. (6) uses the fact that $e^x \le 1 + x + (e-2)x^2$ for $x \le 1$; the expression in the preceding line is at most 1 by Eq. (2). Eq. (8) uses Eqs. (3) and (4). Taking logarithms and using $1 + x \le e^x$ gives

$$\ln \frac{W_{t+1}}{W_t} \le \frac{\gamma/K}{1-\gamma} x_{i_t}(t) + \frac{(e-2)(\gamma/K)^2}{1-\gamma} \sum_{i=1}^{K} \hat{x}_i(t).$$

Summing over t we then get

$$\ln \frac{W_{T+1}}{W_1} \le \frac{\gamma/K}{1-\gamma} G_{\mathbf{Exp3}} + \frac{(e-2)(\gamma/K)^2}{1-\gamma} \sum_{t=1}^{T} \sum_{i=1}^{K} \hat{x}_i(t) \ . \qquad (9)$$

For any action j,

$$\ln \frac{W_{T+1}}{W_1} \ge \ln \frac{w_j(T+1)}{W_1} = \frac{\gamma}{K} \sum_{t=1}^{T} \hat{x}_j(t) - \ln K.$$

Combining with Eq. (9), we get

$$G_{\mathbf{Exp3}} \ge (1-\gamma) \sum_{t=1}^{T} \hat{x}_j(t) - \frac{K \ln K}{\gamma} - (e-2)\frac{\gamma}{K} \sum_{t=1}^{T} \sum_{i=1}^{K} \hat{x}_i(t) \ . \qquad (10)$$

We next take the expectation of both sides of (10) with respect to the distribution of $\langle i_1, \ldots, i_T \rangle$. For the expected value of each $\hat{x}_i(t)$, we have:

$$\mathbf{E}[\hat{x}_i(t) \mid i_1, \ldots, i_{t-1}] = \mathbf{E}\left[p_i(t) \cdot \frac{x_i(t)}{p_i(t)} + (1 - p_i(t)) \cdot 0\right] = x_i(t) \ . \qquad (11)$$

Combining (10) and (11), we find that

$$\mathbf{E}[G_{\mathbf{Exp3}}] \geq (1-\gamma)\sum_{t=1}^{T} x_j(t) - \frac{K\ln K}{\gamma} - (e-2)\frac{\gamma}{K}\sum_{t=1}^{T}\sum_{i=1}^{K} x_i(t) \; .$$

Since j was chosen arbitrarily and

$$\sum_{t=1}^{T}\sum_{i=1}^{K} x_i(t) \leq K\, G_{\max}$$

we obtain the inequality in the statement of the theorem. □

Additional notation. As our other player algorithms will be variants of **Exp3**, we find it convenient to define some further notation based on the quantities used in the analysis of **Exp3**.

For each $1 \leq i \leq K$ and for each $t \geq 1$ define

$$\begin{aligned} G_i(t+1) &\stackrel{\text{def}}{=} \sum_{s=1}^{t} x_i(s) \\ \hat{G}_i(t+1) &\stackrel{\text{def}}{=} \sum_{s=1}^{t} \hat{x}_i(s) \\ \hat{G}_{\max}(t+1) &\stackrel{\text{def}}{=} \max_{1\leq i\leq K} \hat{G}_i(t+1) \end{aligned}$$

4 Bounds on the weak regret that hold uniformly over time

In Section 3, we showed that **Exp3** yields an expected regret of $O(\sqrt{Kg\ln K})$ whenever an upper bound g on the return $G_{\max}$ of the best action is known in advance. A bound of $O(\sqrt{KT\ln K})$, which holds uniformly over T, could be easily proven via the "guessing techniques" which will be used to prove Corollaries 8.4 and 8.5 in Section 8. In this section, instead, we describe an algorithm, called **Exp3.1**, whose expected weak regret is $O(\sqrt{KG_{\max}\ln K})$ uniformly over T. As $G_{\max} = G_{\max}(T) \leq T$, this bound is never worse than $O(\sqrt{KT\ln K})$ and is substantially better whenever the return of the best arm is small compared to T.

Our algorithm **Exp3.1**, described in Figure 2, proceeds in *epochs*, where each epoch consists of a sequence of trials. We use $r = 0, 1, 2, \ldots$ to index the epochs. On epoch r, the algorithm "guesses" a bound g_r for the return of the best action. It then uses this guess to tune the parameter γ of **Exp3**, restarting **Exp3** at the beginning of each epoch. As usual, we use t to denote the current time step.[2] **Exp3.1** maintains an estimate $\hat{G}_i(t+1)$ of the return of each action i. Since $\mathbf{E}[\hat{x}_i(t)] = x_i(t)$, this estimate will be unbiased in the sense that $\mathbf{E}[\hat{G}_i(t+1)] = G_i(t+1)$ for all i and t. Using these estimates, the algorithm detects (approximately) when the actual gain of some action has advanced beyond g_r. When this happens, the algorithm goes on to the next epoch, restarting **Exp3** with a larger bound on the maximal gain.

[2] Note that, in general, this t may differ from the "local variable" t used by **Exp3** which we now regard as a subroutine. Throughout this section, we will only use t to refer to the total number of trials as in Figure 2.

```
Algorithm Exp3.1
Initialization: Let $t = 1$, and $\hat{G}_i(1) = 0$ for $i = 1, \ldots, K$

Repeat for $r = 0, 1, 2, \ldots$
    1. Let $g_r = (K \ln K)/(e-1)\ 4^r$.
    2. Restart Exp3 choosing $\gamma_r = \min\left\{1, \sqrt{\frac{K \ln K}{(e-1)g_r}}\right\}$.
    3. While $\max_i \hat{G}_i(t) \le g_r - K/\gamma_r$ do:
        (a) Let $i_t$ be the random action chosen by Exp3 and $x_{i_t}(t)$ the corresponding reward.
        (b) $\hat{G}_i(t+1) = \hat{G}_i(t) + \hat{x}_i(t)$ for $i = 1, \ldots, K$.
        (c) $t := t + 1$
```

Figure 2: Pseudo-code of algorithm **Exp3.1** to control the weak regret uniformly over time.

The performance of the algorithm is characterized by the following theorem which is the main result of this section.

Theorem 4.1 *For any $K > 0$,*

$$\begin{aligned} G_{\max} - \mathbf{E}[G_{\mathbf{Exp3.1}}] &\le 8\sqrt{e-1}\sqrt{G_{\max} K \ln K} + 8(e-1)K + 2K \ln K \\ &\le 10.5\ \sqrt{G_{\max} K \ln K} + 13.8\ K + 2K \ln K \end{aligned}$$

holds for any assignment of rewards and for any $T > 0$.

The proof of the theorem is divided into two lemmas. The first bounds the regret suffered on each epoch, and the second bounds the total number of epochs.

Fix T arbitrarily and define the following random variables: Let R be the total number of epochs (i.e., the final value of r). Let S_r and T_r be the first and last time steps completed on epoch r (where, for convenience, we define $T_R = T$). Thus, epoch r consists of trials $S_r, S_r + 1, \ldots, T_r$. Note that, in degenerate cases, some epochs may be empty in which case $S_r = T_r + 1$. Let $\hat{G}_{\max} = \hat{G}_{\max}(T+1)$.

Lemma 4.2 *For any action j and for every epoch r,*

$$\sum_{t=S_r}^{T_r} x_{i_t}(t) \ge \sum_{t=S_r}^{T_r} \hat{x}_j(t) - 2\sqrt{e-1}\sqrt{g_r K \ln K}\ .$$

Proof. If $S_r > T_r$ (so that no trials occur on epoch r), then the lemma holds trivially since both summations will be equal to zero. Assume then that $S_r \le T_r$. Let $g = g_r$ and $\gamma = \gamma_r$. We use (10) from the proof of Theorem 3.1:

$$\sum_{t=S_r}^{T_r} x_{i_t}(t) \ge \sum_{t=S_r}^{T_r} \hat{x}_j(t) - \gamma \sum_{t=1}^{T_r} \hat{x}_j(t) - \frac{K \ln K}{\gamma} - (e-2)\frac{\gamma}{K} \sum_{t=S_r}^{T_r} \sum_{i=1}^{K} \hat{x}_i(t) \, .$$

From the definition of the termination condition we know that $\hat{G}_i(T_r) \le g - K/\gamma$. Using (2), we get $\hat{x}_i(t) \le K/\gamma$. This implies that $\hat{G}_i(T_r + 1) \le g$ for all i. Thus,

$$\sum_{t=S_r}^{T_r} x_{i_t}(t) \ge \sum_{t=S_r}^{T_r} \hat{x}_j(t) - g\left(\gamma + \gamma(e-2)\right) - \frac{K \ln K}{\gamma} \, .$$

By our choice for γ, we get the statement of the lemma. □

The next lemma gives an implicit upper bound on the number of epochs R. Let $c = (K \ln K)/(e - 1)$.

Lemma 4.3 *The number of epochs R satisfies*

$$2^{R-1} \le \frac{K}{c} + \sqrt{\frac{\hat{G}_{\max}}{c}} + \frac{1}{2} \, .$$

Proof. If $R = 0$, then the bound holds trivially. So assume $R \ge 1$. Let $z = 2^{R-1}$. Because epoch $R - 1$ was completed, by the termination condition,

$$\hat{G}_{\max} \ge \hat{G}_{\max}(T_{R-1} + 1) > g_{R-1} - \frac{K}{\gamma_{R-1}} = c\, 4^{R-1} - K\, 2^{R-1} = cz^2 - Kz \, . \tag{12}$$

Suppose the claim of the lemma is false. Then $z > K/c + \sqrt{\hat{G}_{\max}/c}$. Since the function $cx^2 - Kx$ is increasing for $x > K/(2c)$, this implies that

$$cz^2 - Kz > c\left(\frac{K}{c} + \sqrt{\frac{\hat{G}_{\max}}{c}}\right)^2 - K\left(\frac{K}{c} + \sqrt{\frac{\hat{G}_{\max}}{c}}\right) = K\sqrt{\frac{\hat{G}_{\max}}{c}} + \hat{G}_{\max} \, ,$$

contradicting (12). □

Proof of Theorem 4.1. Using the lemmas, we have that

$$\begin{aligned} G_{\mathbf{Exp3.1}} = \sum_{t=1}^{T} x_{i_t}(t) \quad &= \quad \sum_{r=0}^{R} \sum_{t=S_r}^{T_r} x_{i_t}(t) \\ &\ge \quad \max_j \sum_{r=0}^{R} \left(\sum_{t=S_r}^{T_r} \hat{x}_j(t) - 2\sqrt{e-1}\sqrt{g_r K \ln K} \right) \end{aligned}$$

$$\begin{aligned}
&= \max_j \hat{G}_j(T+1) - 2K\ln K\sum_{r=0}^{R} 2^r \\
&= \hat{G}_{\max} - 2K\ln K(2^{R+1}-1) \\
&\geq \hat{G}_{\max} + 2K\ln K - 8K\ln K\left(\frac{K}{c} + \sqrt{\frac{\hat{G}_{\max}}{c}} + \frac{1}{2}\right) \\
&= \hat{G}_{\max} - 2K\ln K - 8(e-1)K - 8\sqrt{e-1}\sqrt{\hat{G}_{\max}K\ln K}\ . \qquad (13)
\end{aligned}$$

Here, we used Lemma 4.2 for the first inequality and Lemma 4.3 for the second inequality. The other steps follow from definitions and simple algebra.

