ADVERSARIALLY BATCHED ONLINE LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

We introduce *adversarially batched online learning*, where the learner may update its decision rule only at a restricted set of time steps. Equivalently, the time horizon is partitioned into *batches*, within which the learner must use a fixed policy. We study this model for four canonical problems: stochastic bandits, online convex optimization, prediction with expert advice, and adversarial bandits. For stochastic bandits, we identify a universal complexity measure based on relative batch growth and establish both gap-dependent and gap-independent regret bounds. We show that the multiplicative penalty relative to the standard gap-dependent complexity can grow superlinearly in batch hardness. For the remaining three, we show that the optimal regret is tightly characterized by the 2-norm of the batch-length sequence. The upper bound is achieved by a *batch-oblivious* learner who does not know the batch structure, via a generic adaptive doubling trick tailored to this complexity measure. In contrast, the lower bound holds even for *batch-aware* learners.

1 INTRODUCTION

Online learning provides a framework for sequential decision-making in uncertain environments. In standard settings, the learner plays an action at each round and observes the associated losses, which may immediately factor into the decision in the next round. The objective is to minimize the cumulative regret, defined as the difference between the total loss of the learner and that of the best fixed action in hindsight. While this framework has been studied extensively (e.g., see Cesa-Bianchi & Lugosi (2006); Lattimore & Szepesvári (2020)), the assumption that the learner can freely revise its policy after every round is often violated in practice. Frequent policy updates may be infeasible due to physical, logistical, or system constraints: outcomes may only become available after a trial phase, human annotations may need to be collected and validated together, and system-level updates may require synchronization.

This work addresses *Batched Online Learning*, where the learner is only allowed to update their decision rules at a set of specific time points. Prior work in this domain has largely focused on scenarios where the learner designs the batch structure to optimize the trade-off between the regret and the number of batches (e.g., Gao et al. (2019); Esfandiari et al. (2021)). In contrast, we investigate the *Adversarially Batched* setting, where the update points are imposed by an external environment (or adversary) and are immutable from the learner's perspective. For example, a clinical trial may release cumulative outcome data only at irregular interim review dates, a crowdsourcing platform may return validated annotations only after unpredictable quality-control cycles, and a distributed system may synchronize logs or deploy updates only when communication windows become available at unpredictable times.

The core challenge in our setting is that the learner must make a commitment at the beginning of each batch that cannot be revised until the batch ends, potentially accruing significant regret if the batch is long. We study this setting under four classic online learning problems: *Stochastic Bandits*, *Online Convex Optimization (OCO)*, *Experts Problem*, and *Adversarial Bandits*.

The main contributions of this work are summarized below. We formally introduce our setup in Section 1.1 and state our results in Section 1.2.

- We formulate a model of adversarially batched online learning, where the environment specifies the time steps on which the learner is allowed to update its decision rule.
- For stochastic bandits, we identify a universal batch-complexity measure that captures the multiplicative regret increase over the non-batched setting. We prove both gap-dependent and gap-independent regret bounds that depend on this quantity.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

- For the other three problems, we show that the optimal regret is characterized by the 2-norm of the batch length sequence by proving nearly matching upper and lower bounds. The upper bound is achieved by a *batch-oblivious* learner that does not know the batch structure, while the lower bound applies to *batch-aware* learners as well.

1.1 Problem Setup

In standard online learning settings, the learner interacts with the environment over a sequence of time steps $t \in [T] \coloneqq 1, 2, \ldots, T$. At each time step, the learner makes a decision based on the feedback observed from previous rounds, receives feedback from the environment, and updates its decision rule. The goal of the learner is to minimize the cumulative regret, which quantifies the sub-optimality of its actions compared to the optimal fixed action in hindsight.

Our model: Adversarially Batched Online Learning. Departing from the standard model, we study an *adversarially batched* setting: the learner will receive the cumulative loss at certain update time points, where both the loss at each round and the update points are determined by the environment (adversarially). In other words, the environment segments the time horizon into B batches, and the learner can only update its strategy at the end of each batch. We denote the set of update time steps by $\mathcal{T} = \{t_1, t_2, \ldots, t_B\}$. Equivalently, the instance can be described by its *batch structure* $\mathcal{B} = (b_1, b_2, \ldots, b_B) = (t_1 - 0, t_2 - t_1, \ldots, t_B - t_{B-1})$.[1] We will primarily use the batch structure $\mathcal{B}$ to specify an instance. We also write $L_i^{(j)} = \sum_{k=t_{j-1}+1}^{t_j} l_i^{(k)}$ as the cumulative loss of action $i \in [N]$ within batch $j \in [B]$.

We further distinguish between two settings regarding the learner's knowledge of the batch structure. In the *batch-aware* setting, the learner is given the entire update time set $\mathcal{T}$ (or equivalently the batch structure $\mathcal{B}$) before the learning process begins. The learner may then leverage this information, e.g., to tune the learning rate. In contrast, in the *batch-oblivious* setting, the learner does not know the future update times in advance. Instead, the learner only gets notified that a batch has ended at each time step $t_j \in \mathcal{T}$, and still does not know the future update times $t_{j+1}, \ldots, t_B$.

We instantiate the model of adversarially batched online learning in four canonical settings: *stochastic bandits*, *online convex optimization (OCO)*, *experts problem*, and *adversarial bandits*.

Stochastic Bandits. In *stochastic* Multi-Armed Bandits (MAB), each of the N arms is associated with a fixed but unknown reward distribution over $[0, 1]$. Each pull of an arm results in a reward independently sampled from that distribution. In the adversarially batched setting, the learner commits to a policy that specifies the sequence of arms to be pulled (e.g., a round-robin policy that cycles through $1, 2, \ldots, N$). The learner can only update this policy at the end of each batch. Let μ_i denote the mean reward of arm $i \in [N]$, and suppose that the learner pulls arm $i_t \in [N]$ at time t. Then, the learner incurs a regret of $\mathrm{Reg}_N^{(T)} \coloneqq T \cdot \max_{i^* \in [N]} \mu_{i^*} - \sum_{t=1}^{T} \mu_{i_t}$.

OCO. Let $\mathcal{K} \subseteq \mathbb{R}^d$ be a nonempty compact convex action set and $D := \sup_{x,y \in \mathcal{K}} \|x - y\|_2$ be its Euclidean diameter. At every round $t \in [T]$, the environment chooses a convex loss function $f_t : \mathcal{K} \to \mathbb{R}$. We assume that f_t is G-Lipschitz over $\mathcal{K}$, i.e., every subgradient $g \in \partial f_t(x)$ satisfies $\|g\|_2 \leq G$. In the adversarially batched setting, the learner must use the same action $x^{(j)} \in \mathcal{K}$ throughout each batch j. At the end of each batch, the learner receives first-order feedback for the losses incurred in that batch. Let $z_j = \sum_{t=t_{j-1}+1}^{t_j} g_t$ denote the aggregate subgradient in batch j where $g_t \in \partial f_t(x^{(j)})$. Since each f_t is G-Lipschitz, $\|z_j\|_2 \leq G b_j$. The regret is measured against the best fixed action in hindsight: $\mathrm{Reg} = \sum_{j=1}^{B} \sum_{t=t_{j-1}+1}^{t_j} f_t(x^{(j)}) - \min_{x^* \in \mathcal{K}} \sum_{t=1}^{T} f_t(x^*)$.

Experts Problem. In the *experts problem*, the learner interacts with N experts. At every round, the learner selects a distribution over the experts and incurs the expected loss under this distribution. The losses of all experts are then revealed to the learner. Formally, let $p_i^{(j)}$ denote the probability assigned to expert $i \in [N]$ at batch $j \in [B]$.[2] The regret of the learner in the adversarially batched experts problem is then defined as the difference between the cumulative loss of the learner and that of the best fixed expert in hindsight: $\mathrm{Reg}_N^{(B)} = \sum_{j=1}^{B} \sum_{i=1}^{N} \left(p_i^{(j)} L_i^{(j)} \right) - \min_{i^* \in [N]} \sum_{j=1}^{B} L_{i^*}^{(j)}$.

Adversarial Bandits. Similar to the experts problem, an adversarial MAB instance is specified by

[1] We view the end of the horizon as an update point, i.e., $t_B = T$.

[2] The learner must select the same probability distribution within each batch.

the losses $l_i^{(t)} \in [0,1]$ of all actions (or "arms") $i \in [N]$ at all time steps $t \in [T]$. The difference is that the learner, instead of observing all losses after each round, only sees the loss of the chosen arm. Specifically, we study the stricter setting in which the learner commits to a single arm for the entire batch, and only learns the losses associated with that action throughout the batch.

We measure performance by the expected static regret, which is also known as the *pseudo-regret*, which compares the learner's expected cumulative loss with the lowest expected loss over all arms in hindsight, i.e., $\overline{\mathrm{Reg}}_N^{(B)} = \mathbb{E}\left[\sum_{j=1}^{B}\sum_{i=1}^{N} p_i^{(j)} L_i^{(j)} - \min_{i^*\in[N]} \sum_{j=1}^{B} L_{i^*}^{(j)}\right]$. All expectations in the upper bounds are over the learner's internal randomization, and over the adversary's initial randomization if it draws the oblivious loss table from a distribution.

1.2 Our Results

For each problem, we characterize the effect of the batch structure through matching regret bounds governed by different complexity measures.

1.2.1 Stochastic Bandits

Upper Bounds. Let $\Delta_i := \max_{i^*\in[N]} \mu_{i^*} - \mu_i \geq 0$ denote the *gap* of arm i. The well-known Upper Confidence Bound (UCB) algorithm Auer et al. (2002b) has a gap-dependent regret bound of $\mathcal{O}\left((\log T)\cdot\sum_{i:\Delta_i>0} 1/\Delta_i\right)$ and a gap-independent bound of $\mathcal{O}(\sqrt{NT\log N})$.

For the batched setting, Gao et al. (2019) studied an easier setting where the update points are freely chosen by the learner. They proposed an algorithm and proved regret upper bounds under *specific* batch structures. In particular, given the time horizon T and the batch number B, the batch structure $b_j = \Theta(T^{j/B})$ leads to an expected regret upper bound of $\mathcal{O}\left(T^{1/B}\log N\log(NT)\frac{N}{\Delta_{\min}}\right)$, where $\Delta_{\min} := \min_{i:\Delta_i>0}\Delta_i$ is the minimum non-zero gap.

For our adversarially batched setting, we first introduce a universal complexity measure α that captures the hardness of an *arbitrary* batch structure in stochastic bandits.

Definition 1. *The hardness metric α of a batch structure $\mathcal{B} = (b_1, b_2, \ldots, b_B)$ is*

$$\alpha := \max_j \frac{b_j}{\sum_{k=1}^{j-1} b_k} = \max_j \frac{b_j}{t_{j-1}},$$

where, for brevity, we define $t_0 = 1$ to avoid division by zero.[3]

Intuitively, α captures the largest mismatch between a batch length and the evidence available before it. In batch j, the algorithm commits to b_j rounds using only the previous t_{j-1} observations; hence a large ratio b_j/t_{j-1} means long blind commitments given limited data, while small ratios mean information keeps pace with commitment length.

Remark 1. *Under the batch structure $b_j = \Theta\left(T^{j/B}\right)$, we have $\alpha = \Theta\left(T^{1/B}\right)$, so the regret bound of Gao et al. (2019) can be simplified to $\mathcal{O}\left(\alpha\log N\log(NT)\frac{N}{\Delta_{\min}}\right) = \widetilde{\mathcal{O}}(\alpha N/\Delta_{\min})$.*

As a warm-up, we extend the α-dependent regret bound of Gao et al. (2019) (Remark 1) to all batch structures: in Proposition 3 (Appendix B.3), we prove an $\widetilde{\mathcal{O}}(\min\{T, \alpha N/\Delta_{\min}\})$ regret bound for any batch structure. Our main technical contribution then goes further: we propose a successive-elimination algorithm with merging operations in Section 2, and prove both gap-dependent and gap-independent regret guarantees for adversarially batched stochastic bandits.

Theorem 1. *For adversarially batched stochastic bandits with N arms and an arbitrary batch structure $\mathcal{B}$ with hardness metric α, there is an algorithm with a gap-dependent expected regret of $\mathcal{O}\left(\min\left\{T, \alpha^2\log(NT)\sum_{i:\Delta_i>0}\frac{1}{\Delta_i}\right\}\right)$.*

Theorem 2. *For adversarially batched stochastic bandits with N arms and an arbitrary batch structure $\mathcal{B}$ with hardness metric α, there is an algorithm with a gap-independent expected regret of $\widetilde{\mathcal{O}}\left(\min\left\{T, \sqrt{\alpha TN}\right\}\right)$.*

[3] Equivalently, we can write $\alpha = \max\left\{b_1, \max_{j\geq 2}\frac{b_j}{t_{j-1}}\right\}$ without the convention that $t_0 = 1$.

Compared with Gao et al. (2019), our results apply to any batch structure $\mathcal{B}$, rather than only to specially designed ones. Our gap-dependent bound (Theorem 1) replaces the coarse worst-arm dependence $N/\Delta_{\min}$ by the standard UCB-type complexity $\sum_{i:\Delta_i>0} \Delta_i^{-1}$, yielding an improvement of up to a factor of N in its dependence on the gaps. This refinement costs an additional factor of α, but only for the gap-dependent bound. The gap-independent bound, on the other hand, has a better (square root) dependence on the batch hardness measure α. More particularly, when no batch is significantly longer than the cumulative length of the preceding batches, i.e., $\alpha = \mathcal{O}(1)$, both bounds recover the standard non-batched regret rates up to polylogarithmic factors.

Lower Bounds. On the lower-bound side, we show that the dependence on α is unavoidable. Firstly, in Appendix C.1, we extend the analysis of Gao et al. (2019) to establish both gap-dependent and gap-independent worst-case lower bounds.

Proposition 1. *There exists a family of hard instances of adversarially batched stochastic bandits such that every algorithm suffers a gap-dependent regret lower bound of* $\Omega\big(\alpha \sum_{i:\Delta_i>0} \Delta_i^{-1}\big)$.

Theorem 3. *There exists a family of hard instances of adversarially batched stochastic bandits such that every algorithm suffers a gap-independent regret lower bound of* $\Omega\Big(\sqrt{\alpha N T}\Big)$.

Theorem 3 gives a gap-independent lower bound that matches the upper bound in Theorem 2 up to logarithmic factors on a suitable family of hard batch structures. On the other hand, for the gap-dependent setting, Theorem 1 and Proposition 1 still leave a multiplicative gap of α. To narrow this gap, in Appendix C.2 we construct a carefully designed hard instance and establish a stronger gap-dependent lower bound with an $\alpha^{3/2}$ dependence.

Theorem 4. *There exists a hard instance of adversarially batched stochastic bandits such that every algorithm suffers an expected cumulative regret of* $\Omega\Big(\alpha^{\frac{3}{2}} \sum_{i:\Delta_i>0} \frac{1}{\Delta_i}\Big)$.

Moreover, we present an abstract construction in Appendix C.3, showing that the comparison used in our gap-dependent analysis requires a factor of $\Omega(\alpha^{7/4})$ over abstract sequences.

1.2.2 Convex Optimization-based Problems

Batch-Aware Setting. In Appendices D.1, E.1 and F.1, we prove the following regret upper bounds for the three problems with convex optimization formulations under the adversarially batched model, including OCO, the experts problem (which can be formulated as an Online Linear Optimization (OLO) over the probability simplex), and the adversarial bandits (which can be formulated as an instance of bandit convex optimization), using classic algorithms: Online Gradient Descent (OGD) (Zinkevich, 2003), Hedge (Freund & Schapire, 1997; Cesa-Bianchi & Lugosi, 2006), and EXP3 (Auer et al., 2002b).

