

Tight Lower Bounds for Multiplicative Weights Algorithmic Families*

Nick Gravin¹, Yuval Peres², and Balasubramanian Sivan³

- 1 Massachusetts Institute of Technology, Cambridge, MA, USA
ngravin@mit.edu
- 2 Microsoft Research, Redmond, WA, USA
peres@microsoft.com
- 3 Google Research, New York, NY, USA
balusivan@google.com

Abstract

We study the fundamental problem of prediction with expert advice and develop regret lower bounds for a large family of algorithms for this problem. We develop simple adversarial primitives, that lend themselves to various combinations leading to sharp lower bounds for many algorithmic families. We use these primitives to show that the classic Multiplicative Weights Algorithm (MWA) has a regret of $\sqrt{\frac{T \ln k}{2}}$ (where T is the time horizon and k is the number of experts), there by completely closing the gap between upper and lower bounds. We further show a regret lower bound of $\frac{2}{3} \sqrt{\frac{T \ln k}{2}}$ for a much more general family of algorithms than MWA, where the learning rate can be arbitrarily varied over time, or even picked from arbitrary distributions over time. We also use our primitives to construct adversaries in the geometric horizon setting for MWA to precisely characterize the regret at $\frac{0.391}{\sqrt{\delta}}$ for the case of 2 experts and a lower bound of $\frac{1}{2} \sqrt{\frac{\ln k}{2\delta}}$ for the case of arbitrary number of experts k (here δ is the probability that the game ends in any given round).

1998 ACM Subject Classification F.2.0 [Analysis of Algorithms and Problem Complexity] General

Keywords and phrases Multiplicative Weights, Lower Bounds, Adversarial Primitives

Digital Object Identifier 10.4230/LIPIcs.ICALP.2017.48

1 Introduction

In this paper we develop tight lower bounds on the regret obtainable by a broad family of algorithms for the fundamental problem of prediction with expert advice. Predicting future events based on past observations, a.k.a. prediction with expert advice, is a classic problem in learning. The experts framework was the first framework proposed for online learning and encompasses several applications as special cases. The underlying problem is an online optimization problem: a *player* has to make a decision at each time step, namely, decide which of the k experts' advice to follow. At every time t , an *adversary* sets gains for each expert: a gain of $g_{i,t}$ for expert i at time t . Simultaneously, the *player*, seeing the gains from all previous steps except t , has to choose an action, i.e., decide on which expert to follow. If the player follows expert $j(t)$ at time t , he gains $g_{j(t),t}$. At the end of each step t , the gains

* A full version of the paper is available at <http://arxiv.org/abs/1607.02834>.



© Nick Gravin, Yuval Peres, and Balasubramanian Sivan;
licensed under Creative Commons License CC-BY

44th International Colloquium on Automata, Languages, and Programming (ICALP 2017).

Editors: Ioannis Chatzigiannakis, Piotr Indyk, Fabian Kuhn, and Anca Muscholl;

Article No. 48; pp. 48:1–48:14



Leibniz International Proceedings in Informatics

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany



associated with all experts are revealed to the player, and the player's choice is revealed to the adversary. In the *finite horizon model*, this process is repeated for T steps, and the player's goal is to perform (achieve a cumulative gain) as close as possible to the best single action (best expert) in hindsight, i.e., to minimize his *regret* $R_{T,k}$:

$$R_{T,k} = \max_{1 \leq i \leq k} \sum_{t=1}^T g_{i,t} - \sum_{t=1}^T g_{j(t),t}.$$

Apart from assuming that the $g_{i,t}$'s are bounded in $[0, 1]$, we don't assume anything else about the gains¹. Just as natural as the finite horizon model is the model with a *geometric horizon*: the stopping time is a geometric random variable with expectation $\frac{1}{\delta}$. In other words, the process ends at any given step with probability δ , independently of the past. Equivalently, both the player and the adversary discount the future with a $1 - \delta$ factor. In this paper, we study both the finite horizon model and the geometric horizon model. We begin with the discussion for finite horizon model below.

Main contribution

In this paper we develop simple adversarial primitives and demonstrate that, when applied in various combinations, they result in remarkably sharp lower bounds for a broad family of algorithms. We first describe the family of algorithms we study, and then discuss our main results.

Multiplicative Weights Algorithm

We begin with the Multiplicative Weights Algorithm, which is a simple, powerful and widely used algorithm for a variety of learning problems. In the experts problem, at each time t , MWA computes the cumulative gain $G_{i,t-1} = \sum_{s=1}^{t-1} g_{i,s}$ of each expert i accumulated over the past $t - 1$ steps, and will follow expert i 's advice with probability proportional to $e^{\eta G_{i,t-1}}$. Namely, with probability $\frac{e^{\eta G_{i,t-1}}}{\sum_{j=1}^k e^{\eta G_{j,t-1}}}$ where η is a parameter that can be tuned.

The per-step computation of the algorithm is extremely simple and straightforward. The intuition behind the algorithm is to increase the weight of any expert that performs well by a multiplicative factor. Despite the simplicity and the heuristic origins of the algorithm, it is surprisingly powerful: the pioneering work of Cesa Bianchi et al. [6] showed that MWA obtains a sublinear regret of $\sqrt{\frac{T \ln k}{2}}$, and that this is asymptotically optimal as the number of experts k and the number of time steps T both tend to ∞ .

Families of algorithms

The MWA is a single-parameter family of algorithms, i.e., the learning rate parameter η is the only parameter available for the player. In general one could think of η being an arbitrary function of time t , i.e., at step t , algorithm follows expert i with probability $\frac{e^{\eta(t) G_{i,t-1}}}{\sum_{j=1}^k e^{\eta(t) G_{j,t-1}}}$.

Note that this is a T -parameter family of algorithms and is quite general. The most general family of algorithms we study is when at each time t , the quantity $\eta(t)$ is drawn from an

¹ As one might expect, it turns out that restricting the adversary to set gains in $\{0, 1\}$ instead of $[0, 1]$ is without loss of generality (see [12] or [20]). Henceforth, we restrict ourselves to the binary adversary, which just sets gains of 0 or 1.

arbitrary distribution F_t over reals. Since F_t could be arbitrary, this is an infinite-parameter family of algorithms. We denote the

1. single parameter MWA family by $\mathcal{A}_{\text{single}}$;
2. family where $\eta(t)$ decreases with t by \mathcal{A}_{dec} ;
3. family where $\eta(t)$ is arbitrary function of t by \mathcal{A}_{arb} ;
4. family where $\eta(t)$ is drawn from F_t for each t by $\mathcal{A}_{\text{rand}}$.

It is straightforward to see that $\mathcal{A}_{\text{single}} \subseteq \mathcal{A}_{\text{dec}} \subseteq \mathcal{A}_{\text{arb}} \subseteq \mathcal{A}_{\text{rand}}$. The reason we start with $\mathcal{A}_{\text{single}}$ is that it is the classic MWA and precisely characterizing its regret is still open. We study \mathcal{A}_{dec} because often when MWA algorithms are working with unknown T , they employ a strategy where η decreases with time. We move on to further significantly generalize this by studying $\mathcal{A}_{\text{arb}}, \mathcal{A}_{\text{rand}}$.