Let $f(x) = x - a\sqrt{x} - b$ for $x \geq 0$ where $a = 8\sqrt{e-1}\sqrt{K\ln K}$ and $b = 2K\ln K + 8(e-1)K$. Taking expectations of both sides of (13) gives

$$\mathbf{E}[G_{\mathbf{Exp3.1}}] \geq \mathbf{E}[f(\hat{G}_{\max})]\ . \qquad (14)$$

Since the second derivative of f is positive for $x > 0$, f is convex so that, by Jensen's inequality,

$$\mathbf{E}[f(\hat{G}_{\max})] \geq f(\mathbf{E}[\hat{G}_{\max}])\ . \qquad (15)$$

Note that,

$$\mathbf{E}[\hat{G}_{\max}] = \mathbf{E}\left[\max_j \hat{G}_j(T+1)\right] \geq \max_j \mathbf{E}[\hat{G}_j(T+1)] = \max_j \sum_{t=1}^{T} x_j(t) = G_{\max}\ .$$

The function f is increasing if and only if $x > a^2/4$. Therefore, if $G_{\max} > a^2/4$ then $f(\mathbf{E}[\hat{G}_{\max}]) \geq f(G_{\max})$. Combined with (14) and (15), this gives that $\mathbf{E}[G_{\mathbf{Exp3.1}}] \geq f(G_{\max})$ which is equivalent to the statement of the theorem. On the other hand, if $G_{\max} \leq a^2/4$ then, because f is non-increasing on $[0, a^2/4]$,

$$f(G_{\max}) \leq f(0) = -b \leq 0 \leq \mathbf{E}[G_{\mathbf{Exp3.1}}]$$

so the theorem follows trivially in this case as well. □

5 Lower bounds on the weak regret

In this section, we state a lower bound on the expected weak regret of any player. More precisely, for any choice of the time horizon T we show that there exists a strategy for assigning the rewards to the actions such that the expected weak regret of any player algorithm is $\Omega(\sqrt{KT})$. Observe that this does not match the upper bound for our algorithms **Exp3** and **Exp3.1** (see Corollary 3.2 and Theorem 4.1); it is an open problem to close this gap.

Our lower bound is proven using the classical (statistical) bandit model with an crucial difference: the reward distribution depends on the number K of actions and on the time horizon T. This dependence is the reason why our lower bound does not contradict the upper bounds of the form

$O(\ln T)$ for the classical bandit model [14]. There, the distribution over the rewards is fixed as $T \to \infty$.

Note that our lower bound has a considerably stronger dependence on the number K of action than the lower bound $\Theta(\sqrt{T \ln K})$, which could have been proven directly from the results in [3, 6]. Specifically, our lower bound implies that no upper bound is possible of the form $O(T^{\alpha}(\ln K)^{\beta})$ where $0 \leq \alpha < 1, \beta > 0$.

Theorem 5.1 *For any number of actions $K \geq 2$ and for any time horizon T, there exists a distribution over the assignment of rewards such that the expected weak regret of any algorithm (where the expectation is taken with respect to both the randomization over rewards and the algorithm's internal randomization) is at least*

$$\frac{1}{20} \min\{\sqrt{KT}, T\}.$$

The proof is given in Appendix A.

The lower bound implies, of course, that for any algorithm there is a particular choice of rewards that will cause the expected weak regret (where the expectation is now with respect to the algorithm's internal randomization only) to be larger than this value.

6 Bounds on the weak regret that hold with probability 1

In Section 4 we showed that the *expected* weak regret of algorithm $\mathbf{Exp3.1}$ is $O(\sqrt{KT \ln K})$. In this section we show that a modification of $\mathbf{Exp3}$ achieves a weak regret of $O(\sqrt{KT \ln(KT/\delta)})$ with probability at least $1 - \delta$, for any fixed δ and uniformly over T. From this, a bound on the weak regret that holds with probability 1 follows easily.

The modification of $\mathbf{Exp3}$ is necessary since the variance of the regret achieved by this algorithm is large, so large that an interesting high probability bound may not hold. The large variance of the regret comes from the large variance of the estimates $\hat{x}_i(t)$ for the payoffs $x_i(t)$. In fact, the variance of $\hat{x}_i(t)$ can be close to $1/p_i(t)$ which, for γ in our range of interest, is (ignoring the dependence of K) of magnitude $\sqrt{T}$. Summing over trials, the variance of the return of $\mathbf{Exp3}$ is about $T^{3/2}$, so that the regret might be as large as $T^{3/4}$.

To control the variance we modify algorithm $\mathbf{Exp3}$ so that it uses estimates which are based on upper confidence bounds instead of estimates with the correct expectation. The modified algorithm $\mathbf{Exp3.P}$ is given in Figure 3. Let

$$\hat{\sigma}_i(t+1) \stackrel{\text{def}}{=} \sqrt{KT} + \sum_{s=1}^{t} \frac{1}{p_i(t)\sqrt{KT}} \ .$$

Whereas algorithm $\mathbf{Exp3}$ directly uses the estimates $\hat{G}_i(t)$ when choosing i_t at random, algorithm $\mathbf{Exp3.P}$ uses the upper confidence bounds $\hat{G}_i(t) + \alpha\hat{\sigma}_i(t)$. The next lemma shows that, for appropriate α, these are indeed upper confidence bounds. Fix some time horizon T. In what follows, we will use $\hat{\sigma}_i$ to denote $\hat{\sigma}_i(T+1)$ and $\hat{G}_i$ to denote $\hat{G}_i(T+1)$.

```
Algorithm Exp3.P
Parameters: Reals $\alpha > 0$ and $\gamma \in (0,1]$
Initialization: For $i = 1, \ldots, K$
    $w_i(1) = \exp\left(\frac{\alpha\gamma}{3}\sqrt{\frac{T}{K}}\right)$ .
For each $t = 1, 2, \ldots, T$
    1. For $i = 1, \ldots, K$ set
           $p_i(t) = (1-\gamma)\frac{w_i(t)}{\sum_{j=1}^{K} w_j(t)} + \frac{\gamma}{K}$ .
    2. Choose $i_t$ randomly according to the distribution $p_1(t), \ldots, p_K(t)$.
    3. Receive reward $x_{i_t}(t) \in [0,1]$.
    4. For $j = 1, \ldots, K$ set
           $\hat{x}_j(t) = \begin{cases} x_j(t)/p_j(t) & \text{if } j = i_t \\ 0 & \text{otherwise,} \end{cases}$
           $w_j(t+1) = w_j(t)\exp\left(\frac{\gamma}{3K}\left(\hat{x}_j(t) + \frac{\alpha}{p_j(t)\sqrt{KT}}\right)\right)$ .
```

Figure 3: Pseudo-code of algorithm **Exp3**.**P** achieving small weak regret with high probability.

Lemma 6.1 *If* $2\sqrt{\ln(KT/\delta)} \le \alpha \le 2\sqrt{KT}$, *then*

$$\mathbf{P}\left\{\exists i : \hat{G}_i + \alpha\hat{\sigma}_i < G_i\right\} \le \delta.$$

Proof. Fix some i and set

$$s_t \stackrel{\text{def}}{=} \frac{\alpha}{2\hat{\sigma}_i(t+1)}.$$

Since $\alpha \le 2\sqrt{KT}$ and $\hat{\sigma}_i(t+1) \ge \sqrt{KT}$, we have $s_t \le 1$. Now

$$\begin{aligned} &\mathbf{P}\left\{\hat{G}_i + \alpha\hat{\sigma}_i < G_i\right\} \\ &= \mathbf{P}\left\{\sum_{t=1}^{T}(x_i(t) - \hat{x}_i(t)) - \frac{\alpha}{2}\hat{\sigma}_i > \frac{\alpha}{2}\hat{\sigma}_i\right\} \\ &\le \mathbf{P}\left\{s_T\sum_{t=1}^{T}\left(x_i(t) - \hat{x}_i(t) - \frac{\alpha}{2p_i(t)\sqrt{KT}}\right) > \frac{\alpha^2}{4}\right\} \end{aligned} \tag{16}$$

$$
\begin{aligned}
&= \mathbf{P}\left\{\exp\left(s_T\sum_{t=1}^T\left(x_i(t)-\hat{x}_i(t)-\frac{\alpha}{2p_i(t)\sqrt{KT}}\right)\right) > \exp\left(\alpha^2/4\right)\right\} \\
&\leq e^{-\alpha^2/4}\mathbf{E}\left[\exp\left(s_T\sum_{t=1}^T\left(x_i(t)-\hat{x}_i(t)-\frac{\alpha}{2p_i(t)\sqrt{KT}}\right)\right)\right] \qquad (17)
\end{aligned}
$$

where in step (16) we multiplied both sides by s_T and used $\hat{\sigma}_i \geq \sum_{t=1}^T 1/(p_i(t)\sqrt{KT})$, while in step (17) we used Markov's inequality. For $t = 1, \ldots, T$ set

$$
Z_t \stackrel{\text{def}}{=} \exp\left(s_t\sum_{\tau=1}^t\left(x_i(\tau)-\hat{x}_i(\tau)-\frac{\alpha}{2p_i(\tau)\sqrt{KT}}\right)\right) .
$$

Then, for $t = 2, \ldots, T$

$$
Z_t = \exp\left(s_t\left(x_i(t)-\hat{x}_i(t)-\frac{\alpha}{2p_i(t)\sqrt{KT}}\right)\right)\cdot(Z_{t-1})^{\frac{s_t}{s_{t-1}}} .
$$

