Online Learning with Stochastically Partitioning Experts

Puranjay Datta 19D070048@IITB.AC.IN
Indian Institute of Technology Bombay, Mumbai, India

Jaya Prakash Champati JAYA.CHAMPATI@IMDEA.ORG
IMDEA Networks Institute, Madrid, Spain

Sharayu Moharir SHARAYUM@EE.IITB.AC.IN
Indian Institute of Technology Bombay, Mumbai, India

Editors: Shipra Agrawal and Aaron Roth

Abstract

We study a variant of the experts problem in which new experts are revealed over time according to a stochastic process. The experts are represented by partitions of a hypercube $\mathbb{B}$ in d-dimensional Euclidean space. In each round, a point is drawn from $\mathbb{B}$ in an independent and identically distributed manner using an unknown distribution. For each chosen point, we draw d orthogonal hyperplanes parallel to the d faces of $\mathbb{B}$ passing through the point. The set of experts available in a round is the set of partitions of $\mathbb{B}$ created by all the hyperplanes drawn up to that point. Losses are adversarial and the performance metrics of interest include expected regret and high probability bounds on the sample-path regret. We propose a suitably adapted version of the Hedge algorithm called Hedge-G, which uses a constant learning rate and has $O(\sqrt{2^d T \ln T})$ expected regret, which is order-optimal. Further, we show that for Hedge-G, there exists a trade-off between choosing a learning rate that has optimal expected regret and a learning rate that leads to a high probability sample-path regret bound. We address this limitation by proposing AdaHedge-G, a variant of Hedge-G that uses an adaptive learning rate by tracking the loss of the experts revealed up to that time. For $\epsilon > 0$, AdaHedge-G simultaneously achieves $O(T^{\frac{\epsilon}{2}}\sqrt{T \ln T})$ sample-path regret, with probability at least $1 - T^{-\epsilon}$, and $O(\ln(\ln T)\sqrt{T \ln T})$ expected regret.

1. Introduction

In the standard decision-theoretic online learning studied by Freund and Schapire (1997), there are N experts (or actions) at the disposal of a learner. In round t, the learner chooses a probability mass function $\boldsymbol{p}_t$ over the set of experts $\{1, 2, \ldots, N\}$, an adversary reveals the loss vector $\boldsymbol{l}_t = (l_t(1), \ldots, l_t(N)) \in [0,1]^N$, and the learner incurs an (expected) loss of $\langle \boldsymbol{p}_r, \boldsymbol{l}_r \rangle$. The total loss incurred by the learner after T rounds is $L_T = \sum_{r=1}^{T} \langle \boldsymbol{p}_r, \boldsymbol{l}_r \rangle$, and the total loss of choosing expert i in all the rounds is $L_T(i) = \sum_{r=1}^{T} l_r(i)$. The learner aims to minimize its cumulative regret up to round T, defined as $L_T - \min_i L_T(i)$.

The celebrated Hedge algorithm by Freund and Schapire (1997) uses a parameter called the learning rate $\eta \geq 0$, assigns weight $w_t(i) = e^{-\eta L_{t-1}(i)}$ for each expert i based on the observed cumulative loss, and chooses expert i with probability $p_i = w_t(i)/W_t$, where $W_t = \sum_{\tau=1}^{t} w_\tau(i)$. For a suitable choice of η, Hedge has $O(\sqrt{T \ln N})$ regret. Subsequent works explored improved algorithmic techniques seeking regret bounds where the dependency on T is replaced by metrics that capture the variability of the sequence of loss vectors $\boldsymbol{l}_t$ Cesa-Bianchi et al. (2007); Hazan and Kale (2010); Chiang et al. (2012). In contrast to these works, Gofer et al. (2013) studied the dependency of the regret bound on the number of experts N. They introduced the *branching experts setting*,

where new experts may be revealed in each round, and the cumulative loss of any new expert is either equal or close to the cumulative loss of one of the existing experts. Gofer et al. (2013) proposed an algorithm that has $O(\sqrt{TN_T})$ regret, where N_T is the number of experts revealed in the first T rounds.

In this paper, we study the stochastically partitioning experts setting, a stochastic variant of the branching experts setting, where the experts revealed in each round are new sub-partitions of a hypercube $\mathbb{B}$ in d-dimensional Euclidean space[1]. In each round t, the environment draws a point i.i.d. from $\mathbb{B}$ using a fixed (unknown) distribution. For each chosen point, we draw d orthogonal hyperplanes parallel to the d faces of $\mathbb{B}$ passing through the point. The set of experts revealed up to round t is the set of partitions of $\mathbb{B}$ created by the intersection of the d orthogonal hyperplanes passing through each of the t points drawn up to that round, resulting in $(t+1)^d$ experts[2]. The partition of experts is illustrated in Fig.1. In each round, the environment only reveals the losses of the existing experts and we allow the losses to be adversarial. We consider the *perfect clone setting* introduced in Gofer et al. (2013), where a new expert is a perfect clone of its parent expert, i.e., the cumulative loss of a new partition is equal to the cumulative loss of its parent partition. Once the new expert is revealed, its cumulative loss evolves independently from its parent expert in the subsequent rounds. We note that, in contrast to the branching experts setting where N_T is bounded and is independent of T, in the partitioning experts setting, the number of experts in round T is $(T+1)^d$.

1.1. Motivation

The above setting with $\mathbb{B} = [0,1]$ arises in an online learning framework recently studied by Moothedath et al. (2024); Beytur et al. (2024) for ML classification applications that use Deep Learning (DL) inference with a reject/offload option. In this framework, in each round t, a data sample (e.g., image) is presented by the environment. The data sample is input to a pre-trained DL model which outputs softmax values corresponding to different classes. The learner computes a confidence metric $x_t \in [0,1]$ using these softmax values[3]. The learner accepts the classification in round t if the confidence metric x_t is above a threshold, which the learner aims to learn. If the learner accepts the classification, it incurs a loss of zero when the classification is correct, and a loss of one otherwise. If the learner rejects or offloads the classification task, it incurs a cost $c \in [0,1]$. In this problem, the experts are the partitions of $\mathbb{B} = [0,1]$ created by the x_t values corresponding to the data samples that arrive over time Moothedath et al. (2024). If a learner chooses a partition in round t, then the classification of the DL model is accepted if x_t is greater than the supremum of the chosen interval, else the classification is rejected and the data sample is offloaded. For this problem, the partitions are illustrated in Fig.1(*a*).

1. The algorithms and the analysis in this work apply to any convex region in the d-dimensional Euclidean space, but for the ease of exposition, we limit $\mathbb{B}$ to hypercube.
2. Since the points are drawn i.i.d. from Euclidean space, the probability of a chosen point lying on one of the d hyperplanes parallel to the faces of $\mathbb{B}$ passing through another point drawn in some other round is zero. Thus, in round t, there will be $(t+1)^d$ experts with probability one.
3. A typical choice for the confidence metric is the maximum softmax value as the data sample is typically classified into the class with the maximum softmax value.