Minimax regret, and Notation

We study the standard notion of minimax regret for each of the above family of algorithms. Formally, let $R_{T,k}(A, D)$ denote the expected regret achieved by algorithm A when faced with adversary D in the prediction with expert advice game with T steps and k experts. We use $R_k(A, D)$ to denote the asymptotic², in T , value of $R_{T,k}(A, D)$, i.e., $R_k(A, D) = \sqrt{T} \cdot \lim_{T \rightarrow \infty} \frac{R_{T,k}(A, D)}{\sqrt{T}}$. The minimax regret of a family $\mathcal{A}_{\mathcal{F}}$ of algorithms against a family $\mathcal{D}_{\mathcal{F}}$ of adversaries is given by $R_{T,k}(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{\mathcal{F}}) = \min_{A \in \mathcal{A}_{\mathcal{F}}} \max_{D(A) \in \mathcal{D}_{\mathcal{F}}} R_{T,k}(A, D)$. Let $\mathcal{D}_{\text{univ}}$ denote the universe of all adversaries. We use the shorthand $R_{T,k}(\mathcal{A}_{\mathcal{F}})$ for $R_{T,k}(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{\text{univ}}) = \min_{A \in \mathcal{A}_{\mathcal{F}}} \max_{D(A) \in \mathcal{D}_{\text{univ}}} R_{T,k}(A, D)$. We use $R_k(\mathcal{A}_{\mathcal{F}})$ to denote the asymptotic, in T , value of $R_{T,k}(\mathcal{A}_{\mathcal{F}})$, i.e., $R_k(\mathcal{A}_{\mathcal{F}}) = \sqrt{T} \cdot \lim_{T \rightarrow \infty} \frac{R_{T,k}(\mathcal{A}_{\mathcal{F}})}{\sqrt{T}}$.

Goal

One of our goals in this paper is to compute the precise values of $R_k(\mathcal{A}_{\text{single}})$, $R_k(\mathcal{A}_{\text{dec}})$, $R_k(\mathcal{A}_{\text{arb}})$ and $R_k(\mathcal{A}_{\text{rand}})$ for each value of k , and, *describe and compute* the adversarial sequences that realize these regrets. For clarity, we compute the precise values of $R_k(\mathcal{A}_{\mathcal{F}})$ by:

1. computing the best-response adversary in $\mathcal{D}_{\text{univ}}$ for every algorithm in $\mathcal{A}_{\mathcal{F}}$;
2. computing $R_k(\mathcal{A}_{\mathcal{F}})$ the regret of the optimal algorithm in $\mathcal{A}_{\mathcal{F}}$ (i.e., the algorithm that gets the smallest regret w.r.t. its best-response adversary).

In many cases, the first step, namely computing the best-response adversary, is challenging. We find the best-response adversaries for the families $\mathcal{A}_{\text{single}}$ and \mathcal{A}_{dec} . For the families \mathcal{A}_{arb} and $\mathcal{A}_{\text{rand}}$, we perform the first step approximately, i.e., we compute a nearly best-response adversary, and thus we obtain lower bounds on $R_k(\mathcal{A}_{\text{arb}})$ and $R_k(\mathcal{A}_{\text{rand}})$.

What is known, and what to expect?

It is well known that for $\mathcal{A}_{\mathcal{F}} = \mathcal{A}_{\text{single}}, \mathcal{A}_{\text{dec}}, \mathcal{A}_{\text{arb}}, \mathcal{A}_{\text{rand}}$: $R_{T,k}(\mathcal{A}_{\mathcal{F}}) \leq \sqrt{(T \ln k)/2}$ for all T, k , and in the doubly asymptotic limit, as both T and k go to ∞ , the optimal regret of $\mathcal{A}_{\text{single}}$ is $\sqrt{(T \ln k)/2}$, i.e., $\lim_{T \rightarrow \infty, k \rightarrow \infty} (R_{T,k}(\mathcal{A}_{\text{single}})/\sqrt{(T \ln k)/2}) = 1$. (see [6, 5]). While there are useful applications for $k \rightarrow \infty$, there are also several interesting use-cases of the experts problem with just a few experts (rain-or-shine ($k = 2$), buy-or-sell-or-hold ($k = 3$)). It seems like for small k such as 2, 3, 4 etc. $R_k(\mathcal{A}_{\text{single}})$ could be a significant constant factor smaller than $\sqrt{(T \ln k)/2}$. And given that families like \mathcal{A}_{dec} etc. are supersets of $\mathcal{A}_{\text{single}}$,

² Although R_k doesn't have a T in the subscript, R_k is still dependent on T . We suppress T merely to indicate asymptotics in T .

it seems even more likely that $R_k(\mathcal{A}_{\text{dec}})$ etc. are constant factor smaller than $\sqrt{(T \ln k)/2}$. Surprisingly, we show that is not the case: the regret of $\sqrt{(T \ln k)/2}$ that is obtained as $k \rightarrow \infty$ is *already obtained at* $k = 2$. Thus our work completely closes the gap between upper and lower bounds for all k .

1.1 Main Results

Finite horizon model

1. $R_k(\mathcal{A}_{\text{single}}) = R_k(\mathcal{A}_{\text{dec}}) = \sqrt{\frac{T \ln k}{2}}$ for even k ,
 $R_k(\mathcal{A}_{\text{single}}) \geq R_k(\mathcal{A}_{\text{dec}}) \geq \sqrt{\frac{T \ln k}{2} (1 - \frac{1}{k^2})}$ for odd k .
2. $R_k(\mathcal{A}_{\text{arb}}) \geq R_k(\mathcal{A}_{\text{rand}}) \geq \frac{2}{3} \sqrt{\frac{T \ln k}{2}}$ for even k ,
 $R_k(\mathcal{A}_{\text{arb}}) \geq R_k(\mathcal{A}_{\text{rand}}) \geq \frac{2}{3} \sqrt{\frac{T \ln k}{2} (1 - \frac{1}{k^2})}$ for odd k .

Geometric horizon model

In the geometric horizon model, the current time t is not relevant, since the expected remaining time for which the game lasts is the same irrespective of how many steps have passed in the past. Thus $\eta(t)$ is without loss of generality, independent of t . Nevertheless, η could still depend on other aspects of the history of the game, like the cumulative gains of all the experts etc. We establish some quick notation before discussing results. Let δ denote the probability that the game stops at any given step, independently of the past (and therefore the expected length of the game is $\frac{1}{\delta}$). Let $R_{\delta,k}(A, D)$ denote the regret achieved by algorithm A when faced with adversary D in the prediction with expert advice game with stopping probability δ and k experts. The minimax regret for a family $\mathcal{A}_{\mathcal{F}}$ of algorithms is given by $R_{\delta,k}(\mathcal{A}_{\mathcal{F}}) = R_{\delta,k}(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{\text{univ}}) = \min_{A \in \mathcal{A}_{\mathcal{F}}} \max_{D \in \mathcal{D}_{\text{univ}}} R_{\delta,k}(A, D)$. Let³ $R_k(\mathcal{A}_{\mathcal{F}}) = \frac{1}{\sqrt{\delta}} \lim_{\delta \rightarrow 0} \sqrt{\delta} \cdot R_{\delta,k}(\mathcal{A}_{\mathcal{F}})$.