Theorem 5. *For any batch structure $\mathcal{B}$, the following regret guarantees hold:*

1. *In the adversarially batched OCO problem, projected OGD with learning rate $\eta > 0$ satisfies* $\mathrm{Reg} \le \frac{D^2}{2\eta} + \frac{\eta G^2}{2}\|b\|_2^2$. *Setting* $\eta = \frac{D}{G\|b\|_2}$ *gives a regret upper bound of* $\mathcal{O}(DG\|b\|_2)$.
2. *In the adversarially batched experts problem with N experts, Hedge with learning rate $\eta > 0$ satisfies* $\mathrm{Reg} \le \frac{\ln N}{\eta} + \frac{\eta}{8}\|b\|_2^2$. *Setting* $\eta = \frac{\sqrt{8\ln N}}{\|b\|_2}$ *gives a regret upper bound of* $\mathcal{O}\left(\min\left\{\|b\|_2\sqrt{\log N}, T\right\}\right)$.
3. *In the adversarially batched adversarial bandits problem with N arms, EXP3 with learning rate $\eta > 0$ satisfies* $\overline{\mathrm{Reg}} \le \frac{\ln N}{\eta} + \frac{\eta N}{2}\|b\|_2^2$. *Setting* $\eta = \frac{\sqrt{2\ln N}}{\|b\|_2\sqrt{N}}$ *bounds the pseudo-regret by* $\mathcal{O}\left(\min\left\{\|b\|_2\sqrt{N\log N}, T\right\}\right)$.

We complement the upper bounds with matching or nearly matching worst-case lower bounds (up to logarithmic factors), demonstrating that the dependence on $\|b\|_2$ is unavoidable. For OCO, the lower bound holds for every fixed batch structure, whereas for experts and adversarial bandits we establish the corresponding worst-case lower bounds through families of hard adversarially batched instances.

Theorem 6. *The following lower bounds hold for adversarially batched online learning.*

1. *For every batch structure $\mathcal{B}$, every algorithm for adversarially batched OCO suffers an expected regret of* $\Omega\left(DG\|b\|_2\right)$.

2. *For every $N \geq 2$, there exists a family of adversarially batched expert instances such that every algorithm suffers a worst-case regret of* $\Omega\left(\|b\|_2\sqrt{\log N}\right)$.
3. *For every $N \geq 2$, there exists a family of adversarially batched adversarial-bandit instances such that every algorithm suffers a worst-case pseudo-regret of* $\Omega\left(\|b\|_2\sqrt{N}\right)$.

Batch-Oblivious Setting. Algorithms introduced above are batch-aware, since the optimal learning rate depends on the quantity $\|b\|_2$. Towards extending the upper bound to the batch-oblivious setting, we show in Appendices G and H that algorithms from scale-free and delayed-feedback online learning can be adapted. Specifically, for scale-free online learning, we treat each batch as a meta-round and deploy SOLO FTRL (Orabona & Pál, 2018) for OCO and the experts problem, and UMAB-NN (Chen & Zhang, 2023) for adversarial bandits, to recover the same order of upper bounds. For delayed-feedback models, we view batched feedback as a special case of delayed feedback and apply the delay-adaptive algorithms of Joulani et al. (2016) to OCO and the experts problem. We note that existing delayed-feedback results do not directly apply to the stricter adversarial bandits setting studied here, as they typically allow the learner to select new arms while feedback from previous rounds remains outstanding.

1.2.3 An Adaptive Doubling Trick

The reductions above recover batch-oblivious guarantees through existing scale-free or delay-adaptive algorithms. We additionally provide a generic conversion that preserves the choice of the base algorithm: any restartable algorithm satisfying the fixed-learning-rate bound and instantaneous-regret condition in Theorem 7 can be made batch-oblivious using only learning-rate selection and restarts, with no additional logarithmic factor in its regret guarantee.

Theorem 7. *Consider a base online learning algorithm $\mathcal{A}_\eta$, parametrized by a learning rate $\eta > 0$. Suppose that there exist known quantities $A, D > 0$, such that whenever $\mathcal{A}_\eta$ is run from scratch on any batch structure $\mathcal{B}$, its regret satisfies* $\mathrm{Reg}(\mathcal{A}_\eta) = \mathcal{O}\left(\frac{A}{\eta} + D\eta\|b\|_2^2\right)$. *Suppose the instantaneous regret against any comparator is at most L, where $L = \mathcal{O}(\sqrt{AD})$, then there exists a batch-oblivious algorithm whose regret satisfies* $\mathcal{O}\left(\sqrt{AD}\|b\|_2\right)$.

In particular, Theorem 7 allows OGD, Hedge, and EXP3 to retain their batch-aware rates without requiring advance knowledge of the batch structure. The main difficulty is that a batch may overshoot the current budget by an arbitrarily large amount before its length becomes known. Proposition 2 (proved in Appendix I) shows a naïve implementation of the standard doubling trick (Cesa-Bianchi & Lugosi, 2006; Besson & Kaufmann, 2018) can fail dramatically. In contrast, our analysis controls this batch separately, and the next budget is calibrated to the realized sum of squared batch lengths.

Proposition 2. *There exists a family of instances of the adversarially batched experts problem on which the naïve doubling-trick adaptation of Hedge incurs an $\Omega(T)$ regret, whereas a batch-aware algorithm can achieve an $\mathcal{O}(T^{3/4})$ regret.*

1.3 Related Work

Our work builds on the vast literature on online learning, including stochastic MAB (Lai & Robbins, 1985; Auer et al., 2002a), OCO (Zinkevich, 2003), the experts problem (Littlestone & Warmuth, 1994; Freund & Schapire, 1997), and adversarial bandits (Auer et al., 2002b). We refer Cesa-Bianchi & Lugosi (2006); Lattimore & Szepesvári (2020) for a comprehensive introduction.

Selective Prediction with Limited Selectivity. Most similar to our setup is the recent work of Liu & Qiao (2025), which builds on the *selective prediction* formulation of Drucker (2013); Qiao & Valiant (2019) and studies a variant in which the forecaster can only initiate predictions at a prescribed subset of time steps. Their setting can be viewed as prediction-focused in the sense that restrictions are imposed on when predictions may be made, whereas we generalize this timing constraint to regret minimization in online learning.

Batched Bandits. The constraint of batched feedback in multi-armed bandits has garnered significant attention recently. The seminal work of Perchet et al. (2016) characterized the minimax regret for stochastic bandits with a limited number of batches, showing that $\mathcal{O}(\log T)$ batches are sufficient to achieve the optimal $\mathcal{O}(\sqrt{T})$ regret. Gao et al. (2019) extended this by analyzing the trade-off between the number of batches and regret, introducing a "grid" structure. Esfandiari et al. (2021)

further improved these bounds for specific batch structures. Jin et al. (2021) studied the anytime version of batched bandits and developed algorithms that allow the learner to adaptively choose update points while approaching the minimal batch complexity. Crucially, these works generally assume the learner can design the grid (e.g., the batch structure) to minimize the regret. Our setting differs fundamentally: the batch structure is *adversarial* and imposed by the environment, forcing the learner to act under external constraints rather than optimizing the schedule.

Scale-Free and Loss-Adaptive Online Learning. Existing online learning frameworks can potentially be deployed in the adversarially batched model through batch-level aggregation. In this study, we show batch-oblivious reductions in the experts problem and adversarial bandits using scale-free and loss-adaptive online learning. Specifically, Orabona & Pál (2018) developed scale-free online learning algorithms whose regret guarantees adapt to the observed magnitudes of the loss vectors. By treating each batch as a single meta-round, SOLO FTRL can be directly deployed for OCO and the experts problem while retaining the desired dependence on the realized batch lengths. For adversarial bandits, an analogous reduction uses the unbounded-loss bandit algorithm of Chen & Zhang (2023).

Delayed Feedback Model. Our problem is also related to online learning with delayed feedback (Mesterharm, 2005; Joulani et al., 2013; 2016; Schlisselberg et al., 2025). For *OCO* and the *experts problem*, similar to scale-free reductions (Appendix G), we show in Appendix H that delayed-feedback models can be reduced to obtain the same order of batch-oblivious regret upper bounds. In contrast, existing delayed-bandit results do not directly apply to our *adversarial bandits* problem, which requires the learner to commit to a single arm throughout each batch. For *stochastic bandits*, existing delayed-feedback results (Joulani et al., 2013; Schlisselberg et al., 2025) typically incur an additive penalty on top of the standard regret, controlled by *absolute delay*. In the adversarially batched setting, this can be expressed as $\mathcal{O}\left(\sum_{i:\Delta_i>0} \frac{\log T}{\Delta_i} + b_{\max} \sum_{i:\Delta_i>0} \Delta_i\right)$ or $\tilde{\mathcal{O}}\left(\sqrt{NT} + N b_{\max}\right)$, where $b_{\max} := \max_j b_j$. In contrast, our analysis provides a multiplicative characterization through the *relative growth* measure α. Consequently, we provide complementary bounds that can be substantially sharper whenever batch sizes are large but remain controlled relative to the accumulated history. For example, under geometrically growing batches, the maximum batch length and the number of outstanding observations can be $\Theta(T)$, whereas $\alpha = \Theta(1)$.

2 Stochastic Bandits

Our algorithm (Algorithm 1) builds on the Batched Successive Elimination algorithm of Gao et al. (2019), with an additional *merging* operation. Specifically, let $\gamma > 0$ be a constant parameter associated with the UCB bound. Let A_j denote the set of active arms at the beginning of batch j and $n_j := |A_j|$. Then, $S_j := \sum_{k=1}^{j} \frac{b_k}{n_k}$ is the total number of times arm $i \in A_j$ is pulled by the end of batch j (i.e., at time step t_j)[4]. Algorithm 1 eliminates every arm with an empirical mean that is lower, by a margin of at least $\sqrt{\frac{\gamma \ln(NT)}{S_j}}$, than the highest empirical mean $\hat{\mu}_{a^*}$.

A Gap-Dependent Upper Bound. To analyze Algorithm 1, we define j_i to be the index of the block after which arm i is *intended* to be eliminated. Recall that $\Delta_i = \max_{i^* \in [N]} \mu_{i^*} - \mu_i$ is the gap of arm i. Then, j_i is formally defined as $j_i := \min\left\{j \in [B] : S_j \geq \frac{4\gamma \ln(NT)}{\Delta_i^2}\right\}$ if this set is nonempty; otherwise, we define $j_i = B + 1$.

We will condition on a "good event" $\mathcal{E}^{\text{good}}$ defined as follows: (1) the optimal arm i^* remains active throughout the algorithm; (2) every sub-optimal arm i with $\Delta_i > 0$ and $j_i \leq B$ is eliminated by the end of batch j_i. Lemma 1, proved in Appendix B.1, upper bounds the probability that $\mathcal{E}^{\text{good}}$ does not happen by $1/T$. The contribution of this case to the expected regret is at most $T \cdot \frac{1}{T} = 1$. Then, Lemma 2 upper bounds the regret assuming event $\mathcal{E}^{\text{good}}$. These two lemmas together yield Theorem 1. We provide a brief proof sketch below for Lemma 2 and defer the full proof to Appendix B.2.

Lemma 1. $\Pr[\mathcal{E}^{\text{good}}] \geq 1 - 1/T$.

Lemma 2. *Conditioned on $\mathcal{E}^{\text{good}}$, Algorithm 1 has regret* $\mathcal{O}\left(\alpha^2 \log(NT) \sum_{i:\Delta_i>0} \Delta_i^{-1}\right)$.

[4]Under the least-pulled-arm rule, the cumulative pull counts of active arms differ by at most one.

Algorithm 1 Batched Arm Elimination with Merging

Require: Time horizon T. Number of arms N. Parameter $\gamma > 0$.

Initialize $j \leftarrow 1$; $A_1 \leftarrow [N]$; $b_j \leftarrow 0$; ▷ all arms are active
for $t = 1, 2, \ldots, T$ **do**
  Pull the least-pulled arm in A_j; ▷ pull active arms in a round-robin fashion
  $b_j \leftarrow b_j + 1$; ▷ track batch length
  if t is among the updating time steps **then**
    Update the empirical mean $\hat{\mu}_a$ for each arm $a \in A_j$;
    if $j = 1$ or $b_j \geq 2 \cdot b_{j-1}$ **then** ▷ the merged batch is long enough
      $A_{j+1} \leftarrow \left\{ a \in A_j : \max_{a^* \in A_j} \hat{\mu}_{a^*} - \hat{\mu}_a \leq \sqrt{\frac{\gamma \log(NT)}{S_j}} \right\}$; ▷ eliminate arms
      $j \leftarrow j + 1$; $b_j \leftarrow 0$;
    end if
  end if
end for

Proof sketch of Lemma 2. Suppose that a sub-optimal arm i remains active at the beginning of block j, i.e., it has not been eliminated by the end of block $j-1$. By the definition of j_i and the good event, we have $\Delta_i \leq \sqrt{\frac{4\gamma \ln(NT)}{S_{j-1}}}$. Then, the total regret incurred by arms eliminated at the end of block j is at most $(n_j - n_{j+1}) S_j \sqrt{\frac{4\gamma \ln(NT)}{S_{j-1}}}$, where n_j is the number of active arms during block j and $S_j = \sum_{k=1}^{j} \frac{b_k}{n_k}$ is the number of pulls per active arm up to the end of block j. Assume for simplicity that the block lengths grow geometrically except for the last one, i.e., $b_j \geq 2b_{j-1}$ for all $j \in [B-1]$ while $b_B \leq 2b_{B-1}$. Under this assumption, we can show that $t_j < 2b_j$ and $S_j = \Theta(b_j/n_j)$. The total conditional regret is then bounded by $\mathcal{O}(\sqrt{\log(NT)})$ times

$$\sum_{j=1}^{B} (n_j - n_{j+1}) \frac{b_j}{n_j} \sqrt{\frac{n_{j-1}}{b_{j-1}}}. \tag{1}$$

On the other hand, the $\sum_{i:\Delta_i>0} 1/\Delta_i$ term in the target bound is at least, up to a $\sqrt{\log(NT)}$ factor,

$$\sum_{j=1}^{B} (n_j - n_{j+1}) \sqrt{\frac{b_{j-1}}{n_{j-1}}}, \tag{2}$$

To bound Equation (1) by Equation (2), we split blocks into two types. For the first type, the active set size does not shrink too quickly, i.e., $n_j \geq n_{j-1}/(C\alpha)$ for a constant C. For each such block j, the bounds $b_j \leq \mathcal{O}(\alpha) \cdot b_{j-1}$ and $n_j \geq n_{j-1}/\mathcal{O}(\alpha)$ directly imply that the jth term in equation 1 is at most $\mathcal{O}(\alpha^2)$ times the corresponding term in equation 2. For the second type, the active set shrinks rapidly, i.e., $n_j < n_{j-1}/(C\alpha)$. Whenever we have consecutive blocks $j_0, j_0+1, \ldots, j_1$ of this type, their contribution to equation 1 forms a vanishing geometric series, and can be charged to the preceding block (i.e., the (j_0-1)-th term), which is then controlled by $\mathcal{O}(\alpha^2)$ times the corresponding term in equation 2.

Taking both block types into account, we have $\sum_{j=1}^{B} (n_j - n_{j+1}) \frac{b_j}{n_j} \sqrt{\frac{n_{j-1}}{b_{j-1}}} \leq \mathcal{O}(\alpha^2) \sum_{j=1}^{B} (n_j - n_{j+1}) \sqrt{\frac{b_{j-1}}{n_{j-1}}}$, which establishes the $\mathcal{O}\left(\alpha^2 \log(NT) \sum_{i:\Delta_i>0} \frac{1}{\Delta_i}\right)$ regret bound. For a general block structure, the algorithm merges consecutive blocks so that the resulting block sequence $\mathcal{B}'$ satisfies the above geometric-growth condition. The merging only changes the complexity measure α by at most a constant factor, namely, $\alpha(\mathcal{B}') \leq 4\alpha(\mathcal{B}) + 2$. Therefore, the same regret bound holds for the original adversarially batched instance. □

A Gap-Independent Upper Bound. We now prove the gap-independent regret bound of Algorithm 1 (Theorem 2) via a simple application of the Cauchy-Schwarz inequality (Schwarz, 1890).