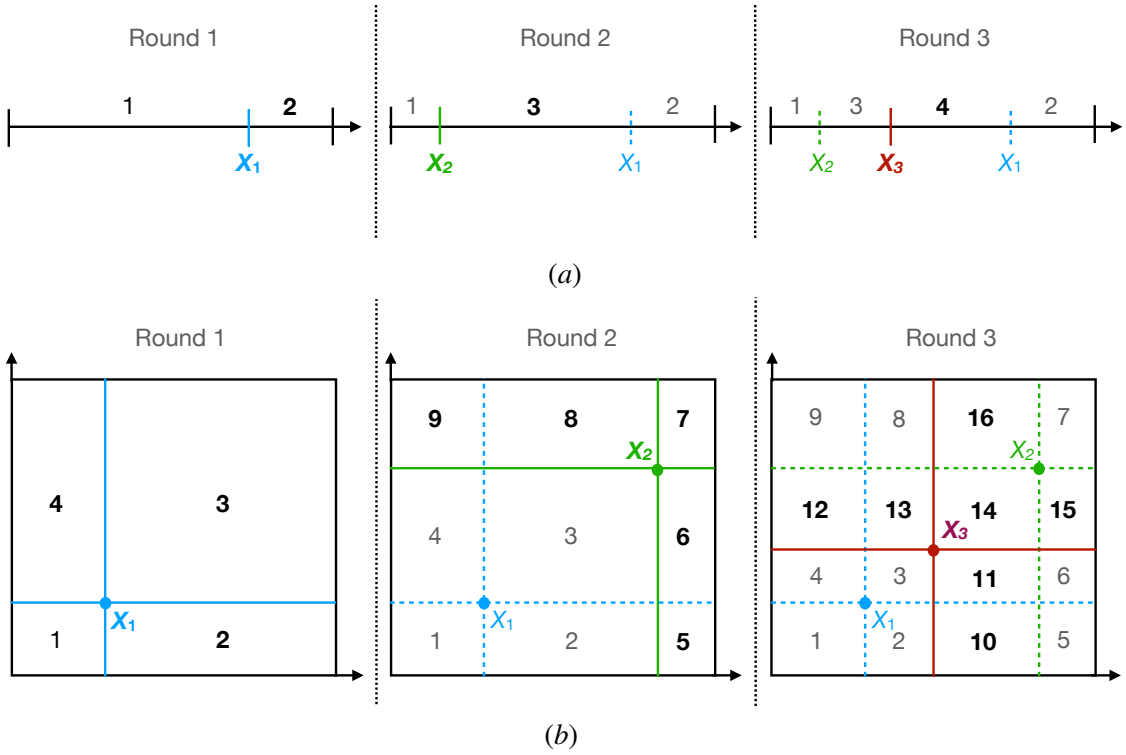

Figure 1: We show the partitioning experts setting for the first three rounds, for one dimension ($d = 1$), on a bounded interval in (a), and for two dimensions ($d = 2$) on a square region in (b). The new point and the new expert indices in each round are highlighted using bold fonts.

1.2. Our Contributions

We study the novel stochastically partitioning experts setting. We propose two algorithms, namely, Hedge-G, a natural extension of the Hedge algorithm for the growing experts setting, and AdaHedge-G, an adaptive learning rate variant of Hedge-G. We prove the following results on the regret of the proposed algorithms.

- Even though the number of experts grow as $(t+1)^d$, we show that Hedge-G has $O(\sqrt{2^d T \ln T})$ expected regret, which is order-optimal in T. Compare this with the Hedge algorithm which has $O(\sqrt{dT \ln T})$ regret in the setting where all the $(T+1)^d$ experts are known apriori.

- We also show that Hedge-G achieves $O(T^{\frac{\epsilon}{2}}\sqrt{T \ln T})$ sample path regret, with probability at least $1 - T^{-\epsilon}$.

- Hedge-G uses a fixed learning rate. We show that there is a trade-off between choosing a rate that gives the optimal expected regret guarantee and a rate that gives a useful sample-path regret guarantee. To address this limitation of Hedge-G, we propose the AdaHedge-G algorithm, a variant of the Hedge-G algorithm that uses an adaptive learning rate. We

show that AdaHedge-G simultaneously achieves $O(\ln(\ln T)\sqrt{T \ln T})$ expected regret, and $O(T^{\frac{\epsilon}{2}}\sqrt{T \ln T})$ sample-path regret, with probability at least $1 - T^{-\epsilon}$, for $\epsilon > 0$.

2. Related Work

The decision-theoretic online learning problem is a variant of the classical prediction with expert advice Littlestone and Warmuth (1994); Vovk (1995) and has received a lot of attention in the past three decades. Variants of this problem, where the set of experts is very large were studied by Chaudhuri et al. (2009); Chernov and Vovk (2010); Luo and Schapire (2015). In Chaudhuri et al. (2009), the authors proposed a parameter-free version of Hedge and showed that it outperforms the classical Hedge algorithm when the set of experts is large. The focus in Chernov and Vovk (2010) was on the setting where multiple experts can be near-clones of each other. For this setting, the authors provided regret guarantees as a function of the effective number of experts, i.e., the number of unique experts available to the learner. The algorithm proposed in Luo and Schapire (2015) is agnostic to the number of experts, and therefore, can be used in a setting where the number of experts is unknown/changing.

As mentioned before, our partitioning experts setting is closely related to the branching experts setting first studied by Gofer et al. (2013). In this work, even though the number of experts increases with time, N_T, the total number of experts revealed after T rounds is assumed to be large but finite. Our setting differs from the branching experts setting as we have an uncountably infinite set of experts from which $(T+1)^d$ experts are revealed in T rounds. Another difference is that the number of new experts revealed in round t is $(t+1)^d - t^d$. The branching experts setting is also the focus in Wu et al. (2021). In addition to the setting in Gofer et al. (2013) where the losses are generated by an adversary, Wu et al. (2021) also considered the setting where the losses are stochastic processes with unknown distributions. The authors proposed a policy that is optimal for both adversarial and stochastic losses.

In Cohen and Mannor (2017), the focus was on the setting where all the experts are known apriori and their losses are revealed in each round, but the number of experts is large, potentially even infinite. The focus here was on identifying a small set of experts such that all other experts are close to any one expert in this small set in terms of their cumulative loss. The authors proposed an algorithm with provable performance guarantees that depend on the ϵ-covering number of the sequence of loss functions. They also proposed a method to compute the optimal ϵ in hindsight.

In Mourtada and Maillard (2017), new experts are revealed over time. The key contribution in this work is two-fold. The authors considered multiple definitions of regret, namely shifting regret and sparse shifting regret to account for the fact that the expert set is growing over time. They designed computationally inexpensive policies with order-optimal regret performance for all the regret definitions considered. The proposed algorithms are anytime and parameter-free. In Gyorfi et al. (1999), the set of experts grew at an exponentially decaying rate and the goal was to make predictions about a stationary ergodic time series. In contrast to this work, in our setting, experts arrive at a much faster rate. In Hazan and Seshadhri (2009); Shalizi et al. (2011), the focus was on the prediction of a non-stationary time series using a growing set of experts.