Denote by $\mathbf{E}_t[Z_t] = \mathbf{E}[Z_t \mid i_1, \ldots, i_{t-1}]$ the expectation of Z_t with respect to the random choice in trial t and conditioned on the past $t-1$ trials. Note that when the past $t-1$ trials are fixed the only random quantities in Z_t are the $\hat{x}_i(t)$'s. Note also that $x_i(t) - \hat{x}_i(t) \leq 1$, and that

$$
\begin{aligned}
\mathbf{E}_t\left[(x_i(t)-\hat{x}_i(t))^2\right] &= \mathbf{E}_t\left[\hat{x}_i(t)^2\right] - x_i(t)^2 \\
&\leq \mathbf{E}_t\left[\hat{x}_i(t)^2\right] \\
&= \frac{x_i(t)^2}{p_i(t)} = \frac{1}{p_i(t)} \qquad (18)
\end{aligned}
$$

Hence, for each $t = 2, \ldots, T$

$$
\begin{aligned}
\mathbf{E}_t[Z_t] &\leq \mathbf{E}_t\left[\exp s_t\left(x_i(t)-\hat{x}_i(t)-\frac{s_t}{p_i(t)}\right)\right](Z_{t-1})^{\frac{s_t}{s_{t-1}}} \qquad (19) \\
&\leq \mathbf{E}_t\left[1+s_t(x_i(t)-\hat{x}_i(t))+s_t^2(x_i(t)-\hat{x}_i(t))^2\right]\exp\left(-\frac{s_t^2}{p_i(t)}\right)(Z_{t-1})^{\frac{s_t}{s_{t-1}}} \qquad (20) \\
&\leq \left(1+s_t^2/p_i(t)\right)\exp\left(-\frac{s_t^2}{p_i(t)}\right)(Z_{t-1})^{\frac{s_t}{s_{t-1}}} \qquad (21) \\
&\leq (Z_{t-1})^{\frac{s_t}{s_{t-1}}} \qquad (22) \\
&\leq 1 + Z_{t-1}. \qquad (23)
\end{aligned}
$$

Eq. (19) uses

$$
\frac{\alpha}{2p_i(t)\sqrt{KT}} \geq \frac{\alpha}{2p_i(t)\hat{\sigma}_i(t+1)} = \frac{s_t}{p_i(t)}
$$

since $\hat{\sigma}_i(t+1) \geq \sqrt{KT}$. Eq. (20) uses $e^a \leq 1 + a + a^2$ for $a \leq 1$. Eq. (21) uses $\mathbf{E}_t[\hat{x}_i(t)] = x_i(t)$. Eq. (22) uses $1 + x \leq e^x$ for any real x. Eq. (23) uses $s_t \leq s_{t-1}$ and $z^u \leq 1 + z$ for any $z > 0$ and $u \in [0, 1]$. Observing that $\mathbf{E}[Z_1] \leq 1$, we get by induction that $\mathbf{E}[Z_T] \leq T$, and the lemma follows by our choice of α. □

The next lemma shows that the return achieved by algorithm **Exp3**.**P** is close to its upper confidence bounds. Let

$$\hat{U} \stackrel{\text{def}}{=} \max_{1 \le i \le K} \left(\hat{G}_i + \alpha \hat{\sigma}_i \right) .$$

Lemma 6.2 *If* $\alpha \le 2\sqrt{KT}$ *then*

$$G_{\mathbf{Exp3.P}} \ge \left(1 - \frac{5\gamma}{3} \right) \hat{U} - \frac{3}{\gamma} K \ln K - 2\alpha\sqrt{KT} - 2\alpha^2 .$$

Proof. We proceed as in the analysis of algorithm **Exp3**. Set $\eta = \gamma/(3K)$ and consider any sequence $i_1, \ldots, i_T$ of actions chosen by **Exp3**.**P**. As $\hat{x}_i(t) \le K/\gamma$, $p_i(t) \ge \gamma/K$, and $\alpha \le 2\sqrt{KT}$, we have

$$\eta \hat{x}_i(t) + \frac{\alpha\eta}{p_i(t)\sqrt{KT}} \le 1 .$$

Therefore,

$$\begin{aligned}
\frac{W_{t+1}}{W_t} &= \sum_{i=1}^{K} \frac{w_i(t+1)}{W_t} \\
&= \sum_{i=1}^{K} \frac{w_i(t)}{W_t} \exp\left(\eta \hat{x}_i(t) + \frac{\alpha\eta}{p_i(t)\sqrt{KT}} \right) \\
&\le \sum_{i=1}^{K} \frac{p_i(t) - \gamma/K}{1-\gamma} \exp\left(\eta \hat{x}_i(t) + \frac{\alpha\eta}{p_i(t)\sqrt{KT}} \right) \\
&\le \sum_{i=1}^{K} \frac{p_i(t) - \gamma/K}{1-\gamma} \left[1 + \eta \hat{x}_i(t) + \frac{\alpha\eta}{p_i(t)\sqrt{KT}} + 2\eta^2 \hat{x}_i(t)^2 + \frac{2\alpha^2\eta^2}{p_i(t)^2 KT} \right] \\
&\le 1 + \frac{\eta}{1-\gamma} \sum_{i=1}^{K} p_i(t)\hat{x}_i(t) + \frac{\alpha\eta}{1-\gamma} \sum_{i=1}^{K} \frac{1}{\sqrt{KT}} \\
&\quad + \frac{2\eta^2}{1-\gamma} \sum_{i=1}^{K} p_i(t)\hat{x}_i(t)^2 + \frac{2\alpha^2\eta^2}{1-\gamma} \sum_{i=1}^{K} \frac{1}{p_i(t) KT} \\
&\le 1 + \frac{\eta}{1-\gamma} x_{i_t}(t) + \frac{\alpha\eta}{1-\gamma}\sqrt{\frac{K}{T}} + \frac{2\eta^2}{1-\gamma} \sum_{i=1}^{K} \hat{x}_i(t) + \frac{2\alpha^2\eta}{1-\gamma}\frac{1}{T} .
\end{aligned}$$

The second inequality uses $e^a \le 1 + a + a^2$ for $a \le 1$, and $(a+b)^2 \le 2(a^2 + b^2)$ for any a, b. The last inequality uses Eqs. (2), (3) and (4). Taking logarithms, using $\ln(1+x) \le x$ and summing over $t = 1, \ldots, T$ we get

$$\ln \frac{W_{T+1}}{W_1} \le \frac{\eta}{1-\gamma} G_{\mathbf{Exp3.P}} + \frac{\alpha\eta}{1-\gamma}\sqrt{KT} + \frac{2\eta^2}{1-\gamma} \sum_{i=1}^{K} \hat{G}_i + \frac{2\alpha^2\eta}{1-\gamma} .$$

Since

$$\ln W_1 = \alpha\eta\sqrt{KT} + \ln K$$

and for any j

$$\ln W_{T+1} \geq \ln w_j(T+1) \geq \eta \hat{G}_j + \alpha\eta\hat{\sigma}_j$$

this implies

$$G_{\mathbf{Exp3.P}} \geq (1-\gamma)\left(\hat{G}_j + \alpha\hat{\sigma}_j\right) - \frac{1}{\eta}\ln K - 2\alpha\sqrt{KT} - 2\eta\sum_{i=1}^{K}\hat{G}_i - 2\alpha^2 \; .$$

for any j. Finally, using $\eta = \gamma/(3K)$ and

$$\sum_{i=1}^{K}\hat{G}_i \leq K\hat{U}$$

yields the lemma. □

Combining Lemmas 6.1 and 6.2 gives the main result of this section.

Theorem 6.3 *For any fixed $T > 0$, for all $K \geq 2$ and for all $\delta > 0$, if*

$$\gamma = \min\left\{\frac{3}{5}, 2\sqrt{\frac{3}{5}\frac{K\ln K}{T}}\right\} \qquad \text{and} \qquad \alpha = 2\sqrt{\ln(KT/\delta)} \; ,$$

then

$$G_{\max} - G_{\mathbf{Exp3.P}} \leq 4\sqrt{KT\ln\frac{KT}{\delta}} + 4\sqrt{\frac{5}{3}KT\ln K} + 8\ln\frac{KT}{\delta}$$

holds for any assignment of rewards with probability at least $1-\delta$.

Proof. We assume without loss of generality that $T \geq (20/3)K\ln K$ and that $\delta \geq KTe^{-KT}$. If either of these conditions do not hold, then the theorem holds trivially. Note that $T \geq (20/3)K\ln K$ ensures $\gamma \leq 3/5$. Note also that $\delta \geq KTe^{-KT}$ implies $\alpha \leq 2\sqrt{KT}$ for our choice of α. So we can apply Lemmas 6.1 and 6.2. By Lemma 6.2 we have

$$G_{\mathbf{Exp3.P}} \geq \left(1 - \frac{5\gamma}{3}\right)\hat{U} - \frac{3}{\gamma}K\ln K - 2\alpha\sqrt{KT} - 2\alpha^2 \; .$$

By Lemma 6.1 we have $\hat{U} \geq G_{\max}$ with probability at least $1-\delta$. Collecting terms and using $G_{\max} \leq T$ gives the theorem. □

It is not difficult to obtain an algorithm that does not need the time horizon T as input parameter and whose regret is only slightly worse than that proven for the algorithm $\mathbf{Exp3.P}$ in Theorem 6.3. This new algorithm, called $\mathbf{Exp3.P.1}$ and shown in Figure 4, simply restarts $\mathbf{Exp3.P}$ doubling its guess for T each time. The only careful issue is the choice of the confidence parameter δ and of the minimum length of the runs to ensure that Lemma 6.1 holds for all the runs of $\mathbf{Exp3.P}$.

```
**Algorithm Exp3.P.1**
**Parameters:** Real $0 < \delta < 1$.
**Initialization:** let $T_r = 2^r$, $\delta_r = \frac{\delta}{(r+1)(r+2)}$ and
        $r^* = \min\{r \in \mathbb{N} : \delta_r \geq KT_r e^{-KT_r}\}$ .        (24)
**Repeat for** $r = r^*, r^*+1, \ldots$
    Run **Exp3.P** for $T_r$ trials choosing $\alpha$ and $\gamma$ as in Theorem 6.3 with $T = T_r$ and $\delta = \delta_r$.
```

Figure 4: Pseudo-code of algorithm **Exp3.P.1** (see Theorem 6.4).