Proof of Theorem 2. Condition on event $\mathcal{E}^{\text{good}}$, under which $\tau_i \leq j_i$. For each arm $i \in [N]$, let τ_i denote the number of blocks in which it remains active. Then, arm i is pulled S_{τ_i} times throughout

the algorithm. By definition of the regret, we have

$$\mathrm{Reg}_N^{(T)} = \sum_{i:\Delta_i>0} \left(\Delta_i \sqrt{S_{\tau_i}}\right) \cdot \sqrt{S_{\tau_i}} \le \sqrt{\sum_{i:\Delta_i>0} \Delta_i^2 S_{\tau_i}} \cdot \sqrt{\sum_{i:\Delta_i>0} S_{\tau_i}} \le \sqrt{\sum_{i:\Delta_i>0} \Delta_i^2 S_{\tau_i}} \cdot \sqrt{T},$$

where the second step applies the Cauchy-Schwarz inequality. Let $\pi_1, \pi_2, \ldots, \pi_N \in [N]$ be a permutation of the N arms in the order in which they are eliminated. Formally, we break ties arbitrarily and list all arms that remain active at the end of the list. For each $i \in [N]$, let $\sigma_i \in [N]$ be the index of arm i in permutation π. By definition, whenever arm i remains active in the algorithm, the number of active arms is at least $N - \sigma_i + 1$. For each arm i with $\Delta_i > 0$, we have

$$S_{\tau_i} = \sum_{k=1}^{\tau_i} \frac{b_k}{n_k} \le \frac{\sum_{k=1}^{\tau_i} b_k}{N - \sigma_i + 1} = \frac{t_{\tau_i}}{N - \sigma_i + 1} \le \frac{(1+\alpha) t_{\tau_i - 1}}{N - \sigma_i + 1},$$

where the last inequality also holds for $\tau_i = 1$ under the convention $t_0 = 1$. Also note that $S_j \ge \sum_{k=1}^{j} b_k / N = t_j / N$. If $\tau_i \ge 2$, then $\tau_i - 1 < j_i$, so the definition of j_i gives

$$t_{\tau_i - 1} \le N S_{\tau_i - 1} < \frac{4\gamma N \ln(NT)}{\Delta_i^2}.$$

For $\tau_i = 1$, the same upper bound on $t_{\tau_i - 1} = t_0 = 1$ holds since $\Delta_i \le 1$ and γ is sufficiently large. Therefore, using $\alpha \ge 1$, we obtain $S_{\tau_i} \le \mathcal{O}(\alpha) \cdot \frac{N \ln(NT)}{(N - \sigma_i + 1)\Delta_i^2}$. Consequently, on $\mathcal{E}^{\mathrm{good}}$,

$$\mathrm{Reg}_N^{(T)} \le \sqrt{\sum_{i:\Delta_i>0} \Delta_i^2 S_{\tau_i}} \cdot \sqrt{T} \le \mathcal{O}(\sqrt{\alpha}) \cdot \sqrt{T \sum_{i:\Delta_i>0} \frac{N \ln(NT)}{N - \sigma_i + 1}}.$$

Finally, $\sum_{i:\Delta_i>0} (N - \sigma_i + 1)^{-1} \le \sum_{i=1}^{N} 1/i = \mathcal{O}(\log N)$ gives a bound of $\tilde{\mathcal{O}}\left(\sqrt{\alpha T N}\right)$. Combining with the trivial bound T and $\mathcal{O}(1)$ regret from bad events, completes the proof. □

An Improved Gap-dependent Lower Bound. Toward narrowing the gap between the gap-dependent upper and lower bounds, we tighten the lower bound extended from previous work (Gao et al. (2019)) from $\Omega\left(\alpha \sum_{i:\Delta_i>0} \Delta_i^{-1}\right)$ (Proposition 1) to $\Omega\left(\alpha^{\frac{3}{2}} \sum_{i:\Delta_i>0} \Delta_i^{-1}\right)$ (Theorem 4) by constructing a carefully designed hard instance. Specifically, we randomly hide a pair of nearly tied Bernoulli arms among N arms and adopt a three-batch structure of lengths (N, N^2, N^3), so that with constant probability the learner allocates only $O(N)$ samples to the two special arms before the last batch and hence cannot reliably distinguish the better one, resulting in $\Omega(N^{5/2}) = \Omega\big(\alpha^{3/2} \sum_{i:\Delta_i>0} \Delta_i^{-1}\big)$. The full proof of Theorem 4 is deferred to Appendix C.2.

3 An Adaptive Doubling Trick

To prove Theorem 7, we introduce an *adaptive* doubling trick algorithm, as shown in Algorithm 2. The algorithm maintains an epoch-wise guess $g^{(r)}$ for the sum of squared batch lengths, and runs $\mathcal{A}_\eta$ with learning rate $\eta^{(r)} = \sqrt{A / \left(D g^{(r)}\right)}$. Within each epoch, it tracks the observed quantity $a^{(r)} = \sum b_j^2$ and starts a new epoch once $a^{(r)} \ge g^{(r)}$. The next guess is set to twice the realized sum in epoch r, i.e., $g^{(r+1)} = 2a^{(r)}$. Thereby, the learning rate is recalibrated only when the observed batch complexity outgrows the current guess, yielding regret controlled by the realized squared-batch mass across epochs.

Proof of Theorem 7. Let R be the index of the last nonempty epoch, and let $\mathrm{Reg}^{(r)}$ denote the regret incurred in epoch $r \in \{0, 1, \ldots, R\}$. For randomized algorithms, all regret bounds below are understood in expectation. By sub-additivity of regret, it suffices to bound each $\mathrm{Reg}^{(r)}$ separately. Let L denote the upper bound on instantaneous regret implied by the problem setup. For the choices of A, D in these three applications, $L = \mathcal{O}(\sqrt{AD})$.

Regret in epoch 0**.** Since epoch 0 consists of a single block of length b_1, $\mathrm{Reg}^{(0)} \le L b_1 = L\sqrt{a^{(0)}} = \mathcal{O}\left(\sqrt{AD a^{(0)}}\right)$. If $R = 0$, this already proves the desired bound. Henceforth, assume $R \ge 1$.

Algorithm 2 Adaptive Learning Rate with Doubling Trick

Require: Base algorithm $\mathcal{A}_\eta$
  Play the uniform distribution until the first batch ends (i.e., at time $b_1 = t_1$);
  $j \leftarrow 1$; $a^{(0)} \leftarrow b_1^2$; ▷ block 1 is regarded as epoch 0
  $r \leftarrow 1$; $a^{(r)} \leftarrow 0$; ▷ start epoch 1
  $g^{(1)} \leftarrow 2a^{(0)}$; $\eta^{(1)} \leftarrow \sqrt{\frac{A}{Dg^{(1)}}}$; ▷ guess squared 2-norm in epoch 1, set learning rate
  for $t = t_1 + 1, t_1 + 2, \ldots, T$ **do**
    Play the action/distribution output by $\mathcal{A}_\eta$;
    if weights can be updated at time t **then**
      $j \leftarrow j + 1$; $t_j \leftarrow t$; $b_j \leftarrow t_j - t_{j-1}$; ▷ batch j is completed
      $a^{(r)} \leftarrow a^{(r)} + b_j^2$; ▷ update squared 2-norm
      Update $\mathcal{A}_\eta$ with learning rate $\eta^{(r)}$;
      if $a^{(r)} \geq g^{(r)}$ **then**
        Restart $\mathcal{A}_\eta$;
        $r \leftarrow r + 1$; $a^{(r)} \leftarrow 0$; ▷ start a new epoch r
        $g^{(r)} \leftarrow 2a^{(r-1)}$; $\eta^{(r)} \leftarrow \sqrt{\frac{A}{Dg^{(r)}}}$; ▷ guess squared 2-norm, set learning rate
      end if
    end if
  end for

Regret within epochs $1, 2, \ldots, R$. Fix $r \in [R]$ and suppose that epoch r spans blocks $j_0, j_0 + 1, \ldots, j_1$. The regret can be decomposed into two parts: $\mathrm{Reg}^{(r)} \leq \mathrm{Reg}^{(r)}_{\mathsf{guess}} + \mathrm{Reg}^{(r)}_{\mathsf{remain}}$, where $\mathrm{Reg}^{(r)}_{\mathsf{guess}}$ is the regret in blocks j_0 through $j_1 - 1$ and $\mathrm{Reg}^{(r)}_{\mathsf{remain}}$ is the regret within the last block j_1. For the first part $\mathrm{Reg}^{(r)}_{\mathsf{guess}}$, note that Algorithm 2 guarantees $\sum_{j=j_0}^{j_1-1} b_j^2 < g^{(r)}$; otherwise, epoch r should have finished after block $j_1 - 1$. Also recall that $\eta^{(r)} = \Theta\left(\sqrt{A/(Dg^{(r)})}\right)$. For a fixed batch structure, the epoch boundaries are deterministic. Thus, applying the regret guarantee of $\mathcal{A}_\eta$ to blocks j_0 through $j_1 - 1$ gives

$$\mathrm{Reg}^{(r)}_{\mathsf{guess}} \leq \mathcal{O}\left(\frac{A}{\eta^{(r)}} + D\eta^{(r)} \sum_{j=j_0}^{j_1-1} b_j^2\right) \leq \mathcal{O}\left(\frac{A}{\eta^{(r)}} + D\eta^{(r)} g^{(r)}\right) = \mathcal{O}\left(\sqrt{ADg^{(r)}}\right).$$

If this prefix is empty, its regret is zero, and the same bound holds. Since $g^{(r)} = 2a^{(r-1)}$, we have $\mathrm{Reg}^{(r)}_{\mathsf{guess}} = \mathcal{O}\left(\sqrt{ADa^{(r-1)}}\right)$. The second part $\mathrm{Reg}^{(r)}_{\mathsf{remain}}$ is upper bounded by Lb_{j_1}. Since $a^{(r)} = \sum_{j=j_0}^{j_1} b_j^2 \geq b_{j_1}^2$, we have $\mathrm{Reg}^{(r)}_{\mathsf{remain}} \leq L\sqrt{a^{(r)}} = \mathcal{O}\left(\sqrt{ADa^{(r)}}\right)$. Thus,

$$\mathrm{Reg}^{(r)} \leq \mathrm{Reg}^{(r)}_{\mathsf{guess}} + \mathrm{Reg}^{(r)}_{\mathsf{remain}} \leq \mathcal{O}\left(\sqrt{ADa^{(r-1)}} + \sqrt{ADa^{(r)}}\right).$$

Put everything together. We have shown that $\mathrm{Reg}^{(0)} = \mathcal{O}\left(\sqrt{ADa^{(0)}}\right)$ and, for each $r \in [R]$, $\mathrm{Reg}^{(r)} \leq \mathcal{O}\left(\sqrt{ADa^{(r-1)}} + \sqrt{ADa^{(r)}}\right)$. Summing over all epochs shows that the regret of Algorithm 2 is upper bounded by $\sum_{r=0}^{R} \mathrm{Reg}^{(r)} \leq \sum_{r=0}^{R} \mathcal{O}\left(\sqrt{ADa^{(r)}}\right) \leq \sum_{r=0}^{R-1} \mathcal{O}\left(\sqrt{ADa^{(r)}}\right) + \mathcal{O}\left(\sqrt{AD}\|b\|_2\right)$, where the second step applies $a^{(R)} \leq \sum_{j=1}^{B} b_j^2 = \|b\|_2^2$. Recall that Algorithm 2 sets $g^{(r)} = 2a^{(r-1)}$ for each epoch $r \in [R]$. Moreover, for every $r \in [R-1]$, $a^{(r)} \geq g^{(r)}$ holds when epoch r ends. Thus, $a^{(r)} \geq g^{(r)} = 2a^{(r-1)}$, which further implies $a^{(r)} \leq \frac{a^{(R-1)}}{2^{R-r-1}}$ for every $r \in \{0, 1, \ldots, R-1\}$. Therefore,

$$\sum_{r=0}^{R-1} \mathcal{O}\left(\sqrt{ADa^{(r)}}\right) \leq \mathcal{O}\left(\sqrt{ADa^{(R-1)}}\right) \sum_{r=0}^{R-1} \sqrt{\frac{1}{2^{R-r-1}}} \leq \mathcal{O}\left(\sqrt{ADa^{(R-1)}}\right).$$

Taking $\mathcal{O}\left(\sqrt{ADa^{(R-1)}}\right) \leq \mathcal{O}\left(\sqrt{AD}\|b\|_2\right)$ completes the proof. □

REFERENCES

Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2):235–256, 2002a. 1.3

Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1):48–77, 2002b. 1.2.1, 1.2.2, 1.3, F

Lilian Besson and Emilie Kaufmann. What doubling tricks can and can't do for multi-armed bandits. *arXiv preprint arXiv:1803.06971*, 2018. 1.2.3

Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006. 1, 1.2.2, 1.2.3, 1.3

Mingyu Chen and Xuezhou Zhang. Improved algorithms for adversarial bandits with unbounded losses. *arXiv preprint arXiv:2310.01756*, 2023. 1.2.2, 1.3, G.2, 10

Andrew Drucker. High-confidence predictions under adversarial uncertainty. *Transactions on Computation Theory*, 5(3):12, 2013. 1.3

Hossein Esfandiari, Amin Karbasi, Abbas Mehrabian, and Vahab Mirrokni. Regret bounds for batched bandits. In *AAAI Conference on Artificial Intelligence (AAAI)*, pp. 7340–7348, 2021. 1, 1.3

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. 1.2.2, 1.3, E

Zijun Gao, Yanjun Han, Zhimei Ren, and Zhengqing Zhou. Batched multi-armed bandits problem. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 503–513, 2019. 1, 1.2.1, 1, 1.2.1, 1.2.1, 1.2.1, 1.3, 2, 2, 3, C.1, C.1, C.1

Tianyuan Jin, Jing Tang, Pan Xu, Keke Huang, Xiaokui Xiao, and Quanquan Gu. Almost optimal anytime algorithm for batched multi-armed bandits. In *International Conference on Machine Learning*, pp. 5065–5073. PMLR, 2021. 1.3

Pooria Joulani, Andras Gyorgy, and Csaba Szepesvári. Online learning under delayed feedback. In *International Conference on Machine Learning (ICML)*, pp. 1453–1461, 2013. 1.3

Pooria Joulani, Andras Gyorgy, and Csaba Szepesvári. Delay-tolerant online convex optimization: Unified analysis and adaptive-gradient algorithms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016. 1.2.2, 1.3, H, 11

Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985. 1.3

Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020. 1, 1.3

Nick Littlestone and Manfred K Warmuth. The weighted majority algorithm. *Information and computation*, 108(2):212–261, 1994. 1.3

Licheng Liu and Mingda Qiao. Online prediction with limited selectivity. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. 1.3

Chris Mesterharm. On-line learning with delayed label feedback. In *International Conference on Algorithmic Learning Theory*, pp. 399–413. Springer, 2005. 1.3

Francesco Orabona and Dávid Pál. Scale-free online learning. *Theoretical Computer Science*, 716: 50–69, 2018. 1.2.2, 1.3, G.1, 9

Vianney Perchet, Philippe Rigollet, Sylvain Chassang, and Erik Snowberg. Batched bandit problems. *The Annals of Statistics*, 44(2):660–681, 2016. 1.3

Mingda Qiao and Gregory Valiant. A theory of selective prediction. In *Conference on Learning Theory (COLT)*, pp. 2580–2594, 2019. 1.3

Ofir Schlisselberg, Tal Lancewicki, Peter Auer, and Yishay Mansour. Improved best-of-both-worlds regret for bandits with delayed feedback. *Advances in Neural Information Processing Systems*, 38: 2035–2072, 2025. 1.3

Hermann Amandus Schwarz. Ueber ein die flächen kleinsten flächeninhalts betreffendes problem der variationsrechnung: Festschrift zum siebzigsten geburtstage des herrn karl weierstrass. In *Gesammelte Mathematische Abhandlungen: Erster Band*, pp. 223–269. Springer, 1890. 2

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th international conference on machine learning (icml-03)*, pp. 928–936, 2003. 1.2.2, 1.3

Appendix

A Discussion **13**

B Upper Bounds for Stochastic Bandits **13**

B.1 Proof of Lemma 1 13

B.2 Proof of Lemma 2 14

B.3 Extension of Existing Bounds for Arbitrary Batch Structure 16

C Lower Bounds for Stochastic Bandits **17**

C.1 Lower Bounds Extended from Gao et al. (2019) 17

C.2 An Improved Gap-Dependent Lower Bound 20

C.3 An Abstract Counterexample of the Current Gap-Dependent Analysis 22

D Online Convex Optimization **23**

D.1 Upper Bound. 23

D.2 Lower Bound 24

E Experts Problem **24**

E.1 Upper Bound 25

E.2 Lower Bound 26

F Adversarial Bandits **28**

F.1 Upper Bound 29

F.2 Lower Bound 30

G Reduction from Scale-Free Online Learning **32**

G.1 OCO and Experts 33

G.2 Adversarial Bandits 33

H Reduction from Delayed-Feedback Models **34**

I Naive Doubling Trick **35**

I.1 Adaptive Doubling Trick Sidesteps the Hard Instance 36

AI USE STATEMENT

We used large language models (LLMs) to assist with writing and editing the manuscript, including improving clarity, grammar, and presentation. The authors reviewed and revised the resulting text and take full responsibility for the content of the paper.