One of the key parameters of the Hedge algorithm is the learning rate which is typically a function of the time horizon of interest T. It follows that the classical Hedge algorithm is not suitable for settings where the time horizon of interest is unknown. The algorithms proposed in Erven et al. (2011); De Rooij et al. (2014) addressed this limitation by adapting the learning rate

without the need to know the value of T. In contrast, we assume T is given, but adapt the learning rate in AdaHedge-G according to the observed losses so that it simultaneously achieves near-optimal bound for expected regret and non-trivial sample-path regret guarantees.

3. Stochastically Partitioning Experts Setting

In this work, experts are represented by partitions of a hypercube $\mathbb{B}$ in a d-dimensional Euclidean space. As discussed above, in each round t, the environment draws a point i.i.d. from $\mathbb{B}$ using a fixed (unknown) distribution. For each such point, we draw d hyperplanes passing through the point, parallel to the d faces of $\mathbb{B}$. The set of experts available in round t is the set of partitions of $\mathbb{B}$ created by all the hyperplanes drawn up to that round. The partitioning process for $d = 1$ and $d = 2$ is illustrated in Fig.1.

In round 1, the environment draws a point $X_1 \in \mathbb{B}$ creating 2^d experts, which we index $1, \ldots, 2^d$. Similarly, in round t, the environment samples point $X_t \in \mathbb{B}$ resulting in $(t+1)^d$ experts. Among these experts, $(t+1)^d - t^d$ are new experts. We say an expert is a child of a parent expert if the former is a sub-partition of the latter expert. We assign the index of each parent expert to one of its children and assign new indices $t^d + 1, \ldots, (t+1)^d$ to the remaining unindexed new experts. We use $\mathcal{B}_t = \{1, \ldots, n_t\}$, where $n_t = (t+1)^d$, to denote the set of indices at the end of round t.

In round t, the environment first samples X_t, and the learner chooses a probability mass function $\boldsymbol{p}_t$ over the set of experts $\mathcal{B}_t$. Following this, the environment adversarially chooses a loss vector $\boldsymbol{l}_t = (l_t(1), \ldots, l_t(n_t)) \in [0,1]^{n_t}$. The learner therefore incurs an expected loss of $\langle \boldsymbol{p}_r, \boldsymbol{l}_r \rangle$. The cumulative loss of expert $i \in \mathcal{B}_t$ up to time t is $L_t(i) = \sum_{r=1}^{t} l_r(i)$, and the expected cumulative loss of the learner up to time t is

$$L_t = \sum_{r=1}^{t} \langle \boldsymbol{p}_r, \boldsymbol{l}_r \rangle.$$

For each new expert $i \in \mathcal{B}_t \backslash \mathcal{B}_{t-1}$, its cumulative loss up to time t, i.e., $L_{t-1}(i)$ is equal to the cumulative loss of its parent expert from $\mathcal{B}_{t-1}$. The subsequent losses of the new experts, however, evolve independently from those of their parent experts.

We define

$$L_t^* = \min_{i \in \mathcal{B}_t} L_t(i).$$

Given the time horizon T, we aim to minimize the *expected regret*

$$R_T = \mathbb{E}[L_T - L_T^*],$$

where the expectation is with respect to the joint distribution of the sequence of points $\mathbf{X}_T = \{X_1, \ldots, X_T\}$ drawn by the environment in T rounds. Note that $\mathbb{E}[L_t^*]$ will be equal to L_t^* if the loss vectors generated are independent of the points sampled by the environment and the bound we prove will still hold.

We also study the *sample-path regret*

$$\hat{R}_T = L_T - L_T^*,$$

and provide bounds in the high probability regime.

Algorithm 1 Hedge-G for partitioning experts

1: **Initialize:** $\mathcal{B}_0 = \{1\}$, $n_0 = 0$, $w_1 = 1$, and $W_1 = 1$.
2: **for** each round $t = 1, 2, \ldots, T$ **do**
3: X_t is drawn i.i.d. from $\mathbb{B}$ and new partitions are revealed
4: $n_t = (t+1)^d$ and $\mathcal{B}_t = \mathcal{B}_{t-1} \cup \{n_{t-1}+1, \ldots, n_t\}$
5: For $i \in \mathcal{B}_t \backslash \mathcal{B}_{t-1}$, given $L_{t-1}(i)$, compute new weights $w_t(i) = e^{-\eta L_{t-1}(i)}$
6: $\hat{W}_t = W_t + \sum_{i \in \mathcal{B}_t \backslash \mathcal{B}_{t-1}} w_t(i)$
7: Compute $p_{i,t} = \frac{w_{i,t}}{\hat{W}_t}$, for all $i \in \mathcal{B}_t$.
8: Choose an expert using $\boldsymbol{p}_t$, observe $\boldsymbol{l}_t$, and incur the loss $\langle \boldsymbol{p}_t, \boldsymbol{l}_t \rangle$.
9: Update the weights $w_{t+1}(i) = e^{-\eta l_i(t)} w_t(i)$, for all $i \in \mathcal{B}_t$.
10: Cumulative weight $W_{t+1} = \sum_{i=1}^{n_t} w_t(i)$.
11: **end for**

Remark 1: One can alternatively interpret the partitioning experts setting as follows. Instead of treating each partition as an expert, consider that each point in $\mathbb{B}$ is an expert. When the environment draws an expert, it only reveals a single loss value per partition instead of losses for all the points in $\mathbb{B}$. For example, this loss value may be the average loss over all experts in the partition. Since we can only work with the loss values per partition instead of losses of the individual experts (points), the setting where we carry over cumulative losses of the parent partition to the new sub-partitions is well-motivated, especially given its applicability in the classification application discussed in Section 1.1.

4. The Hedge-G Algorithm: Regret Analysis

We propose an algorithm called Hedge-G, a natural extension of the Hedge algorithm for the growing experts setting, that introduces a new weight whenever a new expert arrives. For the branching experts setting, these new weights can be readily computed as the cumulative losses of the new experts are the same as their parent experts. In Algorithm 1, we present Hedge-G adapted to the partitioning experts setting.

The regret analysis for Hedge-G differs from Hedge in that the introduction of new weights in line 5 of Algorithm 1 implies that W_t does not normalize the weights $w_t(i)$, for $i \in \mathcal{B}_t$. A key step in our analysis of Hedge-G is to compute the expected value of the quantity Y_t, the ratio between the sum of new weights $w_t(i)$ and W_t, given by

$$Y_t = \frac{\sum_{i=n_{t-1}+1}^{n_t} w_t(i)}{W_t} = \frac{\sum_{i=n_{t-1}+1}^{n_t} e^{-\eta L_{t-1}(i)}}{\sum_{j \in \mathcal{B}_{t-1}} e^{-\eta L_{t-1}(j)}}. \tag{1}$$

The following theorem characterizes an upper bound on the cumulative loss of Hedge-G.