We show the following:

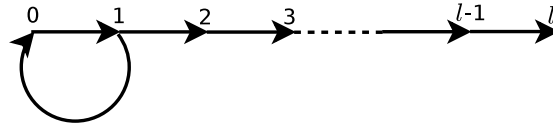
1. $R_2(\mathcal{A}_{\text{single}}) = \frac{0.391}{\sqrt{\delta}}$,
2. $R_k(\mathcal{A}_{\text{single}}) \geq \frac{1}{2} \sqrt{\frac{\ln k}{2\delta}}$ for all k .

The regret lower bound of $\frac{1}{2} \sqrt{\frac{\ln k}{2\delta}}$ we obtain is at most a factor 2 away from the regret upper bound of $\sqrt{\frac{\ln k}{2\delta}}$. Further, we show that the adversarial family that we use for the family of algorithms $\mathcal{A}_{\text{single}}$ to obtain the precise regret for 2 experts, also obtains the optimal regret for the universe $\mathcal{A}_{\text{univ}}$ of *all algorithms*. See the full version [11] for more on this result.

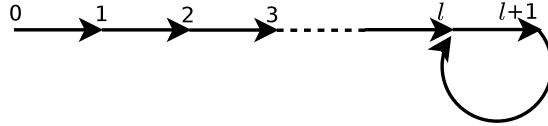
1.2 Simple adversarial primitives and families

While the optimal regret $R_k(\mathcal{A}_{\mathcal{F}})$ is defined by optimizing over the most general family $\mathcal{D}_{\text{univ}}$ of all adversaries, (i.e., $R_k(\mathcal{A}_{\mathcal{F}}) = R_k(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{\text{univ}})$) one of our primary contributions in this work is to develop simple and analytically easy-to-work-with adversarial primitives that we use to construct adversarial families (call a typical such family $\mathcal{D}_{\text{simple}}$) such that:

³ Note that the notation $R_k(\cdot)$ is overloaded: it could refer to finite or geometric horizon setting depending on the context. But since the setting is clear from the context, we drop the δ vs T .



■ **Figure 1** Optimal finite horizon adversary.



■ **Figure 2** Optimal geometric horizon adversary.

- $\mathcal{D}_{\text{simple}}$ is simple to describe and to optimize over, i.e., computing $\max_{D \in \mathcal{D}_{\text{simple}}} R_{T,k}(A, D)$ is much simpler than computing $\max_{D \in \mathcal{D}_{\text{univ}}} R_{T,k}(A, D)$.
- optimizing over $\mathcal{D}_{\text{simple}}$ is guaranteed to be as good (or approximately as good) as optimizing over $\mathcal{D}_{\text{univ}}$ for many algorithmic families $\mathcal{A}_{\mathcal{F}}$, i.e., $R_k(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{\text{univ}}) = R_k(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{\text{simple}})$ for many $\mathcal{A}_{\mathcal{F}}$. As $R_k(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{\text{univ}}) \geq R_k(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{\text{simple}})$, the non-trivial part is to prove (approximate) equality for $\mathcal{A}_{\mathcal{F}}$.

We demonstrate the versatility of our primitives by using simple combinations of them to develop sharp lower bounds to algorithmic families $\mathcal{A}_{\text{single}}$, \mathcal{A}_{dec} , \mathcal{A}_{arb} , and $\mathcal{A}_{\text{rand}}$. There is a lot of room for further combinations of primitives that might be useful to construct adversarial families tailored to other algorithmic families.

The “looping” and “straight-line” primitives

These primitives are best described by focusing on the case of $k = 2$ experts. In the two experts case, the algorithm makes its decision at step t , by just looking at the difference d of the cumulative gains of the leading and lagging experts’ cumulative gains. As such, the adversary has to simply control how the difference d evolves over time. The “looping” primitive simply loops the value of d between 0 and 1 indefinitely (i.e., advances⁴ one expert in one step and advances the other in the next step and so on, so that d simply loops between 0 and 1). The “straight-line” adversary simply keeps advancing the value of d by 1 at each step. Interestingly, the worst-case adversary for each of the finite and geometric horizon settings is a composition of looping and straight-line primitives. Strikingly, despite the apparent similarity between two settings, the optimal adversaries in the two models turn out to be “mirror images” of each other. The optimal adversary in finite horizon loops first and then goes in straight-line, while the geometric horizon’s optimal does the reverse. The structure of these two families is depicted in Figures 1, 2, that shows the evolution of the difference d between the cumulative gains of the leading and lagging experts. This fundamental difference between the structures of the optimal adversary in these two settings also manifests in the optimal regret values of these two settings.

The generalizations of these primitives for arbitrary k is straightforward. The looping primitive partitions the set of experts into two teams, say A and B , and then it advances all experts in team A in one step and in team B in the other, and so on. The straight-line primitive picks an arbitrary expert and keeps advancing that expert by 1 in each step.

⁴ Advances here refers to setting the gain of that expert to 1.

Combining the primitives

Here's how we create effective adversarial families from these primitives. In fact the families are often trivial, i.e., they have only one member and therefore there's nothing to optimize. We ignore the odd and even k distinctions here for ease of description and just focus on the even k case. Please see the technical sections for precise descriptions, which is only slightly different from what is here.

1. Perform $\frac{T-\ell}{2}$ loops and then ℓ straight-line steps, for $\ell = T^{3/4}$. Call this adversary \mathcal{D}_{lsdet} (stands for loop-straight-deterministic). Clearly, this adversarial family is simple-to-describe and there is nothing to optimize here as there is only one member in the family. Most importantly, it gives the precisely optimal regret for algorithmic families $\mathcal{A}_{\text{single}}$ and \mathcal{A}_{dec} as $T \rightarrow \infty$. I.e.,

$$\text{For families } \mathcal{A}_{\mathcal{F}} = \mathcal{A}_{\text{single}}, \mathcal{A}_{\text{dec}}: \quad R_k(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{lsdet}) = \sqrt{\frac{T \ln k}{2}} = R_k(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{\text{univ}}).$$

The best known regret lower bound for $\mathcal{A}_{\text{single}}$ was $\frac{1}{4} \sqrt{T \log_2 k}$ [13], which leaves a factor 2.35 gap between upper and lower bounds, that our work closes. We are not aware of prior lower bounds for \mathcal{A}_{dec} .