A DISCUSSION

A natural open problem is to tighten the gap-dependent regret bound of the stochastic bandits (Theorem 1 and Theorem 4). While our analysis improved the gap-dependent lower bound from $\Omega\left(\alpha \sum_{i:\Delta_i>0} \Delta_i^{-1}\right)$ to $\Omega\left(\alpha^{\frac{3}{2}} \sum_{i:\Delta_i>0} \Delta_i^{-1}\right)$, there remains a $\Theta\left(\alpha^{\frac{1}{2}}\right)$ gap between the upper and lower bound. Appendix C.3 presents an abstract construction showing a limitation of the comparison used in our analysis. Therefore, one concrete starting point is to generalize such an algebraic counterexample to a lower bound of $\Omega\left(\alpha^{\frac{7}{4}} \sum_{i:\Delta_i>0} \Delta_i^{-1}\right)$ with a realizable stochastic bandit instance that works for any algorithm, and then see if such a lower bound can be further improved to $\Omega\left(\alpha^2 \sum_{i:\Delta_i>0} \Delta_i^{-1}\right)$.

Another interesting direction is to extend adversarially batched online learning beyond the four canonical problems considered in this work. Natural directions include reinforcement learning and other bandit models, such as contextual bandits, where feedback constraints and externally imposed update schedules also arise. The complexity measures developed here, including the 2-norm of the batch length sequence and the hardness metric α, provide a natural starting point for understanding how adversarially chosen batch structures affect regret in these broader online learning models.

B UPPER BOUNDS FOR STOCHASTIC BANDITS

B.1 PROOF OF LEMMA 1

Lemma 3 (Hoeffding's inequality)**.** *Let $X_1, \ldots, X_\tau$ be independent random variables such that $X_s \in [a, b]$ for all $s \in [\tau]$. Let $\bar{X} = \frac{1}{\tau}\sum_{s=1}^{\tau} X_s$. Then, for any $\epsilon > 0$,*

$$\Pr\left(\left|\bar{X} - \mathbb{E}[\bar{X}]\right| \geq \epsilon\right) \leq 2\exp\left(-\frac{2\tau\epsilon^2}{(b-a)^2}\right).$$

We now prove Lemma 1. Recall that $j_i = \min\left\{j \in [B] : S_j \geq \frac{4\gamma \ln(NT)}{\Delta_i^2}\right\}$, and the "good event" $\mathcal{E}^{\text{good}}$ is defined as follows: (1) the optimal arm i^* remains active throughout the algorithm; (2) every sub-optimal arm i with $\Delta_i > 0$ and $j_i \leq B$ is eliminated by the end of batch j_i.

Proof of Lemma 1. We first address integer rounding. Under the least-pulled-arm rule, the cumulative pull counts of active arms differ by at most one. Thus, within each block k, the number of pulls of any active arm differs from b_k/n_k by at most two. Summing over blocks, the actual cumulative pull count $C_i(j)$ of an arm surviving through block j satisfies $|C_i(j) - S_j| \leq 2j$. Since merging produces $O(\log T)$ blocks, choosing γ sufficiently large ensures $C_i(j) = \Theta(S_j)$ whenever the elimination threshold is at most one; otherwise, no elimination occurs. We apply concentration inequalities to the actual integer pull counts and suppress these constant-factor rounding effects below.

If Condition (1) fails, then $\hat{\mu}_i - \hat{\mu}_{i^*} > \sqrt{\frac{\gamma \ln(NT)}{S_j}}$ must hold for some $i \neq i^*$ at the end of some batch j. Since the rewards are bounded in $[0, 1]$, $(\hat{\mu}_i - \hat{\mu}_{i^*}) - (\mu_i - \mu_{i^*})$ is an average of S_j independent random variables bounded in the length-4 interval $[-2, 2]$. Thus, by Lemma 3,

$$\begin{aligned}\Pr\left[\hat{\mu}_i - \hat{\mu}_{i^*} > \sqrt{\frac{\gamma \ln(NT)}{S_j}}\right] &\leq \Pr\left[(\hat{\mu}_i - \hat{\mu}_{i^*}) - (\mu_i - \mu_{i^*}) > \sqrt{\frac{\gamma \ln(NT)}{S_j}}\right] \\ &\leq 2\exp\left(-\frac{\gamma \ln(NT)}{16}\right) = \frac{2}{(NT)^{\gamma/16}}.\end{aligned}$$

Applying the union bound to all possible values of $i \in [N]$, $j \in [B]$, and $S_j \in [T]$ gives

$$\Pr\left[\text{Condition (1) fails}\right] \leq \sum_{i=1}^{N}\sum_{t=1}^{T}\sum_{S_j=1}^{T} \frac{2}{(NT)^{\gamma/16}} \leq \frac{1}{2T}$$

for some sufficiently large constant γ.

Next, we analyze the probability that Condition (2) fails to hold. For this to happen, there must be some arm $i \neq i^*$ such that $\hat{\mu}_{i^*} - \hat{\mu}_i \leq \sqrt{\frac{\gamma \ln(NT)}{S_{j_i}}}$ by the end of batch j_i. By definition of $j_i = \min\left\{j \in [B] : S_j \geq \frac{4\gamma \ln(NT)}{\Delta_i^2}\right\}$, we have $\mu_{i^*} - \mu_i = \Delta_i \geq 2\sqrt{\frac{\gamma \ln(NT)}{S_{j_i}}}$. It follows that we have

$$(\mu_{i^*} - \mu_i) - (\hat{\mu}_{i^*} - \hat{\mu}_i) \geq \Delta_i - \sqrt{\frac{\gamma \ln(NT)}{S_{j_i}}} \geq \sqrt{\frac{\gamma \ln(NT)}{S_{j_i}}}.$$

Applying Lemma 3 again shows that the probability for the above to hold is at most $\frac{2}{(NT)^{\gamma/16}}$. Union bounding over all possible values of $i \in [N]$, $j_i \in [B]$, and $S_{j_i} \in [T]$ gives

$$\Pr\left[\text{Condition (2) fails}\right] \leq NT^2 \cdot \frac{2}{(NT)^{\gamma/16}} \leq \frac{1}{2T}$$

for some sufficiently large constant γ. Combining the two probability bounds completes the proof. □

B.2 Proof of Lemma 2

Proof of Lemma 2. We first prove the upper bound under the additional assumption that

$$b_j \geq 2b_{j-1},\ \forall j \in \{2, 3, \ldots, B-1\} \quad \text{and} \quad b_B \leq 2b_{B-1}. \tag{3}$$

Later, we will show that Algorithm 1 effectively ensures such an assumption by merging the blocks.

Assuming Equation (3), it holds for every $j \in [B-1]$ that

$$t_j = b_j + b_{j-1} + \cdots + b_1 \leq b_j \cdot \left(1 + \frac{1}{2} + \cdots + \frac{1}{2^{j-1}}\right) < 2b_j,$$

which further implies $\frac{b_j}{b_{j-1}} \leq \frac{b_j}{t_{j-1}/2} \leq 2\alpha$ for every $j \in \{2, 3, \ldots, B\}$. Moreover, it holds for every $j \in [B-1]$ that $S_j = \Theta(b_j/n_j)$, since we have

$$\frac{b_j}{n_j} \leq S_j = \sum_{k=1}^{j} \frac{b_k}{n_k} \leq \frac{\sum_{k=1}^{j} b_k}{n_j} = \frac{t_j}{n_j} \leq \frac{2b_j}{n_j}.$$

Decomposition of the regret. At the end of each block $j \in [B-1]$, $n_j - n_{j+1}$ sub-optimal arms are removed. Each such arm $i \in A_j \setminus A_{j+1}$ was pulled S_j times and each pull caused a Δ_i regret. We note that, under the least-pulled-arm rule, the cumulative pull counts of active arms differ by at most one. Consequently, the actual cumulative count $C_i(j)$ of an arm surviving through block j satisfies $|C_i(j) - S_j| \leq 2j$. Since merging produces $B = \mathcal{O}(\log T)$ blocks, rounding affects concentration bounds only by constant factors at sample sizes where elimination can occur. Its contribution to regret therefore is $\mathcal{O}(\log T \sum_{i:\Delta_i > 0} \Delta_i)$, which is absorbed by the stated bounds.

Conditioned on event $\mathcal{E}^{\text{good}}$, every sub-optimal arm i is removed before or at the end of block j_i. Then, $i \in A_j$ implies that

$$\Delta_i \leq \sqrt{\frac{\gamma \ln(NT)}{S_{j-1}}}.$$

The above also holds for the last block (i.e., $j = B$), if we define $A_{B+1} \coloneqq \{i \in [N] : \Delta_i = 0\}$ as the set of optimal arms and $n_{B+1} = |A_{B+1}|$.

Summing over all B blocks, we obtain the following upper bound on the regret:

$$\sum_{j=1}^{B} (n_j - n_{j+1}) \cdot S_j \cdot \sqrt{\frac{\gamma \ln(NT)}{S_{j-1}}} = \mathcal{O}(\sqrt{\log(NT)}) \cdot \sum_{j=1}^{B} (n_j - n_{j+1}) \cdot S_j \cdot \sqrt{\frac{1}{S_{j-1}}}.$$

Recall that $S_j = \Theta(b_j/n_j)$ for every $j \in [B-1]$. Therefore, using the shorthand

$$\text{LHS}_j \coloneqq \begin{cases} (n_j - n_{j+1}) \cdot \frac{b_j}{n_j} \cdot \sqrt{\frac{n_{j-1}}{b_{j-1}}}, & j \in [B-1], \\ (n_j - n_{j+1}) \cdot S_j \cdot \sqrt{\frac{n_{j-1}}{b_{j-1}}}, & j = B, \end{cases}$$

the regret is upper bounded by $\mathcal{O}(\sqrt{\log(NT)}) \cdot \sum_{j=1}^{B} \text{LHS}_j$.

Decomposition of the regret bound. Recall that our goal is to upper bound the regret by

$$\alpha^2 \ln(NT) \cdot \sum_{i:\Delta_i>0} \frac{1}{\Delta_i} \geq \alpha^2 \ln(NT) \sum_{j=1}^{B} (n_j - n_{j+1}) \sqrt{\frac{S_{j-1}}{\gamma \log(NT)}}$$
$$\geq \Omega(\sqrt{\log(NT)}) \cdot \alpha^2 \sum_{j=1}^{B} (n_j - n_{j+1}) \sqrt{\frac{b_{j-1}}{n_{j-1}}}.$$

Therefore, it suffices to show that $\sum_{j=1}^{B} \mathrm{LHS}_j \leq \mathcal{O}(\alpha^2) \cdot \sum_{j=1}^{B} \mathrm{RHS}_j$, where

$$\mathrm{RHS}_j \coloneqq (n_j - n_{j+1}) \sqrt{\frac{b_{j-1}}{n_{j-1}}}.$$

Division of blocks. Towards proving $\sum_{j=1}^{B} \mathrm{LHS}_j \leq \mathcal{O}(\alpha^2) \cdot \sum_{j=1}^{B} \mathrm{RHS}_j$, we set $C = 3$ and partition the blocks $1, 2, \ldots, B$ into two sets:

$$J_1 \coloneqq \left\{ j \in [B] : n_j \geq \frac{n_{j-1}}{C\alpha} \right\} \quad \text{and} \quad J_2 \coloneqq \left\{ j \in [B] : n_j < \frac{n_{j-1}}{C\alpha} \right\}.$$

For notational convenience, we extend the initialization convention $t_0 = 1$ by setting $b_0 \coloneqq 1$, $n_0 \coloneqq n_1 = N$, and $S_0 \coloneqq b_0/n_0$. These analysis-only boundary values encode the trivial first-block bounds $\Delta_i \leq 1 \leq \sqrt{\gamma \log(NT)/S_0}$ and $b_1 \leq \alpha b_0$, while ensuring the first block belongs to the J_1.

Blocks in J_1. For any $j \in J_1$, the condition $n_j \geq \frac{n_{j-1}}{C\alpha}$ together with the bound $b_j \leq 2\alpha \cdot b_{j-1}$ gives

$$\mathrm{LHS}_j \leq (n_j - n_{j+1}) \cdot \frac{2\alpha \cdot b_{j-1}}{n_{j-1}/(C\alpha)} \cdot \sqrt{\frac{n_{j-1}}{b_{j-1}}} \leq \mathcal{O}(\alpha^2) \cdot \mathrm{RHS}_j,$$

which further implies

$$\sum_{j \in J_1} \mathrm{LHS}_j \leq \mathcal{O}(\alpha^2) \cdot \sum_{j \in J_1} \mathrm{RHS}_j \leq \mathcal{O}(\alpha^2) \cdot \sum_{j=1}^{B} \mathrm{RHS}_j. \tag{4}$$

Blocks in J_2. To control the contribution from J_2, consider an index pair (j_0, j_1) that specify a maximal contiguous subsequence of indices in J_2. In other words, $j_0, j_0 + 1, \ldots, j_1 \in J_2$ while $j_0 - 1, j_1 + 1 \notin J_2$. We will show that $\sum_{j=j_0}^{j_1} \mathrm{LHS}_j \leq \mathcal{O}(\alpha^2) \cdot \mathrm{RHS}_{j_0-1}$.

When $j \neq B$, we have

$$\mathrm{LHS}_j = (n_j - n_{j+1}) \cdot \frac{b_j}{n_j} \sqrt{\frac{n_{j-1}}{b_{j-1}}} \leq b_j \sqrt{\frac{n_{j-1}}{b_{j-1}}} \leq \mathcal{O}(\alpha) \sqrt{n_{j-1} b_{j-1}},$$

where the last step applies $b_j \leq 2\alpha \cdot b_{j-1}$. For $j = B$, note that $S_{B-1} \leq 2b_{B-1}/n_{B-1}$ and the assumption $b_B \leq 2b_{B-1}$ (from Equation (3)) together imply

$$S_B = S_{B-1} + \frac{b_B}{n_B} \leq \frac{2b_{B-1}}{n_{B-1}} + \frac{2b_{B-1}}{n_B} \leq \frac{4b_{B-1}}{n_B}.$$

It follows that $\mathrm{LHS}_B = (n_B - n_{B+1}) \cdot S_B \sqrt{\frac{n_{B-1}}{b_{B-1}}} \leq 4\sqrt{n_{B-1} b_{B-1}}$. Therefore, we have $\mathrm{LHS}_j \leq \mathcal{O}(\alpha)\sqrt{n_{j-1} b_{j-1}}$ for every $j \in \{j_0, j_0 + 1, \ldots, j_1\}$.