Theorem 1 *An upper bound for the cumulative loss of Hedge-G is given by*

$$L_T \leq L_T^* + \frac{T\eta}{8} + \frac{\sum_{t=1}^{T} Y_t}{\eta}. \tag{2}$$

Proof We write

$$\ln \frac{W_{t+1}}{W_t} = \ln \frac{W_{t+1}}{\hat{W}_t} + \ln \frac{\hat{W}_t}{W_t}. \tag{3}$$

Given $\hat{W}_t = \sum_{i \in \mathcal{B}_t} e^{-\eta L_{t-1}(i)}$, we upper bound the second term of (3) as follows.

$$\begin{aligned} \ln \frac{\hat{W}_t}{W_t} &= \ln \left(\frac{\sum_{i \in \mathcal{B}_{t-1}} e^{-\eta L_{t-1}(i)} + \sum_{i=n_{t-1}+1}^{n_t} e^{-\eta L_{t-1}(i)}}{\sum_{i \in \mathcal{B}_{t-1}} e^{-\eta L_{t-1}(i)}} \right) \\ &= \ln (1 + Y_t) \leq Y_t. \end{aligned} \tag{4}$$

Next, we upper and lower bound $\ln \frac{W_T}{W_1}$. By definition,

$$\begin{aligned} \ln \frac{W_T}{W_1} = \ln \left(\prod_{t=1}^{T-1} \frac{W_{t+1}}{W_t} \right) &= \sum_{t=1}^{T-1} \left[\ln \frac{W_{t+1}}{\hat{W}_t} + \ln \frac{\hat{W}_t}{W_t} \right] \\ &\leq \sum_{t=1}^{T} \left[-\eta \langle \boldsymbol{p}_t, \boldsymbol{l}_t \rangle + \frac{\eta^2}{8} + Y_t \right] \\ &= -\eta L_T + \frac{\eta^2 T}{8} + \sum_{t=1}^{T} Y_t. \end{aligned} \tag{5}$$

In the third step above, we have used (4) and Hoeffding's lemma to upper bound $\ln \frac{W_{t+1}}{\hat{W}_t}$. Also,

$$\begin{aligned} \ln \frac{W_T}{W_1} = \ln \sum_{i=1}^{n_T} e^{-\eta L_T(i)} &\geq \ln \max_{i \in \mathcal{B}_T} e^{-\eta L_T(i)} \\ &\geq \max_{i \in \mathcal{B}_t} \ln e^{-\eta L_T(i)} = -\eta L_T^*. \end{aligned} \tag{6}$$

From (5) and (6), we obtain the result. ■

To obtain a bound on the expected regret of Hedge-G from Theorem 1, we need to compute $\sum_{t=1}^{T} \mathbb{E}[Y_t]$. A primer for computing $\mathbb{E}[Y_t]$ is the following lemma which states that in any slot the new point is equally likely to belong to any one of the existing partitions of $\mathbb{B}$.

Lemma 2 *Given that the sequence of points $\{X_t\}$ are drawn i.i.d. from $\mathbb{B}$, the point X_t drawn in round t is equally likely to belong to any one of the existing t^d partitions, i.e.,*

$$\mathbb{P}(X_t \in \textit{ partition } i) = \frac{1}{t^d}, \ \forall i \in \mathcal{B}_{t-1}.$$

Our next result uses Lemma 2 to compute $\mathbb{E}[Y_t]$.

Lemma 3 $\mathbb{E}[Y_t] = \left(1 + \frac{1}{t}\right)^d - 1 \leq \frac{2^d}{t}$.

The proofs of Lemmas 2 and 3 are given in the Appendix.

Taking expectation on both sides in (2) (Theorem 1) and using Lemma 3, we obtain the following bound on expected regret:

$$R_T \leq \frac{\eta T}{8} + \frac{2^d}{\eta} \sum_{t=1}^{T} \frac{1}{t} \leq \frac{\eta T}{8} + \frac{2^d (\ln T + 1)}{\eta}. \tag{7}$$

The regret bound in the following corollary immediately follows from (7).

Corollary 4 *For the partitioning experts setting, for Hedge-G with* $\eta = \sqrt{2^{d+3}(\ln T + 1)/T}$*, the expected regret* $R_T = O(\sqrt{2^d T \ln T})$.

Lower bound: Under our model, for any realization of $\mathbf{X}_T$, there will be $(T+1)^d$ experts at the end of round T. Since the environment generates losses adversarially, the sample path regret $\hat{R}_T$ for any algorithm is $\Omega(\sqrt{dT \ln T})$ Freund and Schapire (1999). Since this lower bound is valid for any realization, the expected regret of any algorithm is also $\Omega(\sqrt{dT \ln T})$. Thus, from Corollary 4, we see that Hedge-G is order-optimal expected regret with respect to the time-horizon T. Note that, the vanilla Hedge algorithm achieves $O(\sqrt{dT \ln T})$ expected regret only when all the $(T+1)^d$ experts are known apriori and their losses are revealed in each round.

Corollary 5 *For the partitioning experts setting, Hedge-G with* $\eta = \sqrt{\frac{2^{d+3}(\ln T+1)}{T^{1-\epsilon}}}$ *for any* $\epsilon > 0$*, the sample-path regret* $\hat{R}_T = O(\sqrt{2^d T^{1+\epsilon} \ln T})$ *with probability at least* $1 - T^{-\epsilon}$.

Proof Using Markov inequality for the summation of the random variables Y_t, we get

$$\mathbb{P}\left(\sum_{t=1}^{T} Y_t \geq T^{\epsilon} \sum_{t=1}^{T} \mathbb{E}[Y_t]\right) \leq 1 - \frac{\sum_{t=1}^{T} \mathbb{E}[Y_t]}{T^{\epsilon} \sum_{t=1}^{T} \mathbb{E}[Y_t]} = 1 - T^{-\epsilon}.$$

Using this result in (2) and the upper bound for $\mathbb{E}[Y_t]$ from Lemma 2, we obtain, with probability at least $1 - T^{-\epsilon}$,

$$\hat{R}_T \leq \frac{\eta T}{8} + \frac{2^d T^{\epsilon}(\ln T + 1)}{\eta}.$$

Choosing $\eta = \sqrt{\frac{2^{d+3}(\ln T+1)}{T^{1-\epsilon}}}$ results in $\hat{R}_T \leq \sqrt{2^{d-1} T^{1+\epsilon}(\ln T + 1)}$. ■

From Corollaries 4 and 5, it follows that for $\eta = \sqrt{\frac{2^{d+3}(\ln T+1)}{T^{1-\epsilon}}}$, the sample-path regret of Hedge-G, $\hat{R}_T = O(\sqrt{T^{1+\epsilon} \ln T})$ with high probability and the expected regret of Hedge-G $R_T = O(T^{\frac{\epsilon}{2}} \sqrt{T \ln T})$. Compared to this, the expected regret for Hedge-G with $\eta = \sqrt{2^{d+3}(\ln T + 1)/T}$ is $O(\sqrt{T \ln T})$, but this value of η leads to a sample-path regret bound that holds with probability zero, as $\epsilon = 0$. Therefore, to obtain a high probability bound on sample-path regret of Hedge-G using Theorem 3 and Markov's inequality, we use a value of η for which the expected regret is higher than the optimal by a factor of $O(T^{\frac{\epsilon}{2}})$. In Section 5, we address this trade-off by adapting the learning rate based on the losses revealed by the adversary.