2. Perform $\frac{T-r}{2}$ loops and then r straight-line steps, where r is chosen uniformly at random from $\{0, 1, \dots, T^{3/4}\}$. This family is simple and there is nothing to optimize here as well. Call this adversary \mathcal{D}_{lsrand} (denoting loop, straight, uniformly random). We show that when $\mathcal{A}_{\mathcal{F}} = \mathcal{A}_{\text{arb}}$ or when $\mathcal{A}_{\mathcal{F}} = \mathcal{A}_{\text{rand}}$:

$$\text{For families } \mathcal{A}_{\mathcal{F}} = \mathcal{A}_{\text{arb}}, \mathcal{A}_{\text{rand}}: \quad R_k(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{lsrand}) \geq \frac{2}{3} \sqrt{\frac{T \ln k}{2}} \geq \frac{2}{3} R_k(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{\text{univ}}).$$

Note that while this lower bound doesn't precisely match the upper bound, the upper bound $R_k(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{\text{univ}}) \leq \sqrt{(T \ln k)/2}$ and is likely even smaller for small k (particularly for a large family of algorithms like \mathcal{A}_{arb} or $\mathcal{A}_{\text{rand}}$) — *thus our result shows that the ratio between upper and lower bounds is at most $\frac{3}{2}$ and likely even smaller*. To the best of our knowledge our lower bound is the first for the classes \mathcal{A}_{arb} and $\mathcal{A}_{\text{rand}}$.

3. In geometric horizon, even for the family $\mathcal{A}_{\text{single}}$ and at $k = 2$ experts, instead of a single adversary working for all members of $\mathcal{A}_{\text{single}}$, we have a single-parameter family of adversaries to optimize over. Namely, follow the straight-line primitive for r steps and then the looping primitive for $\frac{T-r}{2}$ steps. Call this single-parameter family (parameterized by r) as \mathcal{D}_{sl} . The exact number r is determined by optimizing it as a function of the parameter η used by the algorithm in $\mathcal{A}_{\text{single}}$. Specifically, for the case of 2 experts we show that: $R_2(\mathcal{A}_{\text{single}}, \mathcal{D}_{sl}) = \frac{0.391}{\sqrt{\delta}} = R_2(\mathcal{A}_{\text{single}}, \mathcal{D}_{\text{univ}})$. Note that \mathcal{D}_{sl} is again simple-to-describe and straightforward-to-optimize over. Further, it is the precisely optimal adversary family for not just $\mathcal{A}_{\text{single}}$ but also the *universe of all algorithms* $\mathcal{A}_{\text{univ}}$ (see the full version [11] for this result), i.e., $R_2(\mathcal{A}_{\text{univ}}, \mathcal{D}_{\text{univ}}) = R_2(\mathcal{A}_{\text{univ}}, \mathcal{D}_{sl})$.
4. But in the geometric horizon setting, if we don't shoot for the precisely optimal adversary family, and aim for just approximately optimal, then we don't need a single-parameter family: just following one of the two looping/straight-line primitives gives a lower bound of $\frac{1}{2} \sqrt{\frac{\ln k}{2\delta}}$. Let $\mathcal{D}_{\ell}, \mathcal{D}_s$ be the looping and straight line primitives. Then: $R_k(\mathcal{A}_{\text{single}}, \{\mathcal{D}_{\ell}, \mathcal{D}_s\}) \geq \frac{1}{2} \sqrt{\frac{\ln k}{2\delta}} \geq \frac{1}{2} R_k(\mathcal{A}_{\text{single}}, \mathcal{D}_{\text{univ}})$. Note that while this lower bound doesn't precisely match the upper bound $R_k(\mathcal{A}_{\text{single}}, \mathcal{D}_{\text{univ}})$, the latter is at most⁵ $\sqrt{\frac{\ln k}{2\delta}}$, which is at most a factor 2 larger than lower bound.

⁵ This is a simple extension of the standard proof that MWA has a regret upper bound of $\sqrt{(T \ln k)/2}$ in

► **Remark.** To give a sense that the primitives offer enough variety in combination, here is a simple modification over the adversary $D_{lstrand}$, that we call $D_{lstrand++}$: use $D_{lstrand}$ with probability p , and with probability $1 - p$ play the looping primitive D_l for all the T steps. This increases the lower bound from $\frac{2}{3}\sqrt{(T \ln k)/2}$ to $0.68\sqrt{(T \ln k)/2}$ (see the full version [11] for this result). We believe that this can be increased further by picking the stopping time for looping from a non-uniform distribution etc.

1.3 Motivation and discussion

In this work we seek to understand the structure of worst case input sequences for a broad family of algorithms and crisply expose their vulnerabilities. By identifying such structures, we also get the precise regret suffered by them. Our motivation in exploring this question includes the following.

1. After 25 years since MWA was introduced [19, 27], we do not have a sharp regret bound for it. \mathcal{A}_{single} is known to suffer a regret of at most $\sqrt{\frac{T \ln k}{2}}$, but the best known lower bound on regret is $\frac{1}{4}\sqrt{T \log_2 k}$ [13], with a factor 2.35 gap between these two bounds. For larger families like \mathcal{A}_{arb} , \mathcal{A}_{rand} no lower bounds were known. For an algorithm as widely used as MWA, it is fruitful to have a sharp regret characterization.
2. The patterns in the worst-case adversarial sequences that we characterize are simple to spot if they exist (or even if anything close exists), and make simple amends to the algorithm that result in significant gains.
3. The problem is theoretically clean and challenging: how powerful are simple input patterns beyond the typically used pure random sequences in inflicting regret?

Related Work

Classic works: The book by Cesa-Bianchi and Lugosi [7] is an excellent source for both applications and references for prediction with expert advice. The prediction with experts advice paradigm was introduced by Littlestone and Warmuth [19] and Vovk [27]. The famous multiplicative weights update algorithm was introduced independently by these two works: as the weighted majority algorithm by [19] and as the aggregating algorithm by [27]. The pioneering work of Cesa-Bianchi et al. [6] considered $\{0, 1\}$ outcome space for nature and showed that for the absolute loss function $\ell(x, y) = |x - y|$ (or $g(x, y) = 1 - |x - y|$), the asymptotically optimal regret is $\sqrt{\frac{T \ln k}{2}}$. This was later extended to $[0, 1]$ outcomes for nature by Haussler et al. [15]. The asymptotic optimality of $\sqrt{\frac{T \ln k}{2}}$ for arbitrary loss (gain) functions follows from the analysis of Cesa-Bianchi [5]. When it is known beforehand that the cumulative loss of the optimal expert is going to be small, the optimal regret can be considerably improved, and such results were obtained by Littlestone and Warmuth [19] and Freund and Schapire [9]. With certain assumptions on the loss function, the simplest possible algorithm of following the best expert already guarantees sub-linear regret (see Hannan [14]). Even when the loss functions are unbounded, if the loss functions are exponential concave, sub-linear regret can still be achieved as shown by Blum and Kalai [4]. While our work is focused on the worst-case adversaries for the MWA family of algorithms, Freund and Schapire [10] prove lower bounds for general adaptive game-playing algorithms.

the finite horizon setting with T steps and k experts, and the realization that in the geometric horizon setting, the expected stopping time is $\frac{1}{\delta}$.