Theorem 6.4 *Let $K \geq 2$, $\delta \in (0,1)$ and $T \geq 2^{r^*}$. Let $c_T = 2\ln(2+\log_2 T)$, and let r^* be as in Eq. (24). Then*

$$G_{\max} - G_{\mathbf{Exp3.P.1}} \leq \frac{10}{\sqrt{2}-1}\sqrt{2KT\left(\ln\frac{KT}{\delta} + c_T\right)} + 10(1+\log_2 T)\left(\ln\frac{KT}{\delta} + c_T\right) ,$$

holds with probability at least $1-\delta$.

Proof. Choose the time horizon T arbitrarily and call *epoch* the sequence of trials between two successive restarts of algorithm **Exp3.P**.

For each $r > r^*$, where r^* is defined in (24), let

$$G_i(r) \stackrel{\text{def}}{=} \sum_{t=2^r+1}^{2^{r+1}} x_i(t) , \quad \hat{G}_i(r) \stackrel{\text{def}}{=} \sum_{t=2^r+1}^{2^{r+1}} \hat{x}_i(t) , \quad \hat{\sigma}_i(r) \stackrel{\text{def}}{=} \sqrt{KT_r} + \sum_{t=2^r+1}^{2^{r+1}} \frac{1}{p_i(t)\sqrt{KT_r}}$$

and similarly define the quantities $G_i(r^*)$ and $\hat{G}_i(r^*)$ with sums that go from $t=1$ to $t=r^*$.

For each $r \geq r^*$, we have $\delta_r \geq KT_r e^{-KT_r}$. Thus we can find numbers α_r such that, by Lemma 6.1,

$$\begin{aligned}
\mathbf{P}\left\{(\exists r \geq r^*)(\exists i) : \hat{G}_i(r) + \alpha_r\hat{\sigma}_i(r) < G_i(r)\right\} &\leq \sum_{r=r^*}^{\infty} \mathbf{P}\left\{\exists i : \hat{G}_i(r) + \alpha_r\hat{\sigma}_i(r) < G_i(r)\right\} \\
&\leq \sum_{r=0}^{\infty} \frac{\delta}{(r+1)(r+2)} \\
&= \delta .
\end{aligned}$$

We now apply Theorem 6.3 to each epoch. Without loss of generality, assume that T satisfies

$$2^{r^*+\ell-1} < T = \sum_{r=0}^{\ell-1} 2^{r^*+r} < 2^{r^*+\ell}$$

for some $\ell \geq 1$. With probability at least $1 - \delta$ over the random draw of $\mathbf{Exp3.P.1}$'s actions $i_1, \ldots, i_T$,

$$
\begin{aligned}
& G_{\max} - G_{\mathbf{Exp3.P.1}} \\
& \leq \sum_{r=0}^{\ell-1} 10 \left[\sqrt{KT_{r^*+r} \ln \frac{KT_{r^*+r}}{\delta_{r^*+r}}} + \ln \frac{KT_{r^*+r}}{\delta_{r^*+r}} \right] \\
& \leq 10 \left[\sqrt{K \ln \frac{KT_{r^*+\ell-1}}{\delta_{r^*+\ell-1}}} \sum_{r=0}^{\ell-1} \sqrt{T_{r^*+r}} + \ell \ln \frac{KT_{r^*+\ell-1}}{\delta_{r^*+\ell-1}} \right] \\
& \leq 10 \left[\sqrt{K \ln \frac{KT_{r^*+\ell-1}}{\delta_{r^*+\ell-1}}} \left(\frac{2^{(r^*+\ell)/2}}{\sqrt{2}-1} \right) + \ell \ln \frac{KT_{r^*+\ell-1}}{\delta_{r^*+\ell-1}} \right] \\
& \leq \frac{10}{\sqrt{2}-1} \sqrt{2KT \left(\ln \frac{KT}{\delta} + c_T \right)} + 10(1 + \log_2 T) \left(\ln \frac{KT}{\delta} + c_T \right)
\end{aligned}
$$

where $c_T = 2 \ln(2 + \log_2 T)$. □

From the above theorem we get, as a simple corollary, a statement about the almost sure convergence of the return of algorithm $\mathbf{Exp3.P}$. The rate of convergence is almost optimal, as one can see from our lower bound in Section 5.

Corollary 6.5 *For any $K \geq 2$ and for any function $f : \mathbb{R} \to \mathbb{R}$ with $\lim_{T\to\infty} f(T) = \infty$,*

$$\lim_{T\to\infty} \frac{G_{\max} - G_{\mathbf{Exp3.P.1}}}{\sqrt{T(\ln T) f(T)}} = 0 \; .$$

holds for any assignment of rewards with probability 1.

Proof. Let $\delta = 1/T^2$. Then, by Theorem 6.4, there exists a constant C such that for all T large enough

$$G_{\max} - G_{\mathbf{Exp3.P.1}} \leq C\sqrt{KT \ln T}$$

with probability at least $1 - 1/T^2$. This implies that

$$\mathbf{P}\left\{ \frac{G_{\max} - G_{\mathbf{Exp3.P.1}}}{\sqrt{(T \ln T) f(T)}} > C\sqrt{\frac{K}{f(T)}} \right\} \leq \frac{1}{T^2}$$

and the theorem follows from the Borel-Cantelli lemma. □

7 The regret against the best strategy from a pool

Consider a setting where the player has preliminarly fixed a set of strategies that could be used for choosing actions. These strategies might select different actions at different iterations. The strategies can be computations performed by the player or they can be external advice given to the player by "experts." We will use the more general term "expert" (borrowed from Cesa-Bianchi et al. [3]) because we place no restrictions on the generation of the advice. The player's goal in this case is to combine the advice of the experts in such a way that its return is close to that of the best expert.

Formally, we assume that the player, prior to choosing an action at time t, is provided with a set of N probability vectors $\boldsymbol{\xi}^1(t), \ldots, \boldsymbol{\xi}^N(t) \in [0,1]^K$, where $\sum_{j=1}^K \xi_j^i(t) = 1$ for each $i = 1, \ldots, N$. We interpret $\boldsymbol{\xi}^i(t)$ as the advice of expert i on trial t, where the j-th component $\xi_j^i(t)$ represents the recommended probability of playing action j. (As a special case, the distribution can be concentrated on a single action, which represents a deterministic recommendation.) If the vector of rewards at time t is $\boldsymbol{x}(t)$, then the expected reward for expert i, with respect to the chosen probability vector $\boldsymbol{\xi}^i(t)$, is simply $\boldsymbol{\xi}^i(t) \cdot \boldsymbol{x}(t)$. In analogy of $G_{\max}$, we define

$$\tilde{G}_{\max} \stackrel{\mathrm{def}}{=} \max_{1 \le i \le N} \sum_{t=1}^{T} \boldsymbol{\xi}^i(t) \cdot \boldsymbol{x}(t)$$

measuring the expected return of the best strategy. Then the *regret for the best strategy* at time horizon T, defined by $\tilde{G}_{\max}(T) - G_A(T)$, measures the difference between the return of the best expert and player's A return up to time T.

Our results hold for any finite set of experts. Formally, we regard each $\boldsymbol{\xi}^i(t)$ as a random variable which is an arbitrary function of the random sequence of plays $i_1, \ldots, i_{t-1}$. This definition allows for experts whose advice depends on the entire past history as observed by the player, as well as other side information which may be available.

We could at this point view each expert as a "meta-action" in a higher-level bandit problem with payoff vector defined at trial t as $(\boldsymbol{\xi}^1(t) \cdot \boldsymbol{x}(t), \ldots, \boldsymbol{\xi}^N(t) \cdot \boldsymbol{x}(t))$. We could then immediately apply Corollary 3.2 to obtain a bound of $O(\sqrt{gN \log N})$ on the player's regret relative to the best expert (where g is an upper bound on $\tilde{G}_{\max}$). However, this bound is quite weak if the player is combining many experts (i.e., if N is very large). We show below that the algorithm $\mathbf{Exp3}$ from Section 3 can be modified yielding a regret term of the form $O(\sqrt{gK \log N})$. This bound is very reasonable when the number of actions is small, but the number of experts is quite large (even exponential).

Our algorithm $\mathbf{Exp4}$ is shown in Figure 5, and is only a slightly modified version of $\mathbf{Exp3}$. ($\mathbf{Exp4}$ stands for "Exponential-weight algorithm for Exploration and Exploitation using Expert advice.") Let us define $\boldsymbol{y}(t) \in [0,1]^N$ to be the vector with components corresponding to the gains of the experts: $y_i(t) = \boldsymbol{\xi}^i(t) \cdot \boldsymbol{x}(t)$.

The simplest possible expert is one which always assigns uniform weight to all actions so that $\xi_j(t) = 1/K$ on each round t. We call this the *uniform expert*. To prove our results, we need to assume that the uniform expert is included in the family of experts.[3] Clearly, the uniform expert can always be added to any given family of experts at the very small expense of increasing N by one.

Theorem 7.1 *For any $K, T > 0$, for any $\gamma \in (0,1]$, and for any family of experts which includes*

[3] In fact, we can use a slightly weaker sufficient condition, namely, that the uniform expert is included in the convex hull of the family of experts, i.e., that there exists nonnegative numbers $\alpha_1, \ldots, \alpha_N$ with $\sum_{j=1}^N \alpha_j = 1$ such that, for all t and all i, $\sum_{j=1}^N \alpha_j \xi_i^j(t) = 1/K$.

```
Algorithm Exp4
Parameters: Real $\gamma \in (0,1]$
Initialization: $w_i(1) = 1$ for $i = 1, \ldots, N$.

For each $t = 1, 2, \ldots$
    1. Get advice vectors $\boldsymbol{\xi}^1(t), \ldots, \boldsymbol{\xi}^N(t)$.
    2. Set $W_t = \sum_{i=1}^{N} w_i(t)$ and for $j = 1, \ldots, K$ set
           $p_j(t) = (1-\gamma) \sum_{i=1}^{N} \frac{w_i(t)\xi_j^i(t)}{W_t} + \frac{\gamma}{K}$ .
    3. Draw action $i_t$ randomly according to the probabilities $p_1(t), \ldots, p_K(t)$.
    4. Receive reward $x_{i_t}(t) \in [0,1]$.
    5. For $j = 1, \ldots, K$ set
           $\hat{x}_j(t) = \begin{cases} x_j(t)/p_j(t) & \text{if } j = i_t \\ 0 & \text{otherwise,} \end{cases}$
    6. For $i = 1, \ldots, N$ set
           $\hat{y}_i(t) = \boldsymbol{\xi}^i(t) \cdot \hat{\boldsymbol{x}}(t)$
           $w_i(t+1) = w_i(t) \exp\left(\gamma \hat{y}_i(t)/K\right)$ .
```

Figure 5: Pseudo-code of algorithm **Exp4** for using expert advice.

the uniform expert,

$$\tilde{G}_{\max} - \mathbf{E}[G_{\mathbf{Exp4}}] \le (e-1)\gamma\tilde{G}_{\max} + \frac{K \ln N}{\gamma} \; .$$

holds for any assignment of rewards.