Remark 2: We now show that if X_t are drawn adversarially from $\mathbb{B}$, Hedge-G has linear regret. We construct the following problem instance for $d = 1$. Let the adversary always split the best expert resulting in two best experts j and k. Assign $l_t(i) = 1,\ \forall i \neq j,\ k$. Uniformly at random, the adversary assigns a loss of one to one expert in the set $\{j, k\}$ and zero to the other expert. For this problem instance, at any time t, $L_t^* = 0$, but the expected loss for Hedge-G in that time step will be at least $\frac{1}{2}$. Hence, Hedge-G has expected regret of at least $\frac{T}{2}$. This result is expected because if X_t are adversarially drawn from $\mathbb{B}$, then the partitioning expert setting is a special case of the branching experts setting studied by Gofer et al. (2013). It is known for the branching experts setting, the regret of any algorithm is $\Omega(\sqrt{TN_T})$, where N_T for the partitioning expert setting is equal to $(T+1)^d$.

5. AdaHedge-G: Hedge-G with Adaptive Learning Rate

In this section, we propose a variant of Hedge-G called AdaHedge-G and show that its expected regret is near-optimal while simultaneously achieving the same high probability bound for the sample-path regret stated in Corollary 5.

The details of AdaHedge-G are presented in Algorithm 2. The key idea behind the algorithm is to track the summation of Y_ts using the variable S and suitably change the learning rate over rounds using a doubling trick. In particular, we partition the time into segments, where *segment* i spans the number of rounds for which $S \leq 2^{id}$. At the start of any segment i, we reset the value of S to zero, choose an equal weight for all the existing experts (from the previous segment), and use Hedge-G with learning rate $\eta = \sqrt{8(2^{id} + \ln \tau_i)/T}$, where τ_i is the round in which the segment starts.

Algorithm 2 AdaHedge-G

Initialize: $r \leftarrow 0, S \leftarrow 0, \tau \leftarrow 1, c \leftarrow 2^d, \mathbf{w}_1 = 1$, and $\eta \leftarrow \sqrt{\frac{8c}{T}}$.
for $t = 1, \ldots, T$ **do**
  X_t is drawn i.i.d. from $\mathbb{B}$
  Calculate Y_t using (1)
  if $S + Y_t > c$ **then**
    Start a new segment
    $\mathbf{w}_t = (w_1, \ldots, w_{t^d}) = \left(\frac{1}{t^d}, \ldots, \frac{1}{t^d}\right)$
    $S \leftarrow 0$
    $c \leftarrow 2^d c$
    $\eta \leftarrow \sqrt{\frac{8(c + d \ln t)}{T}}$
  end if
  $S \leftarrow S + Y_t$
  Use Hedge-G with already observed X_t, initial weight vector $\mathbf{w}_t$ and learning rate η.
end for

The next theorem characterizes an upper bound on the cumulative loss of AdaHedge-G.

Theorem 6 *An upper bound for the cumulative loss of AdaHedge-G is given by*

$$L_T \leq L_T^* + \frac{2^{d-\frac{1}{2}}}{2^{\frac{d}{2}} - 1} \sqrt{T \left(\sum_{t=1}^{T} Y_t + 1 \right)} + \left(1 + \frac{2}{d} \log_2 \left(\sqrt{\sum_{t=1}^{T} Y_t + 1} \right) \right) \sqrt{dT \ln T / 2}. \quad (8)$$

Proof Let r_i be the length of the i^{th} segment, i.e., the number of rounds in the i^{th} segment. By definition of a segment, we have

$$r_i = \min \left\{ r : \sum_{i=\tau_i}^{r} Y_i > 2^{id} \right\} - \tau_i,$$

where τ_i is the round in which the segment i starts and is given by $\tau_i = \sum_{u=1}^{i-1} r_u + 1$. Let $R^{(i)}$ denote the regret incurred in segment i. It follows that

$$R^{(i)} = \sum_{u=\tau_i}^{\tau_{i+1}-1} l_u - \min_{j \in \mathcal{B}_{\tau_{i+1}-1}} \sum_{u=\tau_i}^{\tau_{i+1}-1} l_u(j).$$

We repeat the regret analysis from the proof of Theorem 1 for $R^{(i)}$ and obtain

$$R^{(i)} \leq \frac{\eta_i r_i}{8} + \frac{S_i + d \ln \tau_i}{\eta_i} \leq \sqrt{T(2^{id} + d \ln T)/2},$$

where, we have used $r_i \leq T$, $\tau_i \leq T$,

$$S_i = \sum_{r=\tau_i}^{\tau_{i+1}-1} Y_r \leq 2^{id}, \text{ and } \eta_i = \sqrt{\frac{2^{id+3} + 8d \ln \tau_i}{T}}.$$

Note the weights are reinitialized to $1/\tau_i^d$ at the start of the segment and this yields the additional term of $d \ln \tau_i$ when upper bounding $\ln \frac{W_{\tau_{i+1}-1}}{W_{\tau_i}}$ in the analysis leading to (6).

Let m denote the last segment that started before round T. We add regret across all the m segments and obtain,

$$\begin{aligned} L_T - L_T^* \leq \sum_{i=1}^{m} R^{(i)} &\leq \sqrt{\frac{T}{2}} \left(\sqrt{2^d + d \ln T} + \sqrt{2^{2d} + d \ln T} + \ldots + \sqrt{2^{md} + d \ln T} \right) \\ &\leq \sqrt{\frac{T}{2}} \sum_{i=1}^{m} 2^{\frac{id}{2}} + m \sqrt{dT \ln T/2} \\ &\leq \frac{\sqrt{\frac{T}{2}} 2^{\frac{(m+1)d}{2}}}{2^{\frac{d}{2}} - 1} + m \sqrt{dT \ln T/2}. \end{aligned} \tag{9}$$

Further, we have

$$\sum_{i=1}^{T} Y_t \geq \sum_{i=1}^{m-1} 2^{id} \geq 2^d \frac{2^{(m-1)d} - 1}{2^d - 1}.$$

Therefore,

$$2^{\frac{md}{2}} \leq 2^{\frac{d}{2}} \sqrt{\sum_{t=1}^{T} Y_t + 1} \tag{10}$$

$$\implies m \leq \frac{2}{d} \log_2 \left(2^{\frac{d}{2}} \sqrt{\sum_{t=1}^{T} Y_t + 1} \right) = 1 + \frac{2}{d} \log_2 \left(\sqrt{\sum_{t=1}^{T} Y_t + 1} \right). \tag{11}$$

Substituting (10) and (11) in (9), we obtain the result ■

The next theorem provides guarantees on the regret of AdaHedge-G.