Recent works: Gravin et al. [12] give the minimax optimal algorithm, and the regret for the prediction with expert advice problem for the cases of $k = 2$ and $k = 3$ experts. The focus of [12] was providing a regret upper bound for the family of all algorithms, while the focus of this paper is to provide regret lower bounds for large families of algorithms. Luo and Schapire [20] consider a setting where the adversary is restricted to pick gain vectors from the basis vector space $\{\mathbf{e}_1, \dots, \mathbf{e}_k\}$. Abernethy et al. [3] consider a different variant of experts problem where the game stops when cumulative loss of any expert exceeds given threshold. Abernethy et al. [2] consider general convex games and compute the minimax regret exactly when the input space is a ball, and show that the algorithms of Zinkevich [28] and Hazan et al. [16] are optimal w.r.t. minimax regret. Abernethy et al. [1] provide upper and lower bounds on the regret of an optimal strategy for several online learning problems without providing algorithms, by relating the optimal regret to the behavior of a certain stochastic process. Mukherjee and Schapire [23] consider a continuous experts setting where the algorithm knows beforehand the maximum number of mistakes of the best expert. Rakhlin et al. [25] introduce the notion of sequential Rademacher complexity and use it to analyze the learnability of several problems in online learning w.r.t. minimax regret. Rakhlin et al. [26] use the sequential Rademacher complexity introduced in [25] to analyze learnability w.r.t. general notions of regret (and not just minimax regret). Rakhlin et al. [24] use the notion of conditional sequential Rademacher complexity to find relaxations of problems like prediction with static experts that immediately lead to algorithms and associated regret guarantees. They show that the random playout strategy has a sound basis and propose a general method to design algorithms as a random playout. Koolen [17] studies the regret w.r.t. every expert, rather than just the best expert in hindsight and considers tradeoffs in the Pareto-frontier. McMahan and Abernethy [21] characterize the minimax optimal regret for online linear optimization games as the supremum over the expected value of a function of a martingale difference sequence, and similar characterizations for the minimax optimal algorithm and the adversary. McMahan and Orabona [22] study online linear optimization in Hilbert spaces and characterize minimax optimal algorithms. Chaudhuri et al. [8] describe a parameter-free learning algorithm motivated by the cases of large number of experts k . Koolen and Erven [18] develop a prediction strategy called Squint, and prove bounds that incorporate both quantile and variance guarantees.

2 Finite horizon

We begin our analysis of MWA by focusing on the simple case of $k = 2$ experts. We first identify the structure of the optimal adversary, and through it we obtain the tight regret bound as $T \rightarrow \infty$. Before proceeding further, it is useful to recall that when the gains of the leading and lagging experts are given by $g + d$ and g , the MWA algorithm follows these experts with probabilities $\frac{e^{\eta d}}{e^{\eta d} + 1}$ and $\frac{1}{e^{\eta d} + 1}$ respectively. Thus, when the adversary increases d by 1 i.e., increases the gain of the leading expert by 1, the regret benchmark (namely, the gains of the leading expert) increases by 1, where as MWA is correct only with probability $\frac{e^{\eta d}}{e^{\eta d} + 1}$, and this therefore inflicts a regret of $\frac{1}{e^{\eta d} + 1}$ on MWA. On the other hand, if the adversary decreases d by 1, then the benchmark doesn't change, where as MWA succeeds with probability $\frac{1}{e^{\eta d} + 1}$, and this therefore inflicts a regret of $\frac{-1}{e^{\eta d} + 1}$. When the adversary doesn't change d , the regret inflicted is 0.

Structure of the optimal adversary

Let η be the fixed update rate of the optimal MWA (the parameter in the exponent as explained in Section 1)⁶. Against a specific algorithm, an optimal adversary can always be found in the class of deterministic adversaries. The actions of the optimal adversary (against a specific MWA algorithm) depend only on the distance d between leading and lagging experts and time step t .

1. **Loop aggregation:** At each time step, the adversary may either increase or decrease the gap d by 1, or leave d unchanged. We denote these actions of the adversary by $d \xrightarrow{t} d + 1$, $d \xrightarrow{t} d - 1$, and $d \xrightarrow{t} d$. The respective regret values inflicted on the algorithm are given by $\frac{1}{e^{\eta d+1}}$, $\frac{-1}{e^{\eta d+1}}$, and 0, which are all independent of the time when an action was taken. This means that if the adversary loops between d and $d + 1$ at several disconnected points of time, it may as well aggregate all of them and complete all of them in consecutive time steps. I.e., the optimal adversary starts at $d = 0$ and then weakly monotonically increases d , stopping at various points $d = s$, looping for an arbitrary length of time between $d = s$ and $d = s - 1$ and then proceeding forward.
2. **Staying at same d is dominated:** It is not hard to see that any action $x \rightarrow x$ is dominated for the adversary as this wastes a time step and inflicts 0 regret on the algorithm. Thus the “weakly monotonically increases” in the previous paragraph can be replaced by “strictly monotonically increases” (except of course for the stopping points for looping).
3. **Loop(0) domination:** Define $\text{Loop}(d) \stackrel{\text{def}}{=} d \rightarrow d + 1 \rightarrow d$. It is easy to see that the regret inflicted by $\text{Loop}(d)$ is exactly $\frac{1}{e^{\eta d+1}} - \frac{1}{e^{\eta(d+1)+1}}$ and this quantity is maximized at $d = 0$. Thus, the optimal adversary should replace all loops by loops at 0. This gives us the structure claimed in Figures 1, 2 for the optimal adversary.

Given the optimal adversary’s structure (as described in Figures 1, 2) w.l.o.g. we can assume it to be looping for $\frac{T-\ell}{2}$ steps at 0 and then monotonically increasing d for ℓ steps at which point the game ends. In the following we will analyze the regret inflicted by the optimal adversary (which we showed was optimal for the class of algorithm $\mathcal{A}_{\text{single}}$) against a broader class \mathcal{A}_{dec} of MWA. The regret of the adversary is:

$$\sum_{t=1}^{\frac{T-\ell}{2}} \left[\frac{1}{2} - \frac{1}{e^{\eta(2t)+1}} \right] + \sum_{d=0}^{\ell-1} \frac{1}{e^{\eta(T-\ell+d+1)d+1}}. \quad (1)$$

Asymptotic regret of the optimal adversary

We first notice that for a fixed adversary with a given ℓ , the regret of MWA with decreasing $\eta(t)$ in (1) is greater than or equal to the regret of MWA with a constant $\eta' = \eta(T - \ell)$, i.e.,

$$\frac{T-\ell}{2} \left[\frac{1}{2} - \frac{1}{e^{\eta'+1}} \right] + \sum_{d=0}^{\ell-1} \frac{1}{e^{d\eta'+1}}. \quad (2)$$

This is true as each individual term in (2) is equal to or smaller than the corresponding term in (1). In the following we are going to use $\ell = T^{3/4}$ for the adversary and for convenience, we write $e^{\eta'(T-\ell)} = \tau = 1 + \frac{\alpha}{\sqrt{T}}$. The two terms in (1) together place strong bounds on what α should be: they imply that $\alpha = \Theta(1)$. We show this in 2 steps: first we show that $\alpha = O(1)$, and then show that $\alpha = \Omega(1)$.