Since $b_j \leq 2\alpha \cdot b_{j-1}$ and $n_j < \frac{n_{j-1}}{C\alpha}$ hold for every $j \in \{j_0, j_0 + 1, \ldots, j_1\}$, a simple induction gives $b_{j-1} \leq (2\alpha)^{j-j_0} \cdot b_{j_0-1}$ and $n_{j-1} \leq \frac{n_{j_0-1}}{(C\alpha)^{j-j_0}}$. It follows that

$$\sum_{j=j_0}^{j_1} \mathrm{LHS}_j \leq \mathcal{O}(\alpha) \cdot \sum_{j=j_0}^{j_1} \sqrt{b_{j-1} n_{j-1}} \qquad (\mathrm{LHS}_j \leq \mathcal{O}(\alpha)\sqrt{n_{j-1} b_{j-1}})$$
$$\leq \mathcal{O}(\alpha) \cdot \sum_{j=j_0}^{j_1} \sqrt{(2\alpha)^{j-j_0} \cdot b_{j_0-1} \cdot \frac{n_{j_0-1}}{(C\alpha)^{j-j_0}}}$$
$$\leq \mathcal{O}(\alpha) \cdot \sqrt{b_{j_0-1} n_{j_0-1}} \cdot \sum_{t=0}^{j_1-j_0} \sqrt{(2/3)^t} = \mathcal{O}(\alpha) \cdot \sqrt{b_{j_0-1} n_{j_0-1}}. \qquad (C = 3)$$

For the RHS_{j_0-1} term, note that the assumption $j_0 \in J_2$ implies $n_{j_0} < \frac{n_{j_0-1}}{C\alpha}$. Moreover, since $j_0 - 1 \in J_1$, we have $n_{j_0-1} \geq \frac{n_{j_0-2}}{C\alpha}$ and equivalently $n_{j_0-2} \leq C\alpha \cdot n_{j_0-1}$. We then have

$$\begin{aligned}
\mathrm{RHS}_{j_0-1} &= (n_{j_0-1} - n_{j_0})\sqrt{\frac{b_{j_0-2}}{n_{j_0-2}}} \\
&> \left(n_{j_0-1} - \frac{n_{j_0-1}}{C\alpha}\right)\sqrt{\frac{b_{j_0-2}}{C\alpha \cdot n_{j_0-1}}} \qquad (n_{j_0} < \tfrac{n_{j_0-1}}{C\alpha} \text{ and } n_{j_0-2} \leq C\alpha \cdot n_{j_0-1}) \\
&\geq \Omega(1) \cdot \sqrt{\frac{b_{j_0-2} n_{j_0-1}}{\alpha}}.
\end{aligned}$$

Combining the two bounds above and applying $b_{j_0-1} \leq 2\alpha \cdot b_{j_0-2}$, we conclude that

$$\sum_{j=j_0}^{j_1} \mathrm{LHS}_j \leq \mathcal{O}(\alpha) \cdot \sqrt{b_{j_0-1} n_{j_0-1}} \leq \mathcal{O}(\alpha) \cdot \sqrt{\alpha \cdot b_{j_0-2} n_{j_0-1}} \leq \mathcal{O}(\alpha^2) \cdot \mathrm{RHS}_{j_0-1}.$$

Summing over all contiguous intervals in J_2 (i.e., all pairs (j_0, j_1)) gives

$$\sum_{j \in J_2} \mathrm{LHS}_j \leq \mathcal{O}(\alpha^2) \cdot \sum_{j=1}^{B} \mathrm{RHS}_j. \tag{5}$$

Combining Equations (4) and (5) completes the proof under the assumption in Equation (3).

General block structures. Finally, we note that Algorithm 1 effectively merges the blocks to form a new batch structure, denoted as $\mathcal{B}'$, that ensures $b'_j \geq 2b'_{j-1}$ except for the last block. Since the assumption in Equation (3) holds, our previous analysis bounds the regret by $\mathcal{O}\left([\alpha(\mathcal{B}')]^2 \log(NT) \sum_{i:\Delta_i>0} \frac{1}{\Delta_i}\right)$. It remains to show that the merging does not significantly increase the difficulty parameter α of the batch structure. Concretely, we will prove $\alpha(\mathcal{B}') \leq 4\alpha(\mathcal{B}) + 2$.

Consider an arbitrary block b'_j in the new batch structure $\mathcal{B}'$. If b'_j is the final incomplete merged block, we have $b'_j < 2b'_{j-1} \leq 2t'_{j-1}$. Otherwise, b'_j is a completed merged block, including the case where its completion coincides with the end of the horizon. In this boundary case, the final-block condition in Equation (3) is replaced by $b_B \geq 2b_{B-1}$; all merged blocks then satisfy the geometric-growth condition, so $S_B = \Theta(b_B/n_B)$ and the regular-block estimates in the preceding regret analysis apply to the final block as well. For a completed merged block b'_j, let r be the index of the last original block (in $\mathcal{B}$) that forms b'_j. By construction, we have $b'_j \leq 2b'_{j-1} + b_r$; otherwise, block b'_j should be completed after merging block b_{r-1}. By definition, we have $b_r/t_{r-1} \leq \alpha(\mathcal{B})$. Furthermore,

$$t_{r-1} \leq t'_{j-1} + 2b'_{j-1} = b'_1 + b'_2 + \cdots + b'_{j-1} + 2b'_{j-1} \leq 4b'_{j-1}.$$

Then, we have

$$b'_j \leq 2b'_{j-1} + \alpha(\mathcal{B}) \cdot t_{r-1} \leq 2b'_{j-1} + \alpha(\mathcal{B}) \cdot 4b'_{j-1} \leq (4\alpha(\mathcal{B}) + 2) \cdot t'_{j-1}.$$

Therefore, we have $\alpha(\mathcal{B}') = \max_j \frac{b'_j}{t'_{j-1}} \leq 4\alpha(\mathcal{B}) + 2$, which completes the proof. □

B.3 Extension of Existing Bounds for Arbitrary Batch Structure

Proposition 3. *For the adversarially batched stochastic bandits with N arms and arbitrary batch structure $\mathcal{B}$ characterized by the hardness metric α, there is an algorithm with an expected regret of*

$$\mathcal{O}\left(\alpha \log N \log(NT) \frac{N}{\Delta_{\min}}\right),$$

where $\Delta_{\min} = \min_{i:\Delta_i>0} \Delta_i$.

Proof. We follow the argument used in the proof of Theorem 2. Throughout the following argument, we work on the event $\mathcal{E}^{\text{good}}$. Let τ_i and σ_i be defined as in that proof: τ_i is the number of blocks in which arm i remains active, and σ_i is its position in the elimination order, with ties broken arbitrarily and surviving arms listed last. Then $\tau_i \le j_i$, and

$$\mathrm{Reg}_N^{(t_B)} = \sum_{i:\Delta_i>0} \Delta_i \cdot S_{\tau_i}.$$

Whenever arm i remains active, there are at least $N - \sigma_i + 1$ active arms. Hence,

$$S_{\tau_i} = \sum_{k=1}^{\tau_i} \frac{b_k}{n_k} \le \sum_{k=1}^{\tau_i} \frac{b_k}{N-\sigma_i+1} = \frac{t_{\tau_i}}{N-\sigma_i+1} \le \frac{(1+\alpha)t_{\tau_i-1}}{N-\sigma_i+1}.$$

Since $\tau_i \le j_i$, the definition of j_i gives, for $\tau_i \ge 2$,

$$\frac{t_{\tau_i-1}}{N} \le S_{\tau_i-1} < \frac{4\gamma \log(NT)}{\Delta_i^2}.$$

Consequently,

$$t_{\tau_i-1} \le \frac{4\gamma \log(NT) N}{\Delta_i^2}.$$

The same bound holds for $\tau_i = 1$, using $t_0 = 1$ and the choice of γ in Lemma 1. Moreover,

$$\sum_{i:\Delta_i>0} \frac{1}{N-\sigma_i+1} \le \sum_{n=1}^{N} \frac{1}{n} \le 1 + \ln N,$$

since $(\sigma_i)_{i\in[N]}$ is a permutation of $[N]$. Combining these bounds, we have

$$\begin{aligned}
\mathrm{Reg}_N^{(t_B)} &= \sum_{i:\Delta_i>0} \Delta_i \cdot S_{\tau_i} \\
&\le \sum_{i:\Delta_i>0} \Delta_i \frac{(1+\alpha)t_{\tau_i-1}}{N-\sigma_i+1} \\
&\le \sum_{i:\Delta_i>0} \Delta_i \frac{4\gamma \log(NT) N}{\Delta_i^2} \frac{1+\alpha}{N-\sigma_i+1} \\
&\le 4\gamma(1+\alpha)(1+\ln N)\log(NT)\frac{N}{\Delta_{\min}}.
\end{aligned}$$

By Lemma 1, $\Pr(\mathcal{E}^{\text{good}}) \ge 1 - 1/T$. Since the regret is at most $T = t_B$, the contribution of bad events to the expected regret is at most 1. Using $\alpha \ge 1$ and $N \ge 2$, we obtain

$$\mathbb{E}\left[\mathrm{Reg}_N^{(t_B)}\right] \le 4\gamma(1+\alpha)(1+\ln N)\log(NT)\frac{N}{\Delta_{\min}} + 1 = \mathcal{O}\left(\alpha \log N \log(NT)\frac{N}{\Delta_{\min}}\right),$$

which completes the proof. □

C Lower Bounds for Stochastic Bandits

C.1 Lower Bounds Extended from Gao et al. (2019)

Gap-Dependent Lower Bound. We first give a bounded-reward analogue of (Gao et al., 2019, Lemma 2). The proof follows the same multiple-hypothesis testing argument, but uses Bernoulli reward distributions instead of Gaussian ones.

Lemma 4. *For every $N \ge 2$ and every $0 < \Delta \le 1/8$, and every batched policy, there exists an N-armed stochastic bandit instance with Bernoulli rewards, with all means in $[0,1]$ and all nonzero gaps belonging to $\{\Delta, 2\Delta\}$, such that every batched policy satisfies*

$$\mathbb{E}[\mathrm{Reg}] \ge \frac{\Delta}{4} \sum_{j=1}^{B} b_j \exp\left(-C_0 \cdot \frac{t_{j-1}\Delta^2}{N-1}\right),$$

where $C_0 > 0$ is a universal numerical constant.

Proof. Fix $\Delta \le 1/8$. We construct N candidate bandit instances $P_1, \dots, P_N$. Under P_1, the arm reward distributions are

$$P_1 = \mathrm{Bern}\left(\frac{1}{2} + \Delta\right) \otimes \mathrm{Bern}\left(\frac{1}{2}\right)^{\otimes(N-1)}.$$

For every $i \in \{2, \dots, N\}$, define

$$P_i = \mathrm{Bern}\left(\frac{1}{2} + \Delta\right) \otimes \mathrm{Bern}\left(\frac{1}{2}\right) \otimes \cdots \otimes \mathrm{Bern}\left(\frac{1}{2} + 2\Delta\right) \otimes \cdots \otimes \mathrm{Bern}\left(\frac{1}{2}\right),$$

where the i-th arm has a mean of $1/2 + 2\Delta$. As $\Delta \le 1/8$,

$$1/2 \le 1/2 + \Delta \le 1/2 + 2\Delta \le 3/4,$$

so every reward distribution is supported on $\{0, 1\}$ and has mean in $[0, 1]$. Moreover, under P_1, arm 1 is optimal, and every other arm has gap exactly Δ. Under P_i for each $i \ge 2$, arm i is optimal; arm 1 has gap Δ, while every arm $k \notin \{1, i\}$ has gap 2Δ. Hence every nonzero gap belongs to $\{\Delta, 2\Delta\}$, and in particular is at most $2\Delta \le 1/4$.

For a round t belonging to batch j, let P_i^t denote the distribution, under instance P_i, of all information available to the learner when the action at time t is selected. Since no feedback from the current batch is available before the batch ends, P_i^t depends only on observations from the first $j-1$ batches. Every incorrect arm choice incurs instantaneous regret of at least Δ under each P_i. Therefore,

$$\sup_{i \in [N]} \mathbb{E}_i[\mathrm{Reg}] \ge \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_i[\mathrm{Reg}] \ge \Delta \sum_{t=1}^{T} \frac{1}{N} \sum_{i=1}^{N} P_i^t(i_t \ne i).$$

We apply the same multiple-hypothesis testing inequality used in Gao et al. (2019) with the star tree having edges $(1,2), (1,3), \dots, (1,N)$. For every t,

$$\frac{1}{N} \sum_{i=1}^{N} P_i^t(i_t \ne i) \ge \frac{1}{2N} \sum_{i=2}^{N} \exp(-D_{\mathrm{KL}}(P_1^t \| P_i^t)).$$

The instances P_1 and P_i differ only in the reward distribution of arm i: under P_1 it is $\mathrm{Bern}(1/2)$, whereas under P_i it is $\mathrm{Bern}(1/2 + 2\Delta)$. Hence, the standard KL chain rule for bandit observations gives

$$D_{\mathrm{KL}}(P_1^t \| P_i^t) = \mathbb{E}_1[t_i]\, d_\Delta, \tag{22}$$

where t_i is the number of times arm i has been pulled before the current batch and

$$\begin{aligned} d_\Delta &= D_{\mathrm{KL}}\left(\mathrm{Bern}\left(1/2\right) \| \mathrm{Bern}\left(1/2 + 2\Delta\right)\right) \\ &= \frac{1}{2} \log \frac{1/2}{1/2 + 2\Delta} + \frac{1}{2} \log \frac{1/2}{1/2 - 2\Delta} \\ &= -\frac{1}{2} \log(1 - 16\Delta^2). \end{aligned}$$

Since $\Delta \le 1/8$, we have $16\Delta^2 \le 1/4$. Using

$$-\log(1 - x) \le \frac{x}{1 - x},$$

where $0 \le x < 1$, we obtain

$$d_\Delta \le \frac{1}{2} \frac{16\Delta^2}{1 - 16\Delta^2} \le C_0 \Delta^2.$$

For t in batch j, all observations available to the learner come from the first t_{j-1} rounds. Consequently,

$$\sum_{i=2}^{N} \mathbb{E}_1[t_i] \le t_{j-1}.$$

Together, by Jensen's inequality, we have

$$\begin{aligned}\frac{1}{N}\sum_{i=1}^{N} P_i^t(i_t \neq i) &\geq \frac{1}{2N}\sum_{i=2}^{N}\exp(-d_\Delta \mathbb{E}_1[t_i]) \\ &\geq \frac{N-1}{2N}\exp\left(-\frac{d_\Delta}{N-1}\sum_{i=2}^{N}\mathbb{E}_1[t_i]\right) \\ &\geq \frac{1}{4}\exp\left(-C_0\frac{t_{j-1}\Delta^2}{N-1}\right)\end{aligned}$$

Summing over all b_j rounds in batch j gives

$$\sup_{i\in[N]} \mathbb{E}_i[\mathrm{Reg}] \geq \frac{\Delta}{4}\sum_{j=1}^{B} b_j \exp\left(-C_0\frac{t_{j-1}\Delta^2}{N-1}\right),$$

which marks the end of the proof. □

Now we are ready to prove Proposition 1.