Theorem 7 *For the partitioning experts setting AdaHedge-G has the following regret bounds.*

(i) The expected regret $R_T = O(\ln(\ln T)\sqrt{T \ln T})$.

(ii) For $\epsilon > 0$, *and* $d \geq 1$, *the sample-path regret* $\hat{R}_T = O(T^{\frac{\epsilon}{2}} \sqrt{T \ln T})$, *with probability at least* $1 - T^{-\epsilon}$.

(iii) For $d = 1$, the sample-path regret $\hat{R}_T = O(\ln(\ln T)\sqrt{T \ln T})$, with probability at least $1 - (eT)^{-0.25}$.

Proof

(i) From Lemma 3 and applying Jensen's inequality to (8) and substituting $\mathbb{E}\left[\sum_{t=1}^{T} Y_t\right] \leq \ln T + 1$, we get the final result.

(ii) Using Markov inequality for the summation of the random variables Y_t, we get

$$\mathbb{P}\left(\sum_{t=1}^{T} Y_t \geq T^{\epsilon} \sum_{t=1}^{T} \mathbb{E}[Y_t]\right) \leq 1 - \frac{\sum_{t=1}^{T} \mathbb{E}[Y_t]}{T^{\epsilon} \sum_{t=1}^{T} \mathbb{E}[Y_t]} = 1 - T^{-\epsilon}.$$

Using this result in (8) and the upper bound for $\mathbb{E}[Y_t]$ from Lemma 2, we obtain, with probability at least $1 - T^{-\epsilon}$,

$$\hat{R}_T = O(\sqrt{T^{1+\epsilon} \ln T} + (1+\epsilon) \ln T \sqrt{T \ln T}) = O(\sqrt{T^{1+\epsilon} \ln T}).$$

(iii) For $d = 1$, we have

$$Y_t = \frac{e^{-L_{t-1}(n_t)}}{\sum_{j \in \mathcal{B}_{t-1}} e^{-L_{t-1}(j)}}. \tag{12}$$

Note that $e^{L_{t-1}(i) - L_{t-1}(j)} \geq 0$ for all i, j, and $L_{t-1}(i) \geq L_{t-1}(j)$ implies $e^{L_{t-1}(i) - L_{t-1}(j)} \geq 1$. Therefore,

$$\begin{aligned} Y_t &= \frac{1}{\sum_{j: L_{t-1}(j) > L_{t-1}(n_t)} e^{L_{t-1}(n_t) - L_{t-1}(j)} + \sum_{j: L_{t-1}(j) \leq L_{t-1}(n_t)} e^{L_{t-1}(n_t) - L_{t-1}(j)}} \\ &\leq \frac{1}{\sum_{j \in \mathcal{B}_{t-1}} \mathbb{1}_{\{L_{t-1}(j) \leq L_{t-1}(n_t)\}}}. \end{aligned} \tag{13}$$

In round t, we define a random variable Z_t such that $Z_t = j^{-1}$, if X_t falls in the j^{th} best expert, i.e., if $\sum_{j \in \mathcal{B}_{t-1}} \mathbb{1}_{\{L_t(j) \leq L_t(n_t)\}} = j$. From (13), we have $Y_t \leq Z_t$, for all t. From Lemma 2, the probability that X_t falls in j^{th} is $\frac{1}{t}$, which implies $\mathbb{P}(Z_t = j^{-1}) = 1/t$. Therefore,

$$\mathbb{E}[Z_t] = \sum_{j=1}^{t} \frac{1}{t} \frac{1}{j} \leq \frac{\ln t + 1}{t}, \tag{14}$$

$$\implies \sum_{t=1}^{T} \mathbb{E}[Z_t] \leq (\ln T + 1)^2. \tag{15}$$

Further, we have

$$\mathbb{P}\left(\sum_{t=1}^{T} Y_t - \sum_{t=1}^{T} \mathbb{E}[Y_t] > \delta\right) \leq \mathbb{P}\left(\sum_{t=1}^{T} Z_t - \sum_{t=1}^{T} \mathbb{E}[Y_t] > \delta\right)$$

$$\leq \mathbb{P}\left(\sum_{t=1}^{T} Z_t - \sum_{t=1}^{T}\mathbb{E}[Z_t] > \delta - \sum_{t=1}^{T}\mathbb{E}[Z_t] + \sum_{t=1}^{T}\mathbb{E}\left[Y_t\right]\right)$$
$$\leq \mathbb{P}\left(\sum_{t=1}^{T} Z_t - \sum_{t=1}^{T}\mathbb{E}[Z_t] > \delta - (\ln T + 1)^2 + \ln T\right). \quad (16)$$

To get (16), we use (14), (15), and the fact that $\sum_{t=1}^{T}\mathbb{E}\left[Y_t\right] \geq \ln T$. Since the Z_ts are independent and are upper bounded by one, using Bernstein's inequality, we get

$$\mathbb{P}\left(\sum_{t=1}^{T} Z_t - \sum_{t=1}^{T}\mathbb{E}[Z_t] > \delta'\right) \leq e^{-\frac{\delta'^2/2}{V_n + \delta'/3}}, \quad (17)$$

where $V_n = \sum_{t=1}^{T}\text{Var}(Z_t)$, and $\delta' = \delta - (\ln T + 1)^2 + \ln T$. We have

$$\text{Var}(Z_t) = \sum_{j=1}^{t}\frac{1}{t}\frac{1}{j^2} - \mathbb{E}[Z_t]^2 \leq \frac{\pi^2}{6t} \implies \sum_{t=1}^{T}\text{Var}(Z_t) \leq \frac{\pi^2}{6}(\ln T + 1). \quad (18)$$

Choosing $\delta = (\ln T + 1)^2 + 1$ results in $\delta' = \ln T + 1$. Substituting δ' and (18) in (17),

$$\mathbb{P}\left(\sum_{i=1}^{T} Y_i - \sum_{i=1}^{T}\mathbb{E}[Y_i] > \delta\right) \leq e^{-\frac{(\ln T+1)^2/2}{\frac{\pi^2}{6}(\ln T+1)+(\ln T+1)/3}} \leq e^{\frac{-3\ln(eT)}{\pi^2+2}} \leq (eT)^{-0.25}. \quad (19)$$

Substituting the above result in (8) proves the final result.

■

From parts (i) and (ii) of Theorem 7, we observe that AdaHedge-G has near-optimal expected regret (sub-optimality of a factor of $\ln(\ln T)$) and it also has the same high probability bound on sample-path regret as that of Hedge-G in Corollary 5. AdaHedge-G thus addresses the limitation of Hedge-G discussed at the end of Section 4. Further, in part (iii) of the theorem, for the special case $d = 1$, we provide a sample-path regret that is near-optimal with high probability, independent of ϵ. Proving a tighter bound for $d > 1$, similar to the case $d = 1$, remains an open problem.