⁶ In fact, we can identify the optimal adversary for a much broader family of algorithms (see the full version [11] for more details).

48:10 Tight Lower Bounds for Multiplicative Weights Algorithmic Families

1. The first term in (2) forces α to be $O(1)$. The regret of MWA for $\ell = T^{3/4}$ is at least

$$\frac{T - \ell}{2} \left[\frac{1}{2} - \frac{1}{e^{\eta'} + 1} \right] \simeq \frac{T}{2} \left[\frac{1}{2} - \frac{1}{e^{\eta'} + 1} \right] = \frac{\alpha \sqrt{T}}{4(1 + e^{\eta'})} = \frac{\sqrt{T}}{4(\frac{2}{\alpha} + \frac{1}{\sqrt{T}})}.$$

Since MWA's regret upper bound in the finite horizon model is $\Theta(\sqrt{T})$, α must be $O(1)$.

2. To show that $\alpha = \Theta(1)$ we argue that the regret from the second term of (2) is $\omega(\sqrt{T})$ when $\alpha = o(1)$. For all $d \leq \frac{\sqrt{T}}{\alpha}$, we have $\tau^d = (1 + \frac{\alpha}{\sqrt{T}})^d \leq e$. Thus MWA's regret for $k = \min(\frac{\sqrt{T}}{\alpha}, T^{3/4})$ is at least

$$\sum_{d=0}^{k-1} \frac{1}{\tau^d + 1} \geq \sum_{d=0}^{k-1} \frac{1}{e + 1} = \Omega(k) = \omega(\sqrt{T}).$$

Since MWA's regret upper bound in the finite horizon model is $\Theta(\sqrt{T})$, we get $\alpha = \Omega(1)$.

Now, we obtain the following asymptotic estimate for the second part of (2), where $\eta' \sim e^{\eta'} - 1 = \frac{\alpha}{\sqrt{T}}$.

$$\sum_{d=0}^{\ell-1} \frac{1}{\tau^d + 1} \sim \int_0^{\ell} \frac{dx}{e^{\eta'x} + 1} = \frac{1}{\eta'} \ln \left(2 \frac{e^{\ell\eta'}}{e^{\ell\eta'} + 1} \right) \sim \frac{\sqrt{T}}{\alpha} \left(\ln(2) - \ln(1 + e^{-\ell\eta'}) \right). \quad (3)$$

The first part of (2) can be estimated as follows

$$\frac{T - \ell}{2} \left[\frac{1}{2} - \frac{1}{e^{\eta'} + 1} \right] \sim \frac{T}{2} \left[\frac{e^{\eta'} - 1}{2(e^{\eta'} + 1)} \right] \sim \frac{T}{2} \cdot \frac{\eta'}{4} = \frac{\alpha \sqrt{T}}{8}. \quad (4)$$

As $e^{-\ell\eta'} = e^{-\alpha T^{1/4}} = o(1)$, (3) simplifies to $\frac{\ln(2)\sqrt{T}}{\alpha}$, while the estimate for (4) is $\frac{\alpha\sqrt{T}}{8}$. Now the estimate $\sqrt{T} \left[\frac{\ln(2)}{\alpha} + \frac{\alpha}{8} \right]$ for the regret in (2) is minimized for the choice of parameter $\alpha = \sqrt{8 \ln(2)}$. Then the regret of the optimal MWA is at least $\sqrt{T \cdot \frac{\ln(2)}{2}} (1 + o(1))$. It is known that there is MWA for $k = 2$ experts with regret at most $\sqrt{T \cdot \frac{\ln(2)}{2}}$ (asymptotic in T). Thus, we obtain the following claim 1 (in the claim below, by "optimal MWA" we mean the MWA with the optimally tuned $\eta(t) = \eta'$).

► **Claim 1.** For $\mathcal{A}_{\mathcal{F}} = \mathcal{A}_{\text{single}}, \mathcal{A}_{\text{dec}}$: $R_2(\mathcal{A}_{\mathcal{F}}, D_{\text{lsdet}}) = \sqrt{\frac{T \ln 2}{2}} = R_2(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{\text{univ}})$.

We generalize the adversary for $k = 2$ and obtain a tight lower bound for \mathcal{A}_{dec} matching the known upper bound for arbitrary *even* number k of experts and almost matching bound for *odd* number k of experts. Since $R_k(\mathcal{A}_{\text{single}}) \geq R_k(\mathcal{A}_{\text{dec}})$, the lower bound in Theorem 2 below applies to $\mathcal{A}_{\text{single}}$ as well.

► **Theorem 2.** For $\mathcal{A}_{\mathcal{F}} = \mathcal{A}_{\text{single}}, \mathcal{A}_{\text{dec}}$:

(i) For *even* k : $R_k(\mathcal{A}_{\mathcal{F}}, D_{\text{lsdet}}) = \sqrt{\frac{T \cdot \ln k}{2}} = R_k(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{\text{univ}})$.

(ii) For *odd* k : $R_k(\mathcal{A}_{\mathcal{F}}, D_{\text{lsdet}}) \geq \sqrt{\frac{T \cdot \ln k}{2} \left(1 - \frac{1}{k^2} \right)} \geq \sqrt{1 - \frac{1}{k^2}} R_k(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{\text{univ}})$.

Proof. Let $\eta(t)$ be the update rate of the optimal MWA, we define $\ell = T^{3/4}$ and $\eta' = \eta(T - \ell)$. We employ the following adversary for the even k number of experts:

1. Divide all experts into two equal parties, numbered A and B . For the first $\frac{T-\ell}{2}$ rounds ($\ell = T^{3/4}$), advance all the experts in party A in even numbered rounds, and all experts in party B in odd numbered rounds.
2. For the remaining ℓ steps, pick an arbitrary expert and keep advancing just that expert.

Similar to (1) this adversary obtains the regret of at least $\sum_{t=1}^{\frac{T-\ell}{2}} \left[\frac{1}{2} - \frac{1}{e^{\eta(2t)}+1} \right] + \sum_{d=0}^{\ell-1} \frac{k-1}{e^{d \cdot \eta(T-\ell+d+1)}+k-1}$. We further notice that similar to (2) the regret of MWA with decreasing $\eta(t)$ in the above expression is greater than or equal to the regret of MWA with a constant $\eta' = \eta(T - \ell)$, i.e., the previous expression is at least

$$\frac{T-\ell}{2} \left[\frac{1}{2} - \frac{1}{e^{\eta'}+1} \right] + \sum_{d=0}^{\ell-1} \frac{k-1}{e^{d \cdot \eta'}+k-1}. \quad (5)$$

We use (4) to estimate the first term of (5). We estimate the second term of (5) similar to (3) as follows.

$$\sum_{d=0}^{\ell-1} \frac{k-1}{e^{d \eta'}+k-1} \sim \int_0^\ell \frac{(k-1)dx}{e^{x \eta'}+k-1} = \frac{1}{\eta'} \ln \left(\frac{k \cdot e^{\ell \eta'}}{e^{\ell \eta'}+k-1} \right) \sim \frac{\sqrt{T} \ln(k)}{\alpha}. \quad (6)$$

Now, combining these two estimates the regret from (5) is at least

$$\frac{\alpha \sqrt{T}}{8} + \frac{\sqrt{T} \ln(k)}{\alpha} \geq 2 \cdot \sqrt{\frac{\alpha \sqrt{T}}{8} \cdot \frac{\sqrt{T} \ln(k)}{\alpha}} = \sqrt{\frac{T \ln(k)}{2}},$$

which precisely matches the upper bound on the regret of MWA[6].