Proof. We prove this theorem along the lines of the proof of Theorem 3.1. Let $q_i(t) = w_i(t)/W_t$. Then

$$\begin{aligned} \frac{W_{t+1}}{W_t} &= \sum_{i=1}^{N} \frac{w_i(t+1)}{W_t} \\ &= \sum_{i=1}^{N} q_i(t) \exp\left(\frac{\gamma}{K}\hat{y}_i(t)\right) \end{aligned}$$

$$\begin{aligned}
&\le \sum_{i=1}^{N} q_i(t)\left[1+\frac{\gamma}{K}\hat{y}_i(t)+(e-2)\left(\frac{\gamma}{K}\hat{y}_i(t)\right)^2\right] \\
&\le 1+(\gamma/K)\sum_{i=1}^{N} q_i(t)\hat{y}_i(t)+(e-2)(\gamma/K)^2\sum_{i=1}^{N} q_i(t)\hat{y}_i(t)^2 \,.
\end{aligned}$$

Taking logarithms and summing over t we get

$$\ln\frac{W_{T+1}}{W_1} \le (\gamma/K)\sum_{t=1}^{T}\sum_{i=1}^{N} q_i(t)\hat{y}_i(t)+(e-2)(\gamma/K)^2\sum_{t=1}^{T}\sum_{i=1}^{N} q_i(t)\hat{y}_i(t)^2 \,.$$

Since, for any expert k,

$$\ln\frac{W_{T+1}}{W_1} \ge \ln\frac{w_k(T+1)}{W_1} = \frac{\gamma}{K}\sum_{t=1}^{T}\hat{x}_k(t)-\ln N$$

we get

$$\sum_{t=1}^{T}\sum_{i=1}^{N} q_i(t)\hat{y}_i(t) \ge \sum_{t=1}^{T}\hat{y}_k(t)-\frac{K\ln N}{\gamma}-(e-2)\frac{\gamma}{K}\sum_{t=1}^{T}\sum_{i=1}^{N} q_i(t)\hat{y}_i(t)^2 \,.$$

Note that

$$\begin{aligned}
\sum_{i=1}^{N} q_i(t)\hat{y}_i(t) &= \sum_{i=1}^{N} q_i(t)\left(\sum_{j=1}^{K}\xi_j^i(t)\hat{x}_j(t)\right) \\
&= \sum_{j=1}^{K}\left(\sum_{i=1}^{N} q_i(t)\xi_j^i(t)\right)\hat{x}_j(t) \\
&= \sum_{j=1}^{K}\left(\frac{p_i(t)-\gamma/K}{1-\gamma}\right)\hat{x}_j(t) \le \frac{x_j(t)}{1-\gamma} \,.
\end{aligned}$$

Also

$$\begin{aligned}
\sum_{i=1}^{N} q_i(t)\hat{y}_i(t)^2 &= q_{i_t}(t)(\xi_{i_t}^i(t)\hat{x}_{i_t}(t))^2 \\
&\le \hat{x}_{i_t}(t)^2\,\frac{p_{i_t}(t)}{1-\gamma} \\
&\le \frac{\hat{x}_{i_t}(t)}{1-\gamma} \,.
\end{aligned}$$

Therefore, for all experts k,

$$G_{\mathbf{Exp4}} = \sum_{t=1}^{T}\hat{x}_{i_t}(t) \ge (1-\gamma)\sum_{t=1}^{T}\hat{y}_k(t)-\frac{K\ln N}{\gamma}-(e-2)\frac{\gamma}{K}\sum_{t=1}^{T}\sum_{j=1}^{K}\hat{x}_j(t) \,.$$

We now take expectations of both sides of this inequality. Note that

$$\mathbf{E}[\hat{y}_k(t)] = \mathbf{E}\left[\sum_{j=1}^{K} \xi_j^k(t)\hat{x}_j(t)\right] = \sum_{j=1}^{K} \xi_j^k(t)x_j(t) = y_k(t) \; .$$

Further,

$$\frac{1}{K}\mathbf{E}\left[\sum_{t=1}^{T}\sum_{j=1}^{K} \hat{x}_j(t)\right] = \sum_{t=1}^{T}\frac{1}{K}\sum_{j=1}^{K} x_j(t) \le \max_{1\le i\le N}\sum_{t=1}^{T} y_i(t) = \tilde{G}_{\max}$$

since we have assumed that the uniform expert is included in the family of experts. Combining these facts immediately implies the statement of the theorem. □

8 The regret against arbitrary strategies

In this section we present a variant of algorithm **Exp3** and prove a bound on its expected regret for any sequence $(j_1, \ldots, j_T)$ of actions. To prove this result, we rank all sequences of actions according to their "hardness". The *hardness* of a sequence $(j_1, \ldots, j_T)$ is defined by

$$\mathrm{H}(j_1, \ldots, j_T) \stackrel{\text{def}}{=} 1 + |\{1 \le \ell < T \; : \; j_\ell \ne j_{\ell+1}\}| \; .$$

So, $\mathrm{H}(1, \ldots, 1) = 1$ and $\mathrm{H}(1, 1, 3, 2, 2) = 3$. The bound on the regret which we will prove grows with the hardness of the sequence for which we are measuring the regret. In particular, we will show that the player algorithm **Exp3.S** described in Figure 6 has an expected regret of $O(\mathrm{H}(j^T)\sqrt{KT\ln(KT)})$ for any sequence $j^T = (j_1, \ldots, j_T)$ of actions. On the other hand, if the regret is measured for any sequence j^T of actions of hardness $\mathrm{H}(j^T) \le S$, then the expected regret of **Exp3.S** (with parameters tuned to this S) reduces to $O(\sqrt{SKT\ln(KT)})$. In what follows, we will use G_{j^T} to denote the return $x_{j_1}(1) + \ldots x_{j_T}(T)$ of a sequence $j^T = (j_1, \ldots, j_T)$ of actions.

Theorem 8.1 *For any $K > 0$, for any $\gamma \in (0, 1]$, and for any $\alpha > 0$,*

$$G_{j^T} - \mathbf{E}\left[G_{\mathbf{Exp3.S}}\right] \le \frac{K(\mathrm{H}(j^T)\ln(K/\alpha) + e\alpha T)}{\gamma} + (e-1)\gamma T$$

holds for any assignment of rewards, for any $T > 0$, and for any sequence $j^T = (j_1, \ldots, j_T)$ of actions.

Corollary 8.2 *Assume that algorithm* **Exp3.S** *is run with input parameters $\alpha = 1/T$ and*

$$\gamma = \min\left\{1, \sqrt{\frac{K\ln(KT)}{T}}\right\} \; .$$

Then

$$G_{j^T} - \mathbf{E}\left[G_{\mathbf{Exp3.S}}\right] \le \mathrm{H}(j^T)\sqrt{KT\ln(KT)} + 2e\sqrt{\frac{KT}{\ln(KT)}}$$

holds for any sequence $j^T = (j_1, \ldots, j_T)$ of actions.

```
Algorithm Exp3.S
Parameters: Reals $\gamma \in (0,1]$ and $\alpha > 0$.
Initialization: $w_i(1) = 1$ for $i = 1, \ldots, K$.

For each $t = 1, 2, \ldots$
    1. Set
           $p_i(t) = (1-\gamma)\frac{w_i(t)}{\sum_{j=1}^{K} w_j(t)} + \frac{\gamma}{K} \qquad i = 1, \ldots, K.$
    2. Draw $i_t$ randomly accordingly to the probabilities $p_1(t), \ldots, p_K(t)$.
    3. Receive reward $x_{i_t}(t) \in [0,1]$.
    4. For $j = 1, \ldots, K$ set
           $\hat{x}_j(t) = \begin{cases} x_j(t)/p_j(t) & \text{if } j = i_t \\ 0 & \text{otherwise,} \end{cases}$
           $w_j(t+1) = w_j(t)\, \exp\left(\gamma \hat{x}_j(t)/K\right) + \frac{e\alpha}{K} \sum_{i=1}^{K} w_i(t)\ .$
```

Figure 6: Pseudo-code of algorithm **Exp3**.**S** to control the expected regret.

Note that the statement of Corollary 8.2 can be equivalently written as

$$\begin{aligned} \mathbf{E}\left[G_{\mathbf{Exp3.S}}\right] \;\geq\; & \max_{j^T} \left(G_{j^T} - \mathrm{H}(j^T)\sqrt{KT\ln(KT)} \right) \\ & - 2e\sqrt{\frac{KT}{\ln(KT)}} \end{aligned}$$

revealing that algorithm **Exp3**.**S** is able to automatically trade-off between the return G_{j^T} of a sequence j^T and its hardness $\mathrm{H}(j^T)$.

Corollary 8.3 *Assume that algorithm* **Exp3**.**S** *is run with input parameters* $\alpha = 1/T$ *and*

$$\gamma = \min\left\{1, \sqrt{\frac{K(S\ln(KT) + e)}{(e-1)T}}\right\}\ .$$

Then

$$G_{j^T} - \mathbf{E}\left[G_{\mathbf{Exp3.S}}\right] \leq 2\sqrt{e-1}\sqrt{KT\left(S\ln(KT) + e\right)}$$

holds for any sequence $j^T = (j_1, \ldots, j_T)$ *of actions such that* $\mathrm{H}(j^T) \leq S$.