Proof of Proposition 1. We consider a broad family of hard instances that satisfy $t_{j^*-1} \geq CN$. Taking advantage of Lemma 4, we have

$$\begin{aligned}\Delta\cdot\mathbb{E}\left[\mathrm{Reg}_N^{(B)}\right] &\geq \Delta^2\cdot\sum_{j=1}^{B}\frac{t_j - t_{j-1}}{4}\exp\left(-\frac{C_0 t_{j-1}\Delta^2}{N-1}\right) \\ &= \Delta^2\cdot\sum_{j=1}^{B}\frac{b_j}{4}\exp\left(-\frac{C_0 t_{j-1}\Delta^2}{N-1}\right) \\ &\geq \Delta^2\cdot\frac{b_{j^*}}{4}\exp\left(-\frac{C_0 t_{j^*-1}\Delta^2}{N-1}\right),\end{aligned}$$

where we set j^* to be the block index chosen by α, i.e., $\alpha = \max_j \frac{b_j}{t_{j-1}} = \frac{b_{j^*}}{t_{j^*-1}}$. Next, we consider a case where every sub-optimal arm has the smallest gap $\Delta = \Delta_{i:i\neq i^*} = \sqrt{\frac{N-1}{C_0 t_{j^*-1}}}$, which leads to

$$\begin{aligned}\Delta\cdot\mathbb{E}\left[\mathrm{Reg}_N^{(B)}\right] &\geq \Delta^2\cdot\frac{b_{j^*}}{4}\exp\left(-\frac{C_0 t_{j^*-1}\Delta^2}{N-1}\right) \\ &\geq \frac{N-1}{t_{j^*-1}}\frac{b_{j^*}}{4}\exp\left(-\frac{C_0 t_{j^*-1}}{N-1}\frac{N-1}{t_{j^*-1}}\right) \\ &= \Omega(1)\cdot\frac{b_{j^*}}{t_{j^*-1}}N = \Omega(1)\cdot\alpha N.\end{aligned}$$

We note that, based on the hard instance construction ($t_{j^*-1} \geq CN$), $\Delta = \sqrt{\frac{N-1}{C_0 t_{j^*-1}}} \leq 1/8$ is a valid bounded-reward gap. Finally, we have

$$\Delta\cdot\mathbb{E}\left[\mathrm{Reg}_N^{(B)}\right] \geq \Omega(1)\cdot N\alpha \iff \mathbb{E}\left[\mathrm{Reg}_N^{(B)}\right] \geq \Omega(1)\cdot\alpha\frac{N}{\Delta} = \Omega\left(\alpha\sum_{i\neq i^*}\Delta_i^{-1}\right),$$

which completes the proof. □

Gap-Independent Lower Bound. Similar to the proof of Proposition 1, we consider a family of hard instances that satisfy $t_{j^*-1} \geq CN$ and $t_{j^*-1} \geq \Omega(T/\alpha)$, where, intuitively speaking, the

second part asks that the hard instance's largest relative batch must occur late enough. Based on the proof of Proposition 1, we have

$$\begin{aligned}\mathbb{E}\left[\mathrm{Reg}_N^{(B)}\right] &\geq \Delta \cdot \sum_{j=1}^{B} \frac{t_j - t_{j-1}}{4} \exp\left(-\frac{C_0 t_{j-1}\Delta^2}{N-1}\right) \\ &= \Delta \cdot \sum_{j=1}^{B} \frac{b_j}{4} \exp\left(-\frac{C_0 t_{j-1}\Delta^2}{N-1}\right) \\ &\geq \frac{\Delta}{4} b_{j^*} \exp\left(-\frac{C_0 t_{j^*-1}\Delta^2}{N-1}\right) \\ &= \frac{e^{-1}}{4\sqrt{C_0}} b_{j^*} \sqrt{\frac{N-1}{t_{j^*-1}}}.\end{aligned}$$

Recall $b_{j^*} = \Theta(\alpha t_{j^*-1})$ and the condition of $t_{j^*-1} = \Omega\left(\frac{T}{\alpha}\right)$, we have

$$\mathbb{E}\left[\mathrm{Reg}_N^{(B)}\right] \geq \Omega(1) \cdot \alpha\sqrt{N t_{j^*-1}} \geq \Omega(1) \cdot \sqrt{\alpha N T}.$$

C.2 An Improved Gap-Dependent Lower Bound

Proof. Set $\delta = \frac{1}{16\sqrt{N}}$. For every unordered pair $\{u, v\} \subseteq [N]$, define two instances $\nu^+_{u,v}$ and $\nu^-_{u,v}$. Under $\nu^+_{u,v}$,

$$\mu_u = \frac{1}{2} + \frac{\delta}{2},\ \mu_v = \frac{1}{2} - \frac{\delta}{2},$$

while every arm $i \notin \{u, v\}$ has deterministic reward 0. Under $\nu^-_{u,v}$, the means of u and v are exchanged, i.e.,

$$\mu_u = \frac{1}{2} - \frac{\delta}{2},\ \mu_v = \frac{1}{2} + \frac{\delta}{2}.$$

The two special arms have Bernoulli rewards.

Our hard distribution samples $\{U, V\} \sim \mathrm{Unif}\left(\binom{[N]}{2}\right)$, and independently samples $S \in \{+, -\}$ uniformly. The resulting instance is $\nu^S_{U,V}$. The batch structure is set to be $\mathcal{B} = \{N, N^2, N^3\}$, which gives $\alpha = \max\left\{N, \frac{N^2}{N}, \frac{N^3}{N+N^2}\right\} = \Theta(N)$. We first focus on the deterministic algorithm and extend to the randomized algorithm at the end.

The First Batch. Recall that the first batch has N rounds without any prior feedback. Let m_i be the number of times that arm i has been pulled in the first batch; then $\sum_{i=1}^{N} m_i = N$. Define the set of arms that are not heavily sampled (less than 3 times) in the first batch as $\mathcal{L} = \{i \in [N] : m_i \leq 2\}$. Since every arm outside $\mathcal{L}$ is pulled at least three times, we have

$$3(N - |\mathcal{L}|) \leq \sum_{i=1}^{N} m_i = N,$$

and hence $|\mathcal{L}| \geq \frac{2N}{3}$. Let E be the event that all rewards received in the first batch are 0, and define $F = E \cap \{U, V \in \mathcal{L}\}$. Recall that $\mu_i \leq \frac{1}{2} + \frac{\delta}{2} < \frac{1}{2} + \frac{1}{4} = \frac{3}{4}$ as $\delta = \frac{1}{16\sqrt{N}} < 1/2$. For any $u, v \in \mathcal{L}$ and either $S = +$ or $S = -$, both special arms have means smaller than $3/4$. Therefore,

$$\Pr\left(r_i = 0 \mid i \in \{u, v\}\right) \geq \frac{1}{4}.$$

Moreover, $m_u + m_v \leq 4$. Thus,

$$\Pr\left(E \mid U = u, V = v, S\right) \geq 4^{-(m_u + m_v)} \geq 4^{-4} = \Omega(1).$$

Consequently,

$$\Pr(F) = \Pr(U, V \in \mathcal{L}) \cdot \Pr(E \mid U, V \in \mathcal{L}) \geq \frac{\binom{|\mathcal{L}|}{2}}{\binom{N}{2}} 4^{-4} = \Omega(1).$$

The Second Batch. Conditioned on event E, the learner observes the same all-zero history regardless of the underlying pair and sign. Let q_i be the number of pulls of arm i in the second batch. Since the algorithm is deterministic, $\sum_{i=1}^{N} q_i = N^2$, and the values $q_1, \ldots, q_N$ are fixed conditioned on E. For $u, v \in \mathcal{L}$, we denote $p_{u,v} = \Pr(E \mid U = u, V = v)$, where the probability averages over the random sign S. By the argument above, $p_{u,v} \geq 4^{-4}$. We check the expected sample complexity of special arms in the second batch conditioning on F:

$$\mathbb{E}[q_U + q_V | F] = \frac{\sum_{u<v;\, u,v\in\mathcal{L}} p_{u,v} \cdot (q_u + q_v)}{\sum_{u<v;\, u,v\in\mathcal{L}} p_{u,v}} \leq \frac{\sum_{u<v;\, u,v\in\mathcal{L}} (q_u + q_v)}{\sum_{u<v;\, u,v\in\mathcal{L}} p_{u,v}}$$

as $p_{u,v} \leq 1$. Moreover, recall that $p_{u,v} \geq 4^{-4}$ and note that every q_i will appear in $(q_u + q_v)$ with a total number of $|\mathcal{L}| - 1$ times. Thus,

$$\frac{\sum_{u<v;\, u,v\in\mathcal{L}} (q_u + q_v)}{\sum_{u<v;\, u,v\in\mathcal{L}} p_{u,v}} \leq \frac{(|\mathcal{L}| - 1)\sum_{i\in\mathcal{L}} q_i}{4^{-4}\binom{|\mathcal{L}|}{2}} \leq \frac{(|\mathcal{L}| - 1)N^2}{4^{-4}\binom{|\mathcal{L}|}{2}} = \mathcal{O}(N),$$

where the last step follows from $|\mathcal{L}| = \Omega(N)$.

The Third Batch. Next, conditioned on event F, we fix one (u, v) pair with $u, v \in \mathcal{L}$ form a binary hypothesis testing:

$$H_+ : \mu_u = \frac{1}{2} + \frac{\delta}{2};\ \mu_v = \frac{1}{2} - \frac{\delta}{2},$$

and

$$H_- : \mu_u = \frac{1}{2} - \frac{\delta}{2};\ \mu_v = \frac{1}{2} + \frac{\delta}{2}.$$

We first note that conditioning on the first-batch event E does not make the two hypotheses equally likely in general. Nevertheless, since $u, v \in \mathcal{L}$, their posterior probabilities remain bounded away from zero. Indeed,

$$\frac{\Pr(E \mid U = u, V = v, S = +)}{\Pr(E \mid U = u, V = v, S = -)} = \frac{\left(\frac{1-\delta}{2}\right)^{m_u}\left(\frac{1+\delta}{2}\right)^{m_v}}{\left(\frac{1+\delta}{2}\right)^{m_u}\left(\frac{1-\delta}{2}\right)^{m_v}} = \left(\frac{1-\delta}{1+\delta}\right)^{m_u - m_v}.$$

Since $m_u, m_v \leq 2$ and $\delta \leq 1/16$, this likelihood ratio is bounded above and below by universal constants. In other words, there exists a universal constant $c_0 > 0$ such that

$$\Pr(S = +|E, U = u, V = v) \geq c_0,\ \ \Pr(S = -|E, U = u, V = v) \geq c_0.$$

Let P_+ and P_- denote the distribution of the second-batch observations under H_+ and H_- conditioned on the first-batch history E. Since the second-batch allocation is fixed to be $q_1, \ldots, q_N$, the KL divergence between the distributions induced by the two hypotheses can be formulated as

$$D_{\mathrm{KL}}(P_+ \| P_-) = q_u \cdot \mathrm{kl}\left(\frac{1+\delta}{2}, \frac{1-\delta}{2}\right) + q_v \cdot \mathrm{kl}\left(\frac{1-\delta}{2}, \frac{1+\delta}{2}\right),$$

where $\mathrm{kl}(p\|q)$ denotes the binary relative entropy. It is easy to see that

$$\mathrm{kl}\left(\frac{1+\delta}{2}, \frac{1-\delta}{2}\right) = \mathrm{kl}\left(\frac{1-\delta}{2}, \frac{1+\delta}{2}\right) = \delta \log \frac{1+\delta}{1-\delta} \leq \delta \cdot 4\delta = 4\delta^2.$$

Thus, $\mathrm{KL}(P_+\|P_-) \leq 4\delta^2(q_u + q_v)$. Let X_i denote the number of times that arm i is pulled in the third batch. Since the learner cannot update its policy inside the third batch, the vector $X_1, \ldots, X_N$ denote the number of times that arm i is pulled in the third batch. Since the learner cannot update its policy inside the third batch, the vector $(X_1, \ldots, X_N)$ is determined by the observations available at the end of the second batch. Define the binary test

$$\phi = \begin{cases} +, & X_u \geq X_v, \\ -, & X_u < X_v. \end{cases}$$

By the Bretagnolle-Huber inequality, we have

$$\Pr_+(\phi = -) + \Pr_-(\phi = +) \geq \frac{1}{2} e^{-4\delta^2(q_u + q_v)}.$$

Under such false events where suboptimal arms that have a gap of at least δ have been pulled at least $b_3/2$ times, the regret is at least $\frac{N^3\delta}{2}$. Let $\pi_+ = \Pr(S = +|E, U = u, V = v)$ and $\pi_- = \Pr(S = -|E, U = u, V = v)$. Since $\pi_+, \pi_- \geq c_0$, we have

$$\begin{aligned}\mathbb{E}[R_3|F, U = u, V = v] &\geq \frac{N^3\delta}{2} \cdot \left(\pi_+ \Pr_+(\phi = -) + \pi_- \Pr_-(\phi = +)\right) \\ &\geq \frac{c_0 N^3 \delta}{4} \exp(-4\delta^2(q_u + q_v)).\end{aligned}$$

Taking expectation over (U, V) conditioned on F and using Jensen's inequality gives

$$\begin{aligned}\mathbb{E}[R_3|F] &\geq \frac{c_0 N^3 \delta}{4} \mathbb{E}[\exp(-4\delta^2(q_U + q_V))|F] \\ &\geq \frac{c_0 N^3 \delta}{4} \exp(-4\delta^2 \mathbb{E}[(q_U + q_V)|F])\end{aligned}$$

Recall that $\mathbb{E}[q_U + q_V|F] = \mathcal{O}(N)$ and $\delta = \frac{1}{16\sqrt{N}}$. Therefore, $\delta^2\mathbb{E}[(q_U + q_V)|F] = \mathcal{O}(1)$, and

$$\mathbb{E}[R_3|F] = \Omega\left(N^3 \frac{1}{\sqrt{N}}\right) = \Omega\left(N^{\frac{5}{2}}\right).$$

Finally, recall that $\Pr(F) = \Omega(1)$, we have

$$\mathbb{E}[\mathrm{Reg}] \geq \Pr(F)\mathbb{E}[R_3|F] = \Omega(N^{5/2}) = \Omega(\alpha^{\frac{3}{2}} \sum_{i:\Delta_i > 0} \Delta_i^{-1}),$$

where the last step follows the fact that one special arm has a gap δ in every instance in support of the hard distribution, while the remaining $N - 2$ non-special arms have gaps that are bounded by universal constants, which gives $\sum_{i:\Delta_i>0} \frac{1}{\Delta_i} = \frac{1}{\delta} + \Theta(N) = \Theta(N)$.

Randomized Algorithms. Let one randomized algorithm A's internal randomness be w. Fixing w makes the randomized algorithm a deterministic algorithm, denoted as A_w, where we have shown that for any deterministic algorithm, there exists a worst instance from a hard prior Π that makes $\mathbb{E}_{\nu\sim\Pi}[\mathrm{Reg}] = \Omega(N^{5/2})$. Therefore,

$$\mathbb{E}_w\mathbb{E}_\nu[\mathrm{Reg}] = \mathbb{E}_\nu\mathbb{E}_w[\mathrm{Reg}] = \Omega(N^{5/2}) = \Omega\left(\alpha^{\frac{3}{2}} \sum_{i:\Delta_i>0} \frac{1}{\Delta_i}\right),$$

which completes the proof. □

C.3 An Abstract Counterexample of the Current Gap-Dependent Analysis

We examine the comparison step used in the analysis of Algorithm 1. Specifically, we construct an abstract instance with sequences of batch lengths and active-set sizes for which this comparison requires a factor of $\Omega(\alpha^{7/4})$.

Proposition 4. *There exists a family of abstract sequences $(b_j)_{j=1}^B$ and non-increasing positive integers $(n_j)_{j=1}^{B+1}$ such that, writing*

$$S_j := \sum_{k=1}^{j} \frac{b_k}{n_k}, \qquad \mathrm{LHS}_j := (n_j - n_{j+1})\frac{S_j}{\sqrt{S_{j-1}}}, \ \mathrm{RHS}_j := (n_j - n_{j+1})\sqrt{S_{j-1}} \ (j = 2, \ldots, B),$$

we have

$$\frac{\sum_{j=2}^B \mathrm{LHS}_j}{\sum_{j=2}^B \mathrm{RHS}_j} = \Theta(\alpha^{7/4}),$$

where α is the hardness metric of $(b_j)_{j=1}^B$.