For the odd k number of experts we employ almost the same adversary as for even k , although, since k now is odd, we split experts into two parties of *almost* equal sizes (see the full version [11] for more details). ◀

2.1 General variations of MWA

We have seen that the best known MWA with a flat learning rate η achieves optimal (or almost optimal in the case of odd number of experts) regret among all MWAs with monotone decreasing learning rates $\eta(t)$. However, it seems that in the finite horizon model a better strategy for tuning parameters of MWA would be to use higher rates $\eta(t)$ towards the end T . In the following we study a broader family of MW algorithms \mathcal{A}_{arb} where learning parameter $\eta(t)$ can vary in an arbitrary way. In the following theorem we show that such adaptivity of MWA cannot decrease the regret of the algorithm by more than a factor of $2/3$.

▶ **Remark.** In fact, our analysis extends to the family \mathcal{A}_{rand} where each $\eta(t)$ can be a random variable drawn from a distribution F_t . Effectively, with a random $\eta(t)$ the algorithm player can get any convex combination $f(G_{i,t-1}, t) = \mathbf{E}_{\eta(t)}[e^{\eta(t)G_{i,t-1}}]$ of $e^{\eta(t)G_{i,t-1}}$ in the vector of probabilities for following each expert i at time t . This constitutes a much richer family of algorithms compared to the standard single parameter MWA family.

▶ **Theorem 3.** For $\mathcal{A}_{\mathcal{F}} = \mathcal{A}_{arb}, \mathcal{A}_{rand}$:

- (i) For even k : $R_k(\mathcal{A}_{\mathcal{F}}, D_{lsrand}) \geq \frac{2}{3} \sqrt{\frac{T \cdot \ln k}{2}} \geq \frac{2}{3} R_k(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{univ})$.
- (ii) For odd k : $R_k(\mathcal{A}_{\mathcal{F}}, D_{lsrand}) \geq \frac{2}{3} \sqrt{\frac{T \cdot \ln k}{2} \left(1 - \frac{1}{k^2}\right)} \geq \frac{2}{3} \sqrt{1 - \frac{1}{k^2}} R_k(\mathcal{A}_{\mathcal{F}}, \mathcal{D}_{univ})$.

Proof. Define $\ell = T^{3/4}$ and $R = \lceil \frac{T-\ell-1}{2} \rceil$. We use the following adversary for even number of experts k :

1. Choose $j \in [R]$ uniformly at random. With probability 0.5 don't advance any expert in the first step.
2. Divide all experts into two equal parties, numbered A and B . For the next j rounds, advance all the experts in party A in even numbered rounds, and all experts in party B in odd numbered rounds.
3. For next ℓ steps, pick any expert i and keep advancing just expert i . Do nothing in remaining steps.

The regret of the algorithm is

$$\begin{aligned} & \frac{1}{R} \sum_{j=0}^{R-1} \left[\frac{1}{2} \left(\sum_{t=1}^j \left[\frac{1}{2} - \frac{1}{e^{\eta(2t)} + 1} \right] + \sum_{d=0}^{\ell-1} \frac{k-1}{e^{d \cdot \eta(2j+d+1)} + k-1} \right) + \right. \\ & \left. \frac{1}{2} \left(\sum_{t=1}^j \left[\frac{1}{2} - \frac{1}{e^{\eta(2t+1)} + 1} \right] + \sum_{d=0}^{\ell-1} \frac{k-1}{e^{d \cdot \eta(2j+d+2)} + k-1} \right) \right]. \end{aligned} \quad (7)$$

Since $\eta(t)$ can be arbitrary nonnegative number, we break (7) into terms with the same $\eta(t)$ (we also drop a few terms to simplify the expression). In the following, we will also assume that $e^{\eta(t)} = 1 + \frac{\alpha(t)}{\sqrt{T}}$, where $\alpha(t) = \Theta(1)$ for every $t \in [T]$. Later we will explain why this assumption is without loss of generality.

$$\begin{aligned} (7) & \geq \frac{1}{2R} \sum_{t=\ell}^{T-\ell-1} \left([R - \lceil t/2 \rceil] \cdot \left[\frac{1}{2} - \frac{1}{e^{\eta(t)} + 1} \right] + \sum_{d=0}^{\ell-1} \frac{k-1}{e^{\eta(t)d} + k-1} \right) \\ & \simeq \sum_{t=\ell}^{T-\ell-1} \left(\left[\frac{R - \lceil t/2 \rceil}{R} \right] \frac{\alpha(t)}{8\sqrt{T}} + \frac{\sqrt{T} \ln(k)}{2R \cdot \alpha(t)} \right) \geq \sum_{t=\ell}^{T-\ell-1} 2 \sqrt{\left[\frac{R - \lceil t/2 \rceil}{R} \right] \frac{\alpha(t)}{8\sqrt{T}} \frac{\sqrt{T} \ln(k)}{2R \cdot \alpha(t)}} \\ & = \sqrt{\frac{\ln(k)}{2 \cdot 2R}} \sum_{t=\ell}^{T-\ell-1} \sqrt{\left[\frac{R - \lceil t/2 \rceil}{R} \right]} \simeq \sqrt{\frac{\ln(k)}{2 \cdot T}} \int_0^1 \sqrt{1-x} \, dx = \sqrt{\frac{\ln(k)}{2T}} \cdot \frac{2}{3}. \end{aligned} \quad (8)$$

In the above derivation we obtain the first approximation \simeq by using approximations from (4) and (6).

We now argue that the assumption $\alpha(t) = \Theta(1)$ is without loss of generality for every $t \in [T]$. We apply a similar argument as in Theorem 1, but now for each individual term with a particular $\eta(t)$. The term $\sum_{d=0}^{\ell-1} \frac{k-1}{e^{\eta(t)d} + k-1}$ in (8) is already large enough for the estimate when $\alpha(t) = o(1)$. The term $[R - \lceil t/2 \rceil] \cdot \left[\frac{1}{2} - \frac{1}{e^{\eta(t)} + 1} \right]$ also places a strong bound of $O(1)$ on $\alpha(t)$, when $[R - \lceil t/2 \rceil]$ is constant fraction of T . To argue about t close to the threshold T , we can slightly modify the adversary by playing with a small constant probability ε entirely "looping" strategy (without "straight line" part). This would make the coefficient in front of $\left[\frac{1}{2} - \frac{1}{e^{\eta(t)} + 1} \right]$ to be sufficiently large, and at the same time would decrease the lower bound by at most $1 - \varepsilon$ factor. Taking ε arbitrary small we obtain the bound in (8). This concludes the proof for the even number of experts

For the odd number of experts k . We slightly modify the adversary analogous to the case of odd number of experts in Theorem 2. This gives us an additional factor of $1 - \frac{1}{k^2}$ for each of the looping terms.