Proof of Theorem 8.1. Fix any sequence $j^T = (j_1, \ldots, j_T)$ of actions. With a technique that follows closely the proof of Theorem 3.1, we can prove that for all sequences $i_1, \ldots, i_T$ of actions drawn by $\mathbf{Exp3.S}$,

$$\frac{W_{t+1}}{W_t} \le 1 + \frac{\gamma/K}{1-\gamma} x_{i_t}(t) + \frac{(e-2)(\gamma/K)^2}{1-\gamma} \sum_{i=1}^{K} \hat{x}_i(t) + e\alpha \; . \tag{25}$$

where, as usual, $W_t = w_1(t) + \ldots + w_K(t)$. Now let $S = \mathrm{H}(j^T)$ and partition $(1, \ldots, T)$ in *segments*

$$[T_1, \ldots, T_2), [T_2, \ldots, T_3), \ldots, [T_S, \ldots, T_{S+1})$$

where $T_1 = 1$, $T_{S+1} = T + 1$, and $j_{T_s} = j_{T_s+1} = \ldots = j_{T_{s+1}-1}$ for each segment $s = 1, \ldots, S$. Fix an arbitrary segment $[T_s, T_{s+1})$ and let $\Delta_s = T_{s+1} - T_s$. Furthermore, let

$$G_{\mathbf{Exp3.S}}(s) \stackrel{\text{def}}{=} \sum_{t=T_s}^{T_{s+1}-1} x_{i_t}(t) \; .$$

Taking logarithms on both sides of (25) and summing over $t = T_s, \ldots, T_{s+1} - 1$ we get

$$\ln \frac{W_{T_{s+1}}}{W_{T_s}} \le \frac{\gamma/K}{1-\gamma} G_{\mathbf{Exp3.S}}(s) + \frac{(e-2)(\gamma/K)^2}{1-\gamma} \sum_{t=T_s}^{T_{s+1}-1} \sum_{i=1}^{K} \hat{x}_i(t) + e\alpha\Delta_s \; . \tag{26}$$

Now let j be the action such that $j_{T_s} = \ldots = j_{T_{s+1}-1} = j$. Since

$$\begin{aligned} w_j(T_{s+1}) &\ge w_j(T_s + 1) \exp\left(\frac{\gamma}{K} \sum_{t=T_s+1}^{T_{s+1}-1} \hat{x}_j(t) \right) \\ &\ge \frac{e\alpha}{K} W_{T_s} \exp\left(\frac{\gamma}{K} \sum_{t=T_s+1}^{T_{s+1}-1} \hat{x}_j(t) \right) \\ &\ge \frac{\alpha}{K} W_{T_s} \exp\left(\frac{\gamma}{K} \sum_{t=T_s}^{T_{s+1}-1} \hat{x}_j(t) \right) \end{aligned}$$

where the last step uses $\gamma \hat{x}_j(t)/K \le 1$, we have

$$\ln \frac{W_{T_{s+1}}}{W_{T_s}} \ge \ln \frac{w_j(T_{s+1})}{W_{T_s}} \ge \ln\left(\frac{\alpha}{K}\right) + \frac{\gamma}{K} \sum_{t=T_s}^{T_{s+1}-1} \hat{x}_j(t) \; . \tag{27}$$

Piecing together (26) and (27) we get

$$G_{\mathbf{Exp3.S}}(s) \ge (1-\gamma) \sum_{t=T_s}^{T_{s+1}-1} \hat{x}_j(t) - \frac{K \ln(K/\alpha)}{\gamma} - (e-2)\frac{\gamma}{K} \sum_{t=T_s}^{T_{s+1}-1} \sum_{i=1}^{K} \hat{x}_i(t) - \frac{e\alpha K \Delta_s}{\gamma} \; .$$

Summing over all segments $s = 1, \ldots, S$, taking expectation with respect to the random choices of algorithm $\mathbf{Exp3.S}$, and using

$$G_{(j_1,\ldots,j_T)} \leq T \qquad \text{and} \qquad \sum_{t=1}^{T} \sum_{i=1}^{K} x_i(t) \leq KT$$

yields the inequality in the statement of the theorem. □

If the time horizon T is not known, we can apply techniques similar to those applied for proving Theorem 6.4 in Section 6. More specifically, we introduce a new algorithm, $\mathbf{Exp3.S.1}$, that runs $\mathbf{Exp3.S}$ as a subroutine. Suppose that at each new run (or epoch) $r = 0, 1, \ldots$, $\mathbf{Exp3.S}$ is started with its parameters set as prescribed in Corollary 8.2, where T is set to $T_r = 2^r$, and then stopped after T_r iterations. Clearly, for any fixed sequence $j^T = (j_1, \ldots, j_T)$ of actions, the number of segments (see proof of Theorem 8.1 for a definition of segment) within each epoch r is at most $\mathrm{H}(j^T)$. Hence the expected regret of $\mathbf{Exp3.S.1}$ for epoch r is certainly not more than

$$(\mathrm{H}(j^T) + 2e)\sqrt{KT_r \ln(KT_r)} \,.$$

Let ℓ be such that $2^\ell \leq T < 2^{\ell+1}$. Then the last epoch is $\ell \leq \log_2 T$ and the overall regret (over the $\ell + 1$ epochs) is at most

$$(\mathrm{H}(j^T) + 2e) \sum_{r=0}^{\ell} \sqrt{KT_r \ln(KT_r)} \leq (\mathrm{H}(j^T) + 2e)\sqrt{KT_\ell \ln(KT_\ell)} \sum_{r=0}^{\ell} \sqrt{T_r} \,.$$

Finishing up the calculations proves the following.

Corollary 8.4

$$G_{j^T} - \mathbf{E}\left[G_{\mathbf{Exp3.S.1}}\right] \leq \frac{\mathrm{H}(j^T) + 2e}{\sqrt{2} - 1}\sqrt{2KT \ln(KT)}$$

for any $T > 0$ and for any sequence $j^T = (j_1, \ldots, j_T)$ of actions.

On the other hand, if $\mathbf{Exp3.S.1}$ runs $\mathbf{Exp3.S}$ with parameters set as prescribed in Corollary 8.3, with a reasoning similar to the one above we conclude the following.

Corollary 8.5

$$G_{j^T} - \mathbf{E}\left[G_{\mathbf{Exp3.S.1}}\right] \leq \frac{2\sqrt{e-1}}{\sqrt{2} - 1}\sqrt{2KT(S \ln(KT) + e)}$$

for any $T > 0$ and for any sequence $j^T = (j_1, \ldots, j_T)$ of actions such that $\mathrm{H}(j^T) \leq S$.

9 Applications to game theory

The adversarial bandit problem can be easily related to the problem of playing repeated games. For $N > 1$ integer, a N-person finite game is defined by N finite sets $S_1, \ldots, S_N$ of pure strategies, one set for each player, and by N functions $u_1, \ldots, u_N$, where function $u_i : S_1 \times \ldots \times S_N \to \mathbb{R}$ is player's i payoff function. Note the each player's payoff depends both on the pure strategy chosen by the player and on the pure strategies chosen by the other players. Let $S = S_1 \times \ldots \times S_N$ and let $S_{-i} = S_1 \times \ldots \times S_{i-1} \times S_{i+1} \times \ldots \times S_N$. We use $\boldsymbol{s}$ and $\boldsymbol{s}_{-i}$ to denote typical members of, respectively, S and S_{-i}. Given $\boldsymbol{s} \in S$, we will often write $(j, \boldsymbol{s}_{-i})$ to denote $(s_1, \ldots, s_{i-1}, j, s_{i+1}, \ldots, s_N)$, where $j \in S_i$. Suppose that the game is played repeatedly through time. Assume for now that each player knows all payoff functions and, after each repetition (or round) t, also knows the vector $\boldsymbol{s}(t) = (s_1(t), \ldots, s_N(t))$ of pure strategies chosen by the players. Hence, the pure strategy $s_i(t)$, chosen by player i at round t may depend on what player i and the other players chose in the past rounds. The *average regret* of player i for the pure strategy j after T rounds is defined by

$$R_i^{(j)}(T) = \frac{1}{T} \sum_{t=1}^{T} \left[u_i(j, \boldsymbol{s}_{-i}(t)) - u_i(\boldsymbol{s}(t)) \right] .$$

This is how much player i lost on average for not playing the pure strategy j on all rounds, given that all the other players kept their choices fixed.

A desirable property for a player is Hannan-consistency [8], defined as follows. Player i is *Hannan-consistent* if

$$\limsup_{T \to \infty} \max_{j \in S_i} R_i^{(j)}(T) = 0 \qquad \text{with probability 1.}$$

The existence and properties of Hannan-consistent players have been first investigated by Hannan [10] and Blackwell [2], and later by many others (see [18] for a nice survey).

Hannan-consistency can be also studied in the so-called “unknown game setup”, where it is further assumed that: (1) each player knows neither the total number of players nor the payoff function of any player (including itself), (2) after each round each player sees its own payoffs but it sees neither the choices of the other players nor the resulting payoffs. This setup was previously studied by Baños [1], Megiddo [16], and by Hart and Mas-Colell [11, 12].

We can apply the results of Section 6 to prove that a player using algorithm $\mathbf{Exp3.P.1}$ as mixed strategy is Hannan-consistent in the unknown game setup whenever the payoffs obtained by the player belong to a known bounded real interval. To do that, we must first extend our results to the case when the assignment of rewards can be chosen adaptively. More precisely, we can view the payoff $x_{i_t}(t)$, received by the gambler at trial t of the bandit problem, as the payoff $u_i(i_t, \boldsymbol{s}_{-i}(t))$ received by player i at the t-th round of the game. However, unlike our adversarial bandit framework where all the rewards were assigned to each arm at the beginning, here the payoff $u_i(i_t, \boldsymbol{s}_{-i}(t))$ depends on the (possibly randomized) choices of all players which, in turn, are functions of their realized payoffs. In our bandit terminology, this corresponds to assuming that the vector $(x_1(t), \ldots, x_K(t))$ of rewards for each trial t is chosen by an adversary who knows the gambler's strategy and the outcome of the gambler's random draws up to time $t-1$. We leave to the interested reader the easy but lengthy task of checking that *all* of our results (including those of Section 6) hold under this additional assumption.

Using Theorem 6.4 we then get the following.

Theorem 9.1 *If player i has $K \geq 2$ pure strategies and plays in the unknown game setup (with payoffs in $[0,1]$) using the mixed strategy* $\mathbf{Exp3.P.1}$*, then*

$$\max_{j \in S_i} R_i^{(j)}(T) \leq \frac{10}{\sqrt{2}-1} \sqrt{\frac{2K}{T}\left(\ln \frac{KT}{\delta} + c_T\right)} + \frac{10(1+\log_2 T)}{T}\left(\ln \frac{KT}{\delta} + c_T\right) ,$$

where $c_T = 2\ln(2 + \log_2 T)$, holds with probability at least $1 - \delta$, for all $0 < \delta < 1$ and for all $T = (K/\delta)^{\Omega(1/K)}$.