6. Concluding Remarks

In this work, we propose an adaptation of Hedge for the partitioning experts setting where the number of experts increases polynomially with time. We show that our algorithm and its adaptive rate variant have (near-)optimal expected regret bounds and non-trivial sample path-regret bounds under the high probability regime.

Possible extensions of this work include: *(i)* designing anytime policies when T is unknown, *(ii)* considering the setting where the rate of growth of the experts is random, i.e., the environment samples a random number of points in each round, and *(iii)* studying the setting where the new experts are approximate clones of the parent experts instead of being perfect clones. Further, the setting of stochastically partitioning experts with stochastic losses can also be explored.

References

Hasan Burhan Beytur, Ahmet Gunhan Aydin, Gustavo de Veciana, and Haris Vikalo. Optimization of offloading policies for accuracy-delay tradeoffs in hierarchical inference. In *IEEE INFOCOM (to appear)*, 2024.

Nicolo Cesa-Bianchi, Yishay Mansour, and Gilles Stoltz. Improved second-order bounds for prediction with expert advice. *Machine Learning*, 66:321–352, 2007.

Kamalika Chaudhuri, Yoav Freund, and Daniel J Hsu. A parameter-free hedging algorithm. *Advances in neural information processing systems*, 22, 2009.

Alexey Chernov and Vladimir Vovk. Prediction with advice of unknown number of experts. *arXiv preprint arXiv:1006.0475*, 2010.

Chao-Kai Chiang, Tianbao Yang, Chia-Jung Lee, Mehrdad Mahdavi, Chi-Jen Lu, Rong Jin, and Shenghuo Zhu. Online optimization with gradual variations. In *Conference on Learning Theory*, pages 6–1. JMLR Workshop and Conference Proceedings, 2012.

Alon Cohen and Shie Mannor. Online learning with many experts. *CoRR*, abs/1702.07870, 2017.

Steven De Rooij, Tim Van Erven, Peter D Grünwald, and Wouter M Koolen. Follow the leader if you can, hedge if you must. *The Journal of Machine Learning Research*, 15(1):1281–1316, 2014.

Tim Erven, Wouter M Koolen, Steven Rooij, and Peter Grünwald. Adaptive hedge. *Advances in Neural Information Processing Systems*, 24, 2011.

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, 1997.

Yoav Freund and Robert E Schapire. Adaptive game playing using multiplicative weights. *Games and Economic Behavior*, 29(1-2):79–103, 1999.

Eyal Gofer, Nicolò Cesa-Bianchi, Claudio Gentile, and Yishay Mansour. Regret minimization for branching experts. In *Proceedings of the 26th Annual Conference on Learning Theory*, volume 30 of *Proceedings of Machine Learning Research*, pages 618–638. PMLR, 12–14 Jun 2013.

L Gyorfi, Gábor Lugosi, and Gusztáv Morvai. A simple randomized algorithm for sequential prediction of ergodic time series. *IEEE Transactions on Information Theory*, 45(7):2642–2650, 1999.

Elad Hazan and Satyen Kale. Extracting certainty from uncertainty: Regret bounded by variation in costs. *Machine learning*, 80:165–188, 2010.

Elad Hazan and Comandur Seshadhri. Efficient learning algorithms for changing environments. In *Proceedings of the 26th annual international conference on machine learning*, pages 393–400, 2009.

Nick Littlestone and Manfred K Warmuth. The weighted majority algorithm. *Information and computation*, 108(2):212–261, 1994.

Haipeng Luo and Robert E Schapire. Achieving all with no parameters: Adanormalhedge. In *Conference on Learning Theory*, pages 1286–1304. PMLR, 2015.

Vishnu Narayanan Moothedath, Jaya Prakash Champati, and James Gross. Getting the best out of both worlds: Algorithms for hierarchical inference at the edge. *(accepted to) IEEE Transactions on Machine Learning in Communications and Networking*, 2024. URL https://arxiv.org/abs/2304.00891.

Jaouad Mourtada and Odalric-Ambrym Maillard. Efficient tracking of a growing number of experts. In *International Conference on Algorithmic Learning Theory*, pages 517–539. PMLR, 2017.

Cosma Rohilla Shalizi, Abigail Z Jacobs, Kristina Lisa Klinkner, and Aaron Clauset. Adapting to non-stationarity with growing expert ensembles. *arXiv preprint arXiv:1103.0949*, 2011.

Vladimir G Vovk. A game of prediction with expert advice. In *Proceedings of the eighth annual conference on Computational learning theory*, pages 51–60, 1995.

Yi-Shan Wu, Yi-Te Hong, and Chi-Jen Lu. Lifelong learning with branching experts. In *Asian Conference on Machine Learning*, pages 1161–1175. PMLR, 2021.

Appendix A. Proof of Lemma 2

We first prove the result for one dimension and then extend the result for the d dimension by establishing the independence of the coordinates of the points in each dimension.

For $d = 1$, the points belong to a closed interval on the real line. We have a strict inequality since X_t is drawn from a continuous i.i.d. distribution. Hence for any two permutations $X_{j_1}, X_{j_2}, \ldots, X_{j_t}$ and $X_{k_1}, X_{k_2}, \ldots, X_{k_t}$ of the sequence $\mathbf{X}_t$, we have

$$\mathbb{P}(X_{j_1} < X_{j_2} < \ldots < X_{j_t}) = \mathbb{P}(X_{k_1} < X_{k_2} < \cdots < X_{k_t}).$$

Since the events $\{X_{j_1} < X_{j_2} < \cdots < X_{j_t}\}$ are mutually exclusive and there are $t!$ possible permutations, we have

$$\sum_{\{j_1, j_2, \ldots, j_n\}} \mathbb{P}\left(X_{j_1} < X_{j_2} < \cdots < X_{j_t}\right) = 1 \tag{20}$$

$$\implies \mathbb{P}\left(X_{j_1} < X_{j_2} < \cdots < X_{j_t}\right) = \frac{1}{t!}. \tag{21}$$

Given any realization of the sequence $\mathbf{X}_{t-1}$, for some permutation of $j_1, j_2, \ldots j_{t-1}$ we have $X_{j_1} < X_{j_2} < \cdots < X_{j_{t-1}}$. Let expert i be the kth interval $(X_{j_{k-1}}, X_{j_k})$, then the event $\{X_t \in \text{expert } i\}$ is equivalent to $\{X_t \in (X_{j_{k-1}}, X_{j_k})\}$, i.e., X_t is the kth highest value in the realization $\{\mathbf{X}_{t-1}, X_t\}$. Therefore, we have

$$\begin{aligned}
&\mathbb{P}\left(X_t \in \text{ expert } i \,\middle|\, X_{j_1} < X_{j_2} < \cdots < X_{j_{t-1}}\right) \\
=&\mathbb{P}\left(X_t \in (X_{j_{k-1}}, X_{j_k}) \,\middle|\, X_{j_1} < X_{j_2} < \cdots < X_{j_{t-1}}\right) \\
=&\frac{P\left(X_{j_1} < \cdots < X_{j_{k-1}} < X_t < X_{j_k} < \cdots < X_{j_{t-1}}\right)}{P\left(X_{j_1} < X_{j_2} < \cdots < X_{j_{t-1}}\right)} \\
=&\frac{\frac{1}{t!}}{\frac{1}{(t-1)!}} = \frac{1}{t}.
\end{aligned} \tag{22}$$

Note that the conditional probability is independent of k and thus it is true for any expert i. Finally, using total probability law over the permutations $j_1, j_2, \ldots, j_{t-1}$, we obtain $\mathbb{P}(X_t \in \text{ expert } i) = 1/t$, for all i.