Note that since our adversary is independent of how the $\eta(t)$'s are picked in the algorithm, it follows immediately that our lower bound applies for $\mathcal{A}_{\mathcal{F}} = \mathcal{A}_{\text{arb}}, \mathcal{A}_{\text{rand}}$. \blacktriangleleft

► Remark. One can slightly improve the lower bound in Theorem 3 and get a better factor than $\frac{2}{3}$. To this end we employ a more complicated adversary by playing with some probability $p > 0$ the same strategy as in Theorem 3 and with the remaining $1 - p$ probability playing purely looping strategy (see the full version [11]).

3 Geometric horizon

We prove two main results in this Section. We derive the structure of the optimal adversary for 2 experts and show that the optimal regret for 2 experts is exactly $\frac{0.391}{\sqrt{\delta}}$ as $\delta \rightarrow 0$. For an arbitrary number of experts k , we derive a regret lower bound of $\frac{1}{2} \sqrt{\frac{\ln(k)}{2\delta}}$ (see the full version [11] for proofs of these results).

References

- 1 Jacob Abernethy, Alekh Agarwal, Peter L. Bartlett, and Alexander Rakhlin. A stochastic view of optimal regret through minimax duality. In *COLT 2009 – The 22nd Conference on Learning Theory, Montreal, Quebec, Canada, June 18-21, 2009*, 2009.
- 2 Jacob Abernethy, Peter L. Bartlett, Alexander Rakhlin, and Ambuj Tewari. Optimal strategies and minimax lower bounds for online convex games. In *21st Annual Conference on Learning Theory – COLT 2008, Helsinki, Finland, July 9-12, 2008*, pages 415–424, 2008.
- 3 Jacob Abernethy, Manfred K. Warmuth, and Joel Yellin. When random play is optimal against an adversary. In *COLT*, pages 437–446, 2008.
- 4 Avrim Blum and Adam Kalai. Universal portfolios with and without transaction costs. *Machine Learning*, 35(3):193–205, June 1999.
- 5 Nicolò Cesa-Bianchi. Analysis of two gradient-based algorithms for on-line regression. In *Proceedings of the Tenth Annual Conference on Computational Learning Theory, COLT'97*, pages 163–170, New York, NY, USA, 1997. ACM.
- 6 Nicolò Cesa-Bianchi, Yoav Freund, David Haussler, David P. Helmbold, Robert E. Schapire, and Manfred K. Warmuth. How to use expert advice. *J. ACM*, 44(3):427–485, May 1997.
- 7 Nicolò Cesa-Bianchi and Gabor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, New York, NY, USA, 2006.
- 8 Kamalika Chaudhuri, Yoav Freund, and Daniel J. Hsu. A parameter-free hedging algorithm. In *Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems 2009. Proceedings of a meeting held 7-10 December 2009, Vancouver, British Columbia, Canada.*, pages 297–305, 2009.
- 9 Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.*, 55(1):119–139, August 1997.
- 10 Yoav Freund and Robert E. Schapire. Adaptive game playing using multiplicative weights. *Games and Economic Behavior*, 29(1-2):79–103, 1999.
- 11 Nick Gravin, Yuval Peres, and Balasubramanian Sivan. Tight lower bounds for multiplicative weights algorithmic families, 2016. CoRR, abs/1607.02834. URL: <http://arxiv.org/abs/1607.02834>.
- 12 Nick Gravin, Yuval Peres, and Balasubramanian Sivan. Towards optimal algorithms for prediction with expert advice. In *To appear in the Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, 2016.
- 13 András Gyorgy, Dávid Pál, and Csaba Szepesvári. *Online Learning: Algorithms for Big Data*. Manuscript, 2013.
- 14 James Hannan. Approximation to bayes risk in repeated play. *Contributions to the Theory of Games*, 3:97–139, 1957.

- 15 David Haussler, Jyrki Kivinen, and Manfred K. Warmuth. Tight worst-case loss bounds for predicting with expert advice. In *EuroCOLT*, pages 69–83, 1995.
- 16 Elad Hazan, Adam Kalai, Satyen Kale, and Amit Agarwal. Logarithmic regret algorithms for online convex optimization. In *Learning Theory, 19th Annual Conference on Learning Theory, COLT 2006, Pittsburgh, PA, USA, June 22-25, 2006, Proceedings*, pages 499–513, 2006.
- 17 Wouter M. Koolen. The pareto regret frontier. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 863–871, 2013.
- 18 Wouter M. Koolen and Tim van Erven. Second-order quantile methods for experts and combinatorial games. In *Proceedings of The 28th Conference on Learning Theory, COLT 2015, Paris, France, July 3-6, 2015*, pages 1155–1175, 2015.
- 19 Nick Littlestone and Manfred K. Warmuth. The weighted majority algorithm. *Information and Computation*, 108(2):212–261, February 1994.
- 20 Haipeng Luo and Robert E. Schapire. Towards minimax online learning with unknown time horizon. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, pages 226–234, 2014.
- 21 H. Brendan McMahan and Jacob Abernethy. Minimax optimal algorithms for unconstrained linear optimization. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 2724–2732, 2013.
- 22 H. Brendan McMahan and Francesco Orabona. Unconstrained online linear learning in hilbert spaces: Minimax algorithms and normal approximations. In *Proceedings of The 27th Conference on Learning Theory, COLT 2014, Barcelona, Spain, June 13-15, 2014*, pages 1020–1039, 2014.
- 23 Indraneel Mukherjee and Robert E. Schapire. Learning with continuous experts using drifting games. *Theor. Comput. Sci.*, 411(29-30):2670–2683, 2010.
- 24 Alexander Rakhlin, Ohad Shamir, and Karthik Sridharan. Relax and randomize : From value to algorithms. In *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States.*, pages 2150–2158, 2012.
- 25 Alexander Rakhlin, Karthik Sridharan, and Ambuj Tewari. Online learning: Random averages, combinatorial parameters, and learnability. In *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010. Proceedings of a meeting held 6-9 December 2010, Vancouver, British Columbia, Canada.*, pages 1984–1992, 2010.
- 26 Alexander Rakhlin, Karthik Sridharan, and Ambuj Tewari. Online learning: Beyond regret. In *COLT 2011 – The 24th Annual Conference on Learning Theory, June 9-11, 2011, Budapest, Hungary*, pages 559–594, 2011.
- 27 Volodimir G. Vovk. Aggregating strategies. In *Proceedings of the Third Annual Workshop on Computational Learning Theory, COLT’90*, pages 371–386, 1990.
- 28 Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Machine Learning, Proceedings of the Twentieth International Conference (ICML 2003), August 21-24, 2003, Washington, DC, USA*, pages 928–936, 2003.