Note that, according to Theorem 5.1, the rate of convergence is optimal both in T and K up to logarithmic factors.

Theorem 9.1, along with Corollary 6.5, immediately implies the result below.

Corollary 9.2 *Player's strategy* $\mathbf{Exp3.P.1}$ *is Hannan-consistent in the unknown game setup.*

As pointed out in [18], Hannan-consistency has an interesting consequence for repeated zero-sum games. These games are defined by an $n \times m$ matrix $\mathbf{M}$. On each round t, the *row player* chooses a row i of the matrix. At the same time, the *column player* chooses a column j. The row player then gains the quantity $\mathbf{M}_{ij}$, while the column player loses the same quantity. In repeated play, the row player's goal is to maximize its expected total gain over a sequence of plays, wile the column player's goal is to minimize its expected total loss.

Suppose in some round the row player chooses its next move i randomly according to a probability distribution on rows represented by a (column) vector $\boldsymbol{p} \in [0,1]^n$, and the column player similarly chooses according to a probability vector $\boldsymbol{q} \in [0,1]^m$. Then the expected payoff is $\boldsymbol{p}^T \mathbf{M} \boldsymbol{q}$. Von Neumann's minimax theorem states that

$$\max_{\boldsymbol{p}} \min_{\boldsymbol{q}} \boldsymbol{p}^T \mathbf{M} \boldsymbol{q} = \min_{\boldsymbol{q}} \max_{\boldsymbol{p}} \boldsymbol{p}^T \mathbf{M} \boldsymbol{q} ,$$

where maximum and minimum are taken over the (compact) set of all distribution vectors $\boldsymbol{p}$ and $\boldsymbol{q}$. The quantity v defined by the above equation is called the *value* of the zero-sum game with matrix $\mathbf{M}$. In words, this says that there exists a mixed (randomized) strategy $\overline{\mathbf{p}}$ for the row player that guarantees expected payoff at least v, regardless of the column player's action. Moreover, this payoff is optimal in the sense that the column player can choose a mixed strategy whose expected payoff is at most v, regardless of the row player's action. Thus, if the row player knows the matrix $\mathbf{M}$, it can compute a strategy (for instance, using linear programming) that is certain to bring an expected optimal payoff not smaller than v on each round.

Suppose now that the game $\mathbf{M}$ is entirely unknown to the row player. To be precise, assume the row player knows only the number of rows of the matrix and a bound on the magnitude of the entries of $\mathbf{M}$. Then, using the results of Section 4, we can show that the row player can play in such a manner that its payoff per round will rapidly converge to the optimal maximin payoff v.

Theorem 9.3 *Let $\mathbf{M}$ be an unknown game matrix in $[a,b]^{n \times m}$ with value v. Suppose the row player, knowing only a, b and n, uses the mixed strategy* $\mathbf{Exp3.1}$*. Then the row player's expected payoff per round is at least*

$$v - 8\sqrt{e-1}\sqrt{\frac{n \ln n}{T}} - 8(e-1)\frac{n}{T} - 2\frac{n \ln n}{T} .$$

Proof. We assume that $[a,b]=[0,1]$; the extension to the general case is straightforward. By Theorem 4.1, we have

$$\begin{aligned}\mathbf{E}\left[\sum_{t=1}^{T}\mathbf{M}_{i_t j_t}\right] &= \mathbf{E}\left[\sum_{t=1}^{T}x_{i_t}(t)\right]\\ &\geq \max_i \mathbf{E}\left[\sum_{t=1}^{T}x_i(t)\right]-8\sqrt{e-1}\sqrt{Tn\ln n}-8(e-1)n-2n\ln n\ .\end{aligned}$$

Let $\overline{\mathbf{p}}$ be a maxmin strategy for the row player such that

$$v=\max_{\boldsymbol{p}}\min_{\boldsymbol{q}}\boldsymbol{p}^T\mathbf{M}\boldsymbol{q}=\min_{\boldsymbol{q}}\overline{\mathbf{p}}^T\mathbf{M}\boldsymbol{q}\ ,$$

and let $\boldsymbol{q}(t)$ be a distribution vector whose j_t-th component is 1. Then

$$\max_i \mathbf{E}\left[\sum_{t=1}^{T}x_i(t)\right]\geq\sum_{i=1}^{n}\overline{p}_i\mathbf{E}\left[\sum_{t=1}^{T}x_i(t)\right]=\mathbf{E}\left[\sum_{t=1}^{T}\overline{\mathbf{p}}\cdot\boldsymbol{x}(t)\right]=\mathbf{E}\left[\sum_{t=1}^{T}\overline{\mathbf{p}}^T\mathbf{M}\boldsymbol{q}(t)\right]\geq vT$$

since $\overline{\mathbf{p}}^T\mathbf{M}\boldsymbol{q}\geq v$ for all $\boldsymbol{q}$.

Thus, the row player's expected payoff is at least

$$vT-8\sqrt{e-1}\sqrt{Tn\ln n}-8(e-1)n-2n\ln n\ .$$

Dividing by T to get the average per-round payoff gives the result. □

Note that the theorem is independent of the number of columns of $\mathbf{M}$ and, with appropriate assumptions, the theorem can be easily generalized to column players with an infinite number of strategies. If the matrix $\mathbf{M}$ is very large and all entries are small, then, even if $\mathbf{M}$ is known to the player, our algorithm may be an efficient alternative to linear programming.

Acknowledgments

We express special thanks to Kurt Hornik for his advice and his patience when listening to our ideas and proofs for an earlier draft of this paper. We thank Yuval Peres and Amir Dembo for their help regarding the analysis of martingales. Peter Auer and Nicolò Cesa-Bianchi gratefully acknowledge support of ESPRIT Working Group EP 27150, Neural and Computational Learning II (NeuroCOLT II)

References

[1] Alfredo Baños. On pseudo-games. *The Annals of Mathematical Statistics*, 39(6):1932–1945, 1968.

[2] David Blackwell. Controlled random walks. invited address, Institute of Mathematical Statistics Meeting, Seattle, Washington, 1956.

[3] Nicolò Cesa-Bianchi, Yoav Freund, David Haussler, David P. Helmbold, Robert E. Schapire, and Manfred K. Warmuth. How to use expert advice. *Journal of the Association for Computing Machinery*, 44(3):427–485, May 1997.

[4] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, 1991.

[5] Dean P. Foster and Rakesh V. Vohra. A randomization rule for selecting forecasts. *Operations Research*, 41(4):704–709, July–August 1993.

[6] Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, August 1997.

[7] Yoav Freund and Robert E. Schapire. Adaptive game playing using multiplicative weights. *Games and Economic Behavior*, 29:79–103, 1999.

[8] Drew Fudenberg and David K. Levine. Consistency and cautious fictitious play. *Journal of Economic Dynamics and Control*, 19:1065–1089, 1995.

[9] J. C. Gittins. *Multi-armed Bandit Allocation Indices*. John Wiley & Sons, 1989.

[10] James Hannan. Approximation to Bayes risk in repeated play. In M. Dresher, A. W. Tucker, and P. Wolfe, editors, *Contributions to the Theory of Games*, volume III, pages 97–139. Princeton University Press, 1957.

[11] Sergiu Hart and Andreu Mas-Colell. A simple procedure leading to correlated equilibrium. *Econometrica*, 68:1127–1150, 2000.

[12] Sergiu Hart and Andreu Mas-Colell. A general class of adaptive strategies. *Journal of Economic Theory*, 98(1)26–54, 2001.

[13] T. Ishikida and P. Varaiya. Multi-armed bandit problem revisited. *Journal of Optimization Theory and Applications*, 83(1):113–154, October 1994.

[14] T. L. Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6:4–22, 1985.

[15] Nick Littlestone and Manfred K. Warmuth. The weighted majority algorithm. *Information and Computation*, 108:212–261, 1994.

[16] N. Megiddo. On repeated games with incomplete information played by non-Bayesian players. *International Journal of Game Theory*, 9(3):157–167, 1980.

[17] J. Neveu. *Discrete-Parameter Martingales*. North Holland, 1975.

[18] Dean P. Foster and Rakesh Vohra. Regret in the on-line decision problem. *Games and Economic Behavior*, 29:7–36, 1999.

[19] H. Robbins. Some aspects of the sequential design of experiments. *Bulletin American Mathematical Society*, 55:527–535, 1952.

[20] Volodimir G. Vovk. Aggregating strategies. In *Proceedings of the Third Annual Workshop on Computational Learning Theory*, pages 371–383, 1990.

A Proof of Theorem 5.1

We construct the random distribution of rewards as follows. First, before play begins, one action I is chosen uniformly at random to be the "good" action. The T rewards $x_I(t)$ associated with the good action are chosen independently at random to be 1 with probability $1/2 + \epsilon$ and 0 otherwise for some small, fixed constant $\epsilon \in (0, 1/2]$ to be chosen later in the proof. The rewards $x_j(t)$ associated with the other actions $j \neq I$ are chosen independently at random to be 0 or 1 with equal odds. Then the expected reward of the best action is at least $(1/2 + \epsilon)T$. The main part of the proof below is a derivation of an upper bound on the expected gain of any algorithm for this distribution of rewards.

We write $\mathbf{P}_*\{\cdot\}$ to denote probability with respect to this random choice of rewards, and we also write $\mathbf{P}_i\{\cdot\}$ to denote probability conditioned on i being the good action: $\mathbf{P}_i\{\cdot\} = \mathbf{P}_*\{\cdot \mid I = i\}$. Finally, we write $\mathbf{P}_{unif}\{\cdot\}$ to denote probability with respect to a uniformly random choice of rewards for *all* actions (including the good action). Analogous expectation notation $\mathbf{E}_*[\cdot]$, $\mathbf{E}_i[\cdot]$ and $\mathbf{E}_{unif}[\cdot]$ will also be used.

Let A be the player strategy. Let $r_t = x_{i_t}(t)$ be a random variable denoting the reward received at time t, and let $\mathbf{r}^t$ denote the sequence of rewards received up through trial t: $\mathbf{r}^t = \langle r_1, \ldots, r_t \rangle$. For shorthand, $\mathbf{r} = \mathbf{r}^T$ is the entire sequence of rewards.