For $d > 1$, let $X_t = (Z_t^1, \ldots, Z_t^d)$, where Z_t^r is the Euclidean coordinate of point X_t in r^{th} dimension.

Claim: Z_t^r are i.i.d. across t and r.

From the above claim and from (20), for any permutation $j_1, j_2, \ldots, j_{t-1}$ in dimension k, we obtain

$$\begin{aligned}
&\mathbb{P}\left(Z_{j_1}^r < Z_{j_2}^r < \cdots < Z_{j_t}^r\right) = \frac{1}{t!} \\
\implies &\mathbb{P}\left(\{Z_{m_1}^1 < Z_{m_2}^1 < \cdots < Z_{m_t}^1\}, \ldots, \{Z_{j_1}^d < Z_{j_2}^d < \cdots < Z_{j_t}^d\}\right) = \frac{1}{(t!)^d}.
\end{aligned}$$

Again, given any realization of $\mathbf{X}_{t-1}$, the event $\{X_t \in \text{ expert } i\}$ is equivalent to $\{Z_t^r \in (Z_{j_{k-1}}^r, z_{j_k}^r)\}$ for some permutation $j_1^r, j_2^r, \ldots, j_{t-1}^r$ in each dimension r.

$$\mathbb{P}\left(X_t \in \text{ expert } i \,\middle|\, \{Z_{m_1}^1 < Z_{m_2}^1 < \cdots < Z_{m_{t-1}}^1\}, \ldots, \{Z_{j_1}^d < Z_{j_2}^d < \cdots < Z_{j_{t-1}}^d\}\right) \tag{23}$$
$$= \frac{P\left(\{Z_{m_1}^1 < \cdots < Z_{m_{k-1}}^1 < Z_t^1 < Z_{m_k}^1 < \cdots < Z_{j_{t-1}}^1\}, \ldots \{Z_{j_1}^d < \cdots < Z_{j_{k-1}}^d < Z_t^d < Z_{j_k}^d < \cdots < Z_{j_{t-1}}^d\}\right)}{P\left(\{Z_{m_1}^1 < Z_{m_2}^1 < \cdots < Z_{m_t}^1\}, \ldots, \{Z_{j_1}^d < Z_{j_2}^d < \cdots < Z_{j_t}^d\}\right)}$$
$$= \frac{\frac{1}{(t!)^d}}{\frac{1}{[(t-1)!]^d}} = \frac{1}{t^d}.$$

Appendix B. Proof of Lemma 3

In round $t-1$, let H_t denote the set comprising the history of the losses of the experts and the sequence of arrivals $\mathbf{X}_{t-1}$. Given a realization of H_t, Y_t takes t^d possible values each corresponding to X_t belonging to one of the t^d partitions. From Lemma 2, the latter event has probability $1/t^d$. For $i, j \in \mathcal{B}_{t-1}$, let $c_j(i)$ denote the number of partitions of expert i caused by sampling X_t from expert j, and let $C_i = \sum_{j \in \mathcal{B}_{t-1}} c_j(i)$. The conditional expectation of Y_t given H_t is given by

$$\begin{aligned}\mathbb{E}[Y_t \mid H_t] &= \sum_{j \in \mathcal{B}_{t-1}} \frac{1}{t^d} \frac{\sum_{i \in \mathcal{B}_{t-1}} c_j(i) e^{-\eta L_{t-1}(i)}}{\sum_{j \in \mathcal{B}_{t-1}} e^{-\eta L_{t-1}(j)}} \\ &= \frac{1}{t^d} \frac{\sum_{i \in \mathcal{B}_{t-1}} C_i e^{-\eta L_{t-1}(i)}}{\sum_{j \in \mathcal{B}_{t-1}} e^{-\eta L_{t-1}(j)}}.\end{aligned} \tag{24}$$

Note that C_i is the total number of partitions of expert i created due to sampling X_t from all t^d experts. We compute C_i using the following counting argument. We say an expert i *shares* k hyperplanes with expert j if, for any point in i, exactly k out of the d orthogonal hyperplanes (parallel to the faces of $\mathbb{B}$) that pass through that point will partition expert i. We compute the number of experts that share exactly k hyperplanes with i as follows. Choose any k dimensions from d in $\binom{d}{k}$ possible ways. Further, choose any orthogonal hyperplane passing through i that is parallel to some dimension from the rest of $d-k$ dimensions. There will be $t-1$ basis hyperplanes, i.e., the hyperplanes that partitioned $\mathbb{B}$ by passing through $t-1$ points drawn by the environment, that are parallel to the chosen hyperplane and do not partition i. The $(t-1)^{d-k}$ partitions, which are formed by the intersection of the $t-1$ basis hyperplanes corresponding to each of the $d-k$ dimensions, do not share exactly $d-k$ hyperplanes with i, or they share exactly k hyperplanes with i. Therefore, the total number of experts that share exactly k hyperplanes with i is $\binom{d}{k}(t-1)^{d-k}$, and each point drawn from those experts will result in 2^k partitions of expert i. Since index i will

be assigned to one of its children (sub-partitions), we have $2^k - 1$ new experts from partitioning i.

$$\begin{aligned} C_i &= \sum_{k=1}^{d} \binom{d}{k} (2^k - 1)(t-1)^{d-k} \\ &= (t-1)^d \sum \binom{d}{k} \left(\frac{2}{t-1}\right)^k - \sum_{k=1}^{d} \binom{d}{k} (t-1)^{d-k} \\ &= (t-1)^d \left(\frac{t+1}{t-1}\right)^d - t^d = (t+1)^d - t^d. \end{aligned}$$

Indeed C_i is independent of i and is equal to the total number of new experts revealed in slot t. Substituting $C_i = (t+1)^d - t^d$ in (24), we obtain

$$\mathbb{E}[Y_t \mid H_t] = \left(1 + \frac{1}{t}\right)^d - 1.$$

The result follows from the fact that the conditional expectation is independent of the realization of H_t, and the upper bound is due to the following inequality.

$$(1+x)^r \leq 1 + (2^r - 1)x; \; x \in [0,1] \text{ and } r \in \mathbb{R} \setminus (0,1).$$