Proof. Let N range over eighth powers of integers, and consider a game of five batches with lengths $b_j = N^{\frac{j}{2}}$, i.e., $b_1 = N^{\frac{1}{2}}, b_2 = N, b_3 = N^{\frac{3}{2}}, b_4 = N^2, b_5 = N^{\frac{5}{2}}$. By definition, we have $\alpha = \max_j \frac{b_j}{t_{j-1}} = \Theta(N^{1/2})$. Choose the abstract active-set size as

$$n_1 = n_2 = n_3 = N, n_4 = N^{\frac{3}{4}}, n_5 = N^{\frac{3}{8}}, n_6 = 1.$$

For $j < 3$, it is easy to see that $\mathrm{LHS}_j = \mathrm{RHS}_j = 0$ as $n_j - n_{j+1} = 0$. Moreover, we have $S_2 = \frac{N^{1/2}}{N} + \frac{N}{N} = \Theta(N/N) = \Theta(1)$. Similarly, we have $S_3 = \Theta(N^{\frac{3}{2}}/N) = \Theta(N^{1/2})$, $S_4 = \Theta(N^{5/4})$, and $S_5 = \Theta(N^{17/8})$. Plugging the above into $\sum_j \mathrm{LHS}_j$ and $\sum_j \mathrm{RHS}_j$ gives

$$\sum_{j=1}^{B} \mathrm{LHS}_j = \Theta(1) \cdot \left(N^{3/2} + N^{7/4} + N^{15/8}\right) = \Theta\left(N^{15/8}\right)$$

and

$$\sum_{j=1}^{B} \mathrm{RHS}_j = \Theta(1) \cdot (N + N + N) = \Theta(N).$$

Therefore, we have $\sum_j \mathrm{LHS}_j / \sum_j \mathrm{RHS}_j = N^{15/8}/N = N^{7/8} = \alpha^{7/4}$. Thus, no uniform comparison of these two sums over such abstract sequences can have a multiplicative factor $o(\alpha^{7/4})$. □

D Online Convex Optimization

Online Gradient Descent (OGD) is a classical algorithm for online convex optimization. In the adversarially batched setting, the same algorithm applies, with the only change being that the update is performed once per batch using the aggregate subgradient. Specifically, starting from an arbitrary $x^{(1)} \in \mathcal{K}$, at the end of batch j the learner performs the update

$$x^{(j+1)} = \Pi_{\mathcal{K}}\left(x^{(j)} - \eta z_j\right),$$

where $\Pi_{\mathcal{K}}(y) = \arg\min_{x \in \mathcal{K}} \|x - y\|_2$ is Euclidean projection onto $\mathcal{K}$, and η is the learning rate.

D.1 Upper Bound.

Proof of Theorem 5 (Point 1). Fix $x^* \in \mathcal{K}$. For each batch j, we define the aggregate loss function $F_j(x) = \sum_{t=t_{j-1}+1}^{t_j} f_t(x)$. Recall that $z_j = \sum_{t=t_{j-1}+1}^{t_j} g_t$. As the sum of convex functions is convex, $z_j \in \partial F_j\left(x^{(j)}\right)$, which implies

$$F_j\left(x^{(j)}\right) - F_j\left(x^*\right) \le \left\langle z_j, x^{(j)} - x^* \right\rangle.$$

Therefore,

$$\sum_{j=1}^{B} \left(F_j\left(x^{(j)}\right) - F_j\left(x^*\right)\right) \le \sum_{j=1}^{B} \left\langle z_j, x^{(j)} - x^* \right\rangle.$$

It remains to bound the right-hand side using the standard projected OGD argument. By the non-expansiveness of Euclidean projection, and since $x^* \in \mathcal{K}$, $\Pi_{\mathcal{K}}(x^*) = x^*$, so

$$\left\|x^{(j+1)} - x^*\right\|_2^2 = \left\|\Pi_{\mathcal{K}}\left(x^{(j)} - \eta z_j\right) - \Pi_{\mathcal{K}}\left(x^*\right)\right\|_2^2.$$

Expanding the square gives

$$\left\|x^{(j+1)} - x^*\right\|_2^2 \le \left\|x^{(j)} - x^*\right\|_2^2 - 2\eta \left\langle z_j, x^{(j)} - x^* \right\rangle + \eta^2 \|z_j\|_2^2$$

$$\iff \left\langle z_j, x^{(j)} - x^* \right\rangle \le \frac{\left\|x^{(j)} - x^*\right\|_2^2 - \left\|x^{(j+1)} - x^*\right\|_2^2}{2\eta} + \frac{\eta}{2}\|z_j\|_2^2.$$

Summing over $j = 1, \ldots, B$ gives

$$\sum_{j=1}^{B} \left\langle z_j, x^{(j)} - x^* \right\rangle \le \frac{\left\|x^{(1)} - x^*\right\|_2^2 - \left\|x^{(B)} - x^*\right\|_2^2}{2\eta} + \frac{\eta}{2}\|z_j\|_2^2.$$

As $\|x^{(1)} - x^*\|_2 \le D$, we have

$$\sum_{j=1}^{B} \left\langle z_j, x^{(j)} - x^* \right\rangle \le \frac{D^2}{2\eta} + \frac{\eta}{2} \sum_{j=1}^{B} \|z_j\|_2^2 \le \frac{D^2}{2\eta} + \frac{\eta G^2}{2} \|b\|_2^2,$$

which completes the proof. □

D.2 Lower Bound

We further show that the dependence on D, G and $\|b\|_2$ is unavoidable. It suffices to consider linear losses, which form a subclass of convex losses.

Proof of Theorem 6 (Point 1). Choose $x^+, x^- \in \mathcal{K}$ such that $\|x^+ - x^-\|_2 = D$ and define $u = \frac{x^+ - x^-}{D}$. For every batch j, independently sample a Rademacher random variable $Z_j \in \{-1, +1\}$ and let the environment use the linear loss

$$f_t(x) = G Z_j \langle u, x \rangle$$

throughout batch j. Since the learner chooses $x^{(j)}$ before observing Z_j,

$$\mathbb{E}\left[G b_j Z_j \langle u, x^{(j)} \rangle | Z_1, \dots, Z_{j-1} \right] = 0.$$

Hence the learner's expected cumulative loss is 0.

For the best fixed decision term, since the width of $\mathcal{K}$ in direction u is D, $\max_{x \in \mathcal{K}} \langle u, x \rangle - \min_{x \in \mathcal{K}} \langle u, x \rangle = D$. By symmetry of S,

$$\mathbb{E}\left[\min_{x \in \mathcal{K}} G \langle u, x \rangle \sum_{j=1}^{B} b_j Z_j = -\frac{DG}{2} \mathbb{E}\left| \sum_{j=1}^{B} b_j Z_j \right| \right],$$

which indicates that

$$\mathbb{E}[\mathrm{Reg}] = 0 - \left(-\frac{DG}{2} \mathbb{E}\left| \sum_{j=1}^{B} b_j Z_j \right| \right) = \frac{DG}{2} \mathbb{E}\left| \sum_{j=1}^{B} b_j Z_j \right|.$$

Applying the Khintchine inequality (Lemma 7) with $p = 1$ gives a regret lower bound of $\Omega\left(DG \sqrt{\sum_{j=1}^{B} b_j^2} \right) = \Omega(DG\|b\|_2)$ and completes the proof. □

E Experts Problem

Hedge[5] (Freund & Schapire, 1997) is a classical algorithm for the experts problem. In the non-batched setting, the algorithm maintains a positive weight $w_i^{(t)}$ for each expert $i \in [N]$ at each round $t \in [T]$. The learner then samples an expert according to the probability distribution $p_i^{(t)} = \frac{w_i^{(t)}}{\sum_{j=1}^{N} w_j^{(t)}}$ induced by the weights. After observing the losses, the algorithm updates the weights by assigning smaller weights to experts that incur larger losses with a learning rate $\eta > 0$:

$$w_i^{(t+1)} = w_i^{(t)} \exp(-\eta \ell_i^{(t)}).$$

In the adversarially batched setting, the same idea applies, where updates are performed using the cumulative loss observed for each batch. Note that the weights must remain constant within each batch, i.e., $w_i^{(t_{j-1}+1)} = \cdots = w_i^{(t_j)}$. Thus, we re-define the notation $w_i^{(j)}$ as the weight for expert i at the beginning of batch j in the rest of the paper, and write $p_i^{(j)}$ for the associated probability accordingly. A pseudo-code is provided in Algorithm 3.

[5]The algorithm is also known as the Exponential Weighted Average (EWA).

Algorithm 3 Hedge in the Adversarially Batched Experts Setting

Initialize: $w_i^{(1)} \leftarrow 1 \; \forall i \in [N]$; $j \leftarrow 1$
for $t = 1, \ldots, T$ **do**
  $p_i^{(j)} \leftarrow \dfrac{w_i^{(j)}}{\sum_{k=1}^{N} w_k^{(j)}} \; \forall i \in [N]$
  Sample expert $i_t \sim p^{(j)}$
  if cumulative losses are revealed to the player **then**
    $t_j \leftarrow t$
    Observe cumulative loss $L_i^{(j)} \; \forall i \in [N]$
    Update weights: $w_i^{(j+1)} \leftarrow w_i^{(j)} \exp(-\eta L_i^{(j)}) \qquad \forall i \in [N]$
    $j \leftarrow j + 1$
  end if
end for

E.1 Upper Bound

We show that Hedge naturally adapts to the adversarially batched experts setting. The only change from the standard algorithm is that the multiplicative update is applied once per batch, using the cumulative loss $L_i^{(j)} := \sum_{t=t_{j-1}+1}^{t_j} \ell_i^{(t)} \in [0, b_j]$. Thus, within each batch, the distribution remains fixed, and after observing the cumulative batch losses, the weights are updated as $w_i^{(j+1)} = w_i^{(j)} \exp(-\eta L_i^{(j)})$. The second point of the proof of Theorem 5 then follows from a simple modification to the standard Hedge analysis.

Lemma 5 (Hoeffding's Lemma)**.** *Let X be a real-valued random variable supported on an interval $[a, b]$. Then for any $\eta \in \mathbb{R}$,*

$$\ln \mathbb{E}\left[e^{-\eta X}\right] \leq -\eta \mathbb{E}\left[X\right] + \frac{\eta^2 (b-a)^2}{8}.$$

Proof of Theorem 5 (Point 2). Let $Z_j := \sum_{i=1}^{N} w_i^{(j)}$ be the total weight of all N experts at block j, which makes $p_i^{(j)} = \frac{w_i^{(j)}}{Z_j}$ We conduct the proof by bounding $\ln Z_{B+1}$ and recover the inequality in the regret formula. Recall that in the Hedge algorithm, the weight of each expert i is updated according to $w_i^{(j+1)} = w_i^{(j)} \cdot e^{-\eta L_i^{(j)}}$. It follows that $w_i^{(B+1)} = \exp\left(-\eta \sum_{j=1}^{B} L_i^{(j)}\right)$, which can be used to lower bound $\ln Z_B$ as follows:

$$\ln Z_{B+1} = \ln\left(\sum_{i=1}^{N} \exp\left(-\eta \sum_{j=1}^{B} L_i^{(j)}\right)\right) \geq \max_{i \in [N]} -\eta \sum_{j=1}^{B} L_i^{(j)} = -\eta \min_{i \in [N]} \sum_{j=1}^{B} L_i^{(j)}.$$

For the lower bound, we first bound the ratio $\frac{Z_{j+1}}{Z_j}$ using Lemma 5. As $p_i^{(j)} = \frac{w_i^{(j)}}{Z_j}$, we have

$$\ln \frac{Z_{j+1}}{Z_j} = \ln \sum_{i=1}^{N} \left(\frac{w_i^{(j)}}{Z_j} \exp\left(-\eta L_i^{(j)}\right)\right) = \ln \mathbb{E}_i\left[\exp\left(-\eta L_i^{(j)}\right)\right].$$

Applying Lemma 5 to $L_i^{(j)} = \sum_{k=t_{j-1}+1}^{t_j} l_i^{(k)} \in [0, b_j]$ and summing over all batches gives

$$\sum_{j=1}^{B} \ln \frac{Z_{j+1}}{Z_j} \leq \sum_{j=1}^{B} \left(-\eta \cdot \mathbb{E}\left[L^{(j)}\right] + \frac{\eta^2 b_j^2}{8}\right) = -\eta \sum_{j=1}^{B} \mathbb{E}\left[L^{(j)}\right] + \frac{\eta^2}{8} \sum_{j=1}^{B} b_j^2.$$

Since the left-hand side above is equal to $\ln Z_{B+1} - \ln Z_1 = \ln Z_{B+1} - \ln N$ by telescoping,

$$\ln Z_{B+1} \leq \ln N - \eta \sum_{j=1}^{B} \mathbb{E}\left[L^{(j)}\right] + \frac{\eta^2}{8} \sum_{j=1}^{B} b_j^2.$$

Combining the above with the lower bound gives

$$-\eta \min_{i^* \in [N]} \sum_{j=1}^{B} L_i^{(j)} \leq \ln N - \eta \sum_{j=1}^{B} \mathbb{E}[L^{(j)}] + \frac{\eta^2}{8} \sum_{j=1}^{B} b_j^2,$$

which implies the regret bound

$$\sum_{j=1}^{B} \mathbb{E}\left[L^{(j)}\right] - \min_{i^* \in [N]} \sum_{j=1}^{B} L_i^{(j)} \leq \frac{\ln N}{\eta} + \frac{\eta}{8} \sum_{j=1}^{B} b_j^2.$$

Setting $\eta = \sqrt{\frac{8 \ln N}{\sum_{j=1}^{B} b_j^2}}$ gives a regret bound of $\mathcal{O}(\|b\|_2 \sqrt{\log N})$. This completes the proof, since the regret is always trivially bounded by T. □

E.2 Lower Bound

Towards proving the second point of Theorem 6, we first prove a lower bound for any arbitrary batch structure, and then construct a broad family of hard instances that gives a matching worst-case lower bound. Specifically, we first prove Lemma 6, which gives an $\Omega(\|b\|_2)$ regret lower bound for the $N = 2$ case. Then, we prove the second point of Theorem 6 by bootstrapping from Lemma 6 via dividing the N-expert instance into $\approx \log_2 N$ independent two-expert instances.

Lemma 6. *For the adversarially batched expert problem with $N = 2$ experts and batch structure $\mathcal{B} = (b_1, b_2, \ldots, b_B)$, the minimax regret is at least $\Omega(\|b\|_2)$.*

We prove Lemma 6 using a standard construction, in which the losses are from independent coin flips, and the Khintchine inequality (Lemma 7).

Lemma 7 (Khintchine inequality)**.** *Let $\{Z_i\}_{i=1}^N$ be i.i.d. random variables with $\Pr[Z_i = +1] = \Pr[Z_i = -1] = 1/2$ for each $i \in [N]$. Let $0 < p < \infty$ and let $x_1, \ldots, x_i \in \mathbb{C}$. Then*

$$A_p \left(\sum_{i=1}^{N} |x_i|^2 \right)^{\frac{1}{2}} \leq \left(\mathbb{E}\left[\left| \sum_{i=1}^{N} x_i Z_i \right|^p \right] \right)^{\frac{1}{p}} \leq B_p \left(\sum_{i=1}^{N} |x_i|^2 \right)^{\frac{1}{2}}$$

for some constants $A_p, B_p > 0$.