Any randomized playing strategy is equivalent to an a-priori random choice from the set of all deterministic strategies. Thus, because the adversary strategy we have defined is oblivious to the actions of the player, it suffices to prove an upper bound on the expected gain of any *deterministic* strategy (this is not crucial for the proof but simplifies the notation). Therefore, we can formally regard the algorithm A as a fixed function which, at each step t, maps the reward history $\mathbf{r}^{t-1}$ to its next action i_t.

As usual, $G_A = \sum_{t=1}^T r_t$ denotes the return of the algorithm, and $G_{\max} = \max_j \sum_{t=1}^T x_j(t)$ is the return of the best action.

Let N_i be a random variable denoting the number of times action i is chosen by A. Our first lemma bounds the difference between expectations when measured using $\mathbf{E}_i[\cdot]$ or $\mathbf{E}_{unif}[\cdot]$.

Lemma A.1 *Let $f : \{0,1\}^T \to [0, M]$ be any function defined on reward sequences $\mathbf{r}$. Then for any action i,*

$$\mathbf{E}_i[f(\mathbf{r})] \leq \mathbf{E}_{unif}[f(\mathbf{r})] + \frac{M}{2}\sqrt{-\mathbf{E}_{unif}[N_i]\ln(1-4\epsilon^2)}.$$

Proof. We apply standard methods that can be found, for instance, in Cover and Thomas [4]. For any distributions $\mathbf{P}$ and $\mathbf{Q}$, let

$$\|\mathbf{P} - \mathbf{Q}\|_1 \doteq \sum_{\mathbf{r} \in \{0,1\}^T} |\mathbf{P}\{\mathbf{r}\} - \mathbf{Q}\{\mathbf{r}\}|$$

be the variational distance, and let

$$\mathrm{KL}(\mathbf{P} \parallel \mathbf{Q}) \doteq \sum_{\mathbf{r} \in \{0,1\}^T} \mathbf{P}\{\mathbf{r}\} \lg\left(\frac{\mathbf{P}\{\mathbf{r}\}}{\mathbf{Q}\{\mathbf{r}\}}\right)$$

be the Kullback-Liebler divergence or relative entropy between the two distributions. (We use lg to denote $\log_2$.) We also use the notation

$$\mathrm{KL}\left(\mathbf{P}\{r_t \mid \mathbf{r}^{t-1}\} \;\|\; \mathbf{Q}\{r_t \mid \mathbf{r}^{t-1}\}\right) \doteq \sum_{\mathbf{r}^t \in \{0,1\}^t} \mathbf{P}\{\mathbf{r}^t\} \lg\left(\frac{\mathbf{P}\{r_t \mid \mathbf{r}^{t-1}\}}{\mathbf{Q}\{r_t \mid \mathbf{r}^{t-1}\}}\right)$$

for the conditional relative entropy of r_t given $\mathbf{r}^{t-1}$. Finally, for $p, q \in [0,1]$, we use

$$\mathrm{KL}\left(p \;\|\; q\right) \doteq p \lg\left(\frac{p}{q}\right) + (1-p)\lg\left(\frac{1-p}{1-q}\right)$$

as shorthand for the relative entropy between two Bernoulli random variables with parameters p and q.

We have that

$$\begin{aligned}
\mathbf{E}_i\left[f(\mathbf{r})\right] - \mathbf{E}_{unif}\left[f(\mathbf{r})\right] &= \sum_{\mathbf{r}} f(\mathbf{r})(\mathbf{P}_i\{\mathbf{r}\} - \mathbf{P}_{unif}\{\mathbf{r}\}) \\
&\le \sum_{\mathbf{r}:\mathbf{P}_i\{\mathbf{r}\} \ge \mathbf{P}_{unif}\{\mathbf{r}\}} f(\mathbf{r})(\mathbf{P}_i\{\mathbf{r}\} - \mathbf{P}_{unif}\{\mathbf{r}\}) \\
&\le M \sum_{\mathbf{r}:\mathbf{P}_i\{\mathbf{r}\} \ge \mathbf{P}_{unif}\{\mathbf{r}\}} (\mathbf{P}_i\{\mathbf{r}\} - \mathbf{P}_{unif}\{\mathbf{r}\}) \\
&= \frac{M}{2}\|\mathbf{P}_i - \mathbf{P}_{unif}\|_1. \qquad (28)
\end{aligned}$$

Also, Cover and Thomas's Lemma 12.6.1 states that

$$\|\mathbf{P}_{unif} - \mathbf{P}_i\|_1^2 \le (2\ln 2)\mathrm{KL}\left(\mathbf{P}_{unif} \;\|\; \mathbf{P}_i\right). \qquad (29)$$

The "chain rule for relative entropy" (Cover and Thomas's Theorem 2.5.3) gives that

$$\begin{aligned}
\mathrm{KL}\left(\mathbf{P}_{unif} \;\|\; \mathbf{P}_i\right) &= \sum_{t=1}^{T} \mathrm{KL}\left(\mathbf{P}_{unif}\{r_t \mid \mathbf{r}^{t-1}\} \;\|\; \mathbf{P}_i\{r_t \mid \mathbf{r}^{t-1}\}\right) \\
&= \sum_{t=1}^{T} \left(\mathbf{P}_{unif}\{i_t \ne i\}\ \mathrm{KL}\left(\tfrac{1}{2} \;\|\; \tfrac{1}{2}\right) + \mathbf{P}_{unif}\{i_t = i\}\ \mathrm{KL}\left(\tfrac{1}{2} \;\|\; \tfrac{1}{2}+\epsilon\right)\right) \\
&= \sum_{t=1}^{T} \mathbf{P}_{unif}\{i_t = i\}\left(-\tfrac{1}{2}\lg(1-4\epsilon^2)\right) \\
&= \mathbf{E}_{unif}\left[N_i\right]\left(-\tfrac{1}{2}\lg(1-4\epsilon^2)\right). \qquad (30)
\end{aligned}$$

The second equality can be seen as follows: Regardless of the past history of rewards $\mathbf{r}^{t-1}$, the conditional probability distribution $\mathbf{P}_{unif}\{r_t \mid \mathbf{r}^{t-1}\}$ on the next reward r_t is uniform on $\{0,1\}$. The conditional distribution $\mathbf{P}_i\{r_t \mid \mathbf{r}^{t-1}\}$ is also easily computed: Given $\mathbf{r}^{t-1}$, the next action i_t is fixed by A. If this action is not the good action i, then the conditional distribution is uniform on $\{0,1\}$; otherwise, if $i_t = i$, then r_t is 1 with probability $1/2+\epsilon$ and 0 otherwise.

The lemma now follows by combining (28), (29) and (30). □

We are now ready to prove the theorem. Specifically, we show the following:

Theorem A.2 *For any player strategy A, and for the distribution on rewards described above, the expected regret of algorithm A is lower bounded by:*

$$\mathbf{E}_*\left[G_{\max} - G_A\right] \geq \epsilon \left(T - \frac{T}{K} - \frac{T}{2}\sqrt{-\frac{T}{K}\ln(1-4\epsilon^2)} \right).$$

Proof. If action i is chosen to be the good action, then clearly the expected payoff at time t is $1/2 + \epsilon$ if $i_t = i$ and $1/2$ if $i_t \neq i$:

$$\begin{aligned} \mathbf{E}_i\left[r_t\right] &= \left(\tfrac{1}{2} + \epsilon\right)\mathbf{P}_i\{i_t = i\} + \tfrac{1}{2}\mathbf{P}_i\{i_t \neq i\} \\ &= \tfrac{1}{2} + \epsilon\,\mathbf{P}_i\{i_t = i\}. \end{aligned}$$

Thus, the expected gain of algorithm A is

$$\mathbf{E}_i\left[G_A\right] = \sum_{t=1}^{T} \mathbf{E}_i\left[r_t\right] = \frac{T}{2} + \epsilon\,\mathbf{E}_i\left[N_i\right]. \tag{31}$$

Next, we apply Lemma A.1 to N_i, which is a function of the reward sequence $\mathbf{r}$ since the actions of player strategy A are determined by the past rewards. Clearly, $N_i \in [0, T]$. Thus,

$$\mathbf{E}_i\left[N_i\right] \leq \mathbf{E}_{\mathit{unif}}\left[N_i\right] + \frac{T}{2}\sqrt{-\mathbf{E}_{\mathit{unif}}\left[N_i\right]\ln(1-4\epsilon^2)}$$

and so

$$\begin{aligned} \sum_{i=1}^{K} \mathbf{E}_i\left[N_i\right] &\leq \sum_{i=1}^{K}\left(\mathbf{E}_{\mathit{unif}}\left[N_i\right] + \frac{T}{2}\sqrt{-\mathbf{E}_{\mathit{unif}}\left[N_i\right]\ln(1-4\epsilon^2)}\right) \\ &\leq T + \frac{T}{2}\sqrt{-TK\ln(1-4\epsilon^2)} \end{aligned}$$

using the fact that $\sum_{i=1}^{K}\mathbf{E}_{\mathit{unif}}\left[N_i\right] = T$, which implies that $\sum_{i=1}^{K}\sqrt{\mathbf{E}_{\mathit{unif}}\left[N_i\right]} \leq \sqrt{TK}$. Therefore, combining with (31),

$$\mathbf{E}_*\left[G_A\right] = \frac{1}{K}\sum_{i=1}^{K}\mathbf{E}_i\left[G_A\right] \leq \frac{T}{2} + \epsilon\left(\frac{T}{K} + \frac{T}{2}\sqrt{-\frac{T}{K}\ln(1-4\epsilon^2)}\right).$$

The expected gain of the best action is at least the expected gain of the good action, so $\mathbf{E}_*\left[G_{\max}\right] \geq T(1/2 + \epsilon)$. Thus, we get that the regret is lower bounded by the bound given in the statement of the theorem. □

For small ϵ, the bound given in Theorem A.2 is of the order

$$\Theta\left(T\epsilon - T\epsilon^2\sqrt{\frac{T}{K}}\right).$$

Choosing $\epsilon = c\sqrt{K/T}$ for some small constant c, gives a lower bound of $\Omega(\sqrt{KT})$. Specifically, the lower bound given in Theorem 5.1 is obtained from Theorem A.2 by choosing $\epsilon = (1/4)\min\{\sqrt{K/T}, 1\}$ and using the inequality $-\ln(1-x) \leq (4\ln(4/3))x$ for $x \in [0, 1/4]$.