Proof of Lemma 6. Consider two experts who play opposite actions at random within each batch. Let $X_1, X_2, \ldots, X_B$ be independent random variables that follow $\mathsf{Bern}(1/2)$. The losses are specified by $L_1^{(j)} = b_j \cdot X_j$ and $L_2^{(j)} = b_j \cdot (1 - X_j)$ for each $j \in [B]$, and we write $L^{(j)} = \left(L_1^{(j)}, L_2^{(j)}\right)$ as the (two-dimensional) loss vector. Over the randomness in the losses (i.e., in X), the expectation of the regret is given by

$$\mathop{\mathbb{E}}_{X}\left[\sum_{j=1}^{B} \left\langle p^{(j)}, L^{(j)} \right\rangle\right] - \mathop{\mathbb{E}}_{X}\left[\min_{i^* \in [N]} \sum_{j=1}^{B} L_{i^*}^{(j)}\right].$$

For the first term, note that $p^{(j)}$ only depends on $X_1, X_2, \ldots, X_{j-1}$. Thus, X_j still follows $\mathsf{Bern}(1/2)$ conditioned on $p^{(j)}$. It follows that $\mathbb{E}_X\left[\left\langle p^{(j)}, L^{(j)} \right\rangle\right] = b_j/2$. For the second term, we have

$$\mathop{\mathbb{E}}_{X}\left[\min_{i^* \in [N]} \sum_{j=1}^{B} L_{i^*}^{(j)}\right] = \mathop{\mathbb{E}}_{X}\left[\min\left\{\sum_{j=1}^{B} b_j \cdot X_j, \sum_{j=1}^{B} b_j \cdot (1 - X_j)\right\}\right].$$

Therefore, we can re-write the expected regret as

$$\sum_{j=1}^{B} \frac{b_j}{2} - \mathop{\mathbb{E}}_{X}\left[\min\left\{\sum_{j=1}^{B} b_j \cdot X_j, \sum_{j=1}^{B} b_j \cdot (1-X_j)\right\}\right]$$
$$= \mathop{\mathbb{E}}_{X}\left[\max\left\{\sum_{j=1}^{B} b_j \cdot (1/2 - X_j), \sum_{j=1}^{B} b_j \cdot (X_j - 1/2)\right\}\right]$$
$$= \mathop{\mathbb{E}}_{X}\left[\left|\sum_{j=1}^{B} b_j \cdot (X_j - 1/2)\right|\right] = \frac{1}{2}\mathop{\mathbb{E}}_{Z}\left[\left|\sum_{j=1}^{B} b_j Z_j\right|\right],$$

where $Z_1, Z_2, \ldots, Z_B$ denote independent Rademacher random variables (i.e., $\Pr[Z_j = +1] = \Pr[Z_j = -1] = 1/2$) in the last step. Applying Lemma 7 with $p = 1$ gives a regret lower bound of $\frac{A_1}{2} \cdot \sqrt{\sum_{j=1}^{B} b_j^2} = \Omega(\|b\|_2)$ and completes the proof. □

Lemma 6 gives the lower bound of the adversarially batched expert problem under the $N = 2$ case. To handle the general-N case, the intuition is to partition the B blocks into $\approx \log_2 N$ independent *stages*, such that the block sequence of each stage has a 2-norm of $\approx \|b\|_2/\sqrt{\log N}$. For each stage, we design only two different losses, so each stage can be viewed as an independent 2-expert problem. We then apply Lemma 6 to lower bound the regret within each stage, and argue that these regrets are additive. We start by proving the lemma below, which states that under the worst-case construction that satisfies $\max_{j\in[B]} b_j \le \mathcal{O}(\|b\|_2/\sqrt{\log N})$, we can partition the game into $\Omega(\log N)$ stages. This, in turn, gives an $\Omega(\|b\|_2\sqrt{\log N})$ worst-case lower bound.

Lemma 8. *Suppose that $b_1, b_2, \ldots, b_B > 0$ and $\max_{j\in[B]} b_j \le c \cdot \frac{\|b\|_2}{\sqrt{\log_2 N}}$ for some $c > 0$. Then, there exists $S \ge \left\lfloor \frac{\log_2 N}{1+c^2} \right\rfloor$ and a partition $\{G_1, G_2, \ldots, G_S\}$ of $[B]$ such that, for each $s \in [S]$,*

$$\sqrt{\sum_{j\in G_s} b_j^2} \ge \frac{\|b\|_2}{\sqrt{\log_2 N}}.$$

Moreover, each partition G_s consists of contiguous integers, i.e., $G_s = \{j_0, j_0+1, \ldots, j_1\}$, for some integers $j_0 \le j_1$.

Proof. We construct the partition by greedily forming minimal segments, so that the sum of squared entries within each segment is $\ge M := \|b\|_2^2/(\log_2 N)$. Formally, we start with $t_0 = 0$. For each $s = 1, 2, \ldots$, we define t_s to be the smallest index that satisfies

$$\sum_{j=t_{s-1}+1}^{t_s} b_j^2 \ge M.$$

If such an index exists, we must have $\sum_{j=t_{s-1}+1}^{t_s-1} b_j^2 < M$. If no valid t_s exists, we set $S = s-1$ and reset t_S to B. The resulting partition is then given by $G_s = \{t_{s-1}+1, t_{s-1}+2, \ldots, t_s\}$ for each $s \in [S]$. By construction, each $s \in [S]$ satisfies the desired inequality

$$\sqrt{\sum_{j\in G_s} b_j^2} \ge \sqrt{M} = \frac{\|b\|_2}{\sqrt{\log_2 N}}.$$

It remains to lower bound S, the number of segments. Note that for each $s \in [S-1]$, we have

$$\sum_{j\in G_s} b_j^2 = \sum_{j=t_{s-1}+1}^{t_s-1} b_j^2 + b_{t_s}^2 \le M + \left(c \cdot \frac{\|b\|_2}{\sqrt{\log_2 N}}\right)^2 = (1+c^2) \cdot \frac{\|b\|_2^2}{\log_2 N}.$$

Furthermore, we must have $\sum_{j \in G_S} b_j^2 < 2 \cdot (1+c^2) \cdot \frac{\|b\|_2^2}{\log_2 N}$; otherwise, we should have formed at least two segments from blocks $t_{S-1}+1$ through B. Therefore, we have

$$\|b\|_2^2 = \sum_{s=1}^{S} \sum_{j \in G_s} b_j^2 < (S+1) \cdot (1+c^2) \cdot \frac{\|b\|_2^2}{\log_2 N}.$$

This gives $S > \frac{\log_2 N}{1+c^2} - 1$ and thus $S \geq \left\lfloor \frac{\log_2 N}{1+c^2} \right\rfloor$. □

Now we are ready to prove the second point of Theorem 6.

Proof of Theorem 6 (Point 2). We apply Lemma 8 to partition the block representation $\mathcal{B} = (b_1, \ldots, b_B)$ of the family of hard instances that satisfies $\max_{j \in [B]} b_j \leq \mathcal{O}(\|b\|_2 / \sqrt{\log N})$ into $S \geq \Omega(\log N)$ contiguous stages $G_1, G_2, \ldots, G_S$. Each $G_s = \{t_{s-1}+1, t_{s-1}+2, \ldots, t_s\}$ denotes the indices of the blocks within stage s, and the number of time steps in stage s is $T_s = \sum_{j \in G_s} b_j$.

For each stage $s \in [S]$, we independently apply Lemma 6 to construct two random loss sequences

$$\left(\ell_0^{(t_{s-1}+1)}, \ell_0^{(t_{s-1}+2)}, \ldots, \ell_0^{(t_s)}\right) \quad \text{and} \quad \left(\ell_1^{(t_{s-1}+1)}, \ell_1^{(t_{s-1}+2)}, \ldots, \ell_1^{(t_s)}\right),$$

where $\ell_i^{(j)}$ denotes the total loss of expert $i \in \{0,1\}$ within batch $j \in G_s$.

Without loss of generality, we assume that there are exactly $N = 2^S$ experts. For clarity, we label the experts using length-S binary strings in $\{0,1\}^S$, i.e., each expert is labeled with S bits $i_1, i_2, \ldots, i_S$. Then, the losses in the original instance are defined as

$$L_{i_1, i_2, \ldots, i_S}^{(j)} \coloneqq \ell_{i_s}^{(j)},$$

where $s \in [S]$ is the unique index such that $j \in G_s$. In other words, the expert labeled $i = (i_1, i_2, \ldots, i_S)$ incurs the same loss as "expert i_s" does within stage s.

Now, suppose that the learner incurs a loss of $\widehat{L}^{(j)}$ within each block $j \in [B]$. Then, the regret of the learner is given by

$$\begin{aligned}\sum_{j=1}^{B} \widehat{L}^{(j)} - \min_{i \in \{0,1\}^S} \sum_{j=1}^{B} L_i^{(j)} &= \sum_{s=1}^{S} \sum_{j \in G_s} \widehat{L}^{(j)} - \sum_{s=1}^{S} \min_{i_s \in \{0,1\}} \sum_{j \in G_s} \ell_{i_s}^{(j)} \\ &= \sum_{s=1}^{S} \left[\sum_{j \in G_s} \widehat{L}^{(j)} - \min_{i_s \in \{0,1\}} \sum_{j \in G_s} \ell_{i_s}^{(j)} \right].\end{aligned}$$

Note that $\sum_{j \in G_s} \widehat{L}^{(j)} - \min_{i_s \in \{0,1\}} \sum_{j \in G_s} \ell_{i_s}^{(j)}$ is exactly the learner's regret within stage s. Since we constructed the losses using Lemma 6, the regret within each stage s is at least $\Omega\left(\sqrt{\sum_{j \in G_s} b_j^2}\right)$, which is in turn lower bounded by $\Omega\left(\frac{\|b\|_2}{\sqrt{\log N}}\right)$ by our construction of the partition from Lemma 8. Finally, since Lemma 8 guarantees $S \geq \Omega(\log N)$, the regret is lower bounded by $\Omega(\|b\|_2 \sqrt{\log N})$, which marks the end of the proof. □

F Adversarial Bandits

Recall that in the adversarial bandits setting, the player only observes the loss of the chosen arm $l_{i_t}^{(t)}$ at each round $t \in [T]$. The EXP3 algorithm of Auer, Cesa-Bianchi, Freund, and Schapire (Auer et al., 2002b) uses $\tilde{l}^{(t)} = \left(0, \ldots, l_{i_t}^{(t)} / p_{i_t}^{(t)}, \ldots, 0\right)$ as an estimator for the actual loss vector $l^{(t)}$. This estimator is unbiased since, for each coordinate $i \in [N]$,

$$\mathop{\mathbb{E}}_{i_t \sim p^{(t)}} \left[\tilde{l}_i^{(t)}\right] = \Pr_{i_t \sim p^{(t)}} [i_t = i] \cdot \frac{l_{i_t}^{(t)}}{p_{i_t}^{(t)}} + \Pr_{i_t \sim p^{(t)}} [i_t \neq i] \cdot 0 = l_i^{(t)}.$$

In the adversarially batched setting, we update the weights according to

$$w_i^{(j+1)} = w_i^{(j)} \exp(-\eta \tilde{L}_i^{(j)}),$$

where $\tilde{L}_i^{(j)} = \sum_{t=t_{j-1}+1}^{t_j} \tilde{l}_i^{(t)}$. The weights then determine the probability of pulling each arm via $p_i^{(j+1)} = \frac{w_i^{(j+1)}}{\sum_{n\in[N]} w_n^{(j+1)}}$. A detailed pseudo-code is provided in Algorithm 4.

Algorithm 4 EXP3 in the Adversarially Batched Bandits Setting

Initialize: $w_i^{(1)} \leftarrow 1\ \forall i \in [N]$, $p^{(1)} \leftarrow \frac{w_i^{(1)}}{\sum_{k=1}^N w_k^{(1)}} \forall i \in [N]$, and $j \leftarrow 1$
Sample arm $i_j \sim p^{(j)}$
for $t = 1, \ldots, T$ **do**
  $p_i^{(j)} \leftarrow \dfrac{w_i^{(j)}}{\sum_{k=1}^N w_k^{(j)}}\ \forall i \in [N]$
  if bandit feedback for the current block is revealed to the player **then**
    $t_j \leftarrow t$
    Observe block loss of the played arm: $L_{i_j}^{(j)}$
    Form importance-weighted loss estimates: $\widehat{L}_i^{(j)} \leftarrow \frac{L_i^{(j)}}{p_i^{(j)}} \mathbf{1}\{i = i_j\}$
    Update weights: $w_i^{(j+1)} \leftarrow w_i^{(j)} \exp\left(-\eta \widehat{L}_i^{(j)}\right) \qquad \forall i \in [N]$
    $j \leftarrow j + 1$
    Sample arm $i_j \sim p^{(j)}$
  end if
end for

F.1 Upper Bound

Proof of Theorem 5 (Point 3). Let $Z_j := \sum_{i=1}^N w_i^{(j)}$ to be the total weight of all N experts at block j, and then bound $\ln Z_{B+1}$ on both directions.

Fix an arm $i^* \in [N]$. To lower bound $\ln Z_{B+1}$, we note

$$\ln Z_{B+1} = \ln\left(\sum_{i=1}^N \exp\left(-\eta \sum_{j=1}^B \tilde{L}_i^{(j)}\right)\right) \geq -\eta \cdot \sum_{j=1}^B \tilde{L}_{i^*}^{(j)}.$$

For the upper bound, recall that $p_i^{(j)} = \frac{w_i^{(j)}}{\sum_{n=1}^N w_n^{(j)}}$. Then,

$$\frac{Z_{j+1}}{Z_j} = \sum_{i=1}^N \frac{w_i^{(j)}}{Z_j} \exp\left(-\eta \tilde{L}_i^{(j)}\right) = \sum_{i=1}^N p_i^{(j)} \exp\left(-\eta \tilde{L}_i^{(j)}\right).$$

Applying the inequality $e^{-x} \leq 1 - x + \frac{x^2}{2}$ for $x \geq 0$, we obtain

$$\begin{aligned}\frac{Z_{j+1}}{Z_j} &\leq \sum_{i=1}^N p_i^{(j)} \left[1 - \eta \tilde{L}_i^{(j)} + \frac{\eta^2}{2}\left(\tilde{L}_i^{(j)}\right)^2\right] \\ &= 1 - \eta \sum_{i=1}^N p_i^{(j)} \tilde{L}_i^{(j)} + \frac{\eta^2}{2} \sum_{i=1}^N p_i^{(j)} \left(\tilde{L}_i^{(j)}\right)^2.\end{aligned}$$

Taking the logarithm and summing over all blocks leads to

$$\ln Z_{B+1} - \ln Z_1 \leq \sum_{j=1}^B \ln\left(1 - \eta \sum_{i=1}^N p_i^{(j)} \tilde{L}_i^{(j)} + \frac{\eta^2}{2} \sum_{i=1}^N p_i^{(j)} \left(\tilde{L}_i^{(j)}\right)^2\right).$$

Applying $\ln(1+x) \le x$ and noting $Z_1 = N$ gives

$$\ln Z_{B+1} \le \ln N - \eta \sum_{j=1}^{B} \sum_{i=1}^{N} p_i^{(j)} \tilde{L}_i^{(j)} + \frac{\eta^2}{2} \sum_{j=1}^{B} \sum_{i=1}^{N} p_i^{(j)} \left(\tilde{L}_i^{(j)}\right)^2.$$

Combining the above with the lower bound $\ln Z_{B+1} \ge -\eta \cdot \sum_{j=1}^{B} \tilde{L}_{i^*}^{(j)}$ gives

$$\eta \left(\sum_{j=1}^{B} \sum_{i=1}^{N} p_i^{(j)} \tilde{L}_i^{(j)} - \sum_{j=1}^{B} \tilde{L}_{i^*}^{(j)} \right) \le \ln N + \frac{\eta^2}{2} \sum_{j=1}^{B} \sum_{i=1}^{N} p_i^{(j)} \left(\tilde{L}_i^{(j)}\right)^2.$$

Equivalently,

$$\sum_{j=1}^{B} \sum_{i=1}^{N} p_i^{(j)} \tilde{L}_i^{(j)} - \sum_{j=1}^{B} \tilde{L}_{i^*}^{(j)} \le \frac{\ln N}{\eta} + \frac{\eta}{2} \sum_{j=1}^{B} \sum_{i=1}^{N} p_i^{(j)} \left(\tilde{L}_i^{(j)}\right)^2.$$

Recall that $p^{(j)}$ is itself a random variable and directly affects $\tilde{L}^{(j)}$. Given the definition of pseudo-regret, we check the expectation of $\sum_{j=1}^{B} \sum_{i=1}^{N} p_i^{(j)} \tilde{L}_i^{(j)} - \sum_{j=1}^{B} \tilde{L}_{i^*}^{(j)}$. Specifically,

$$\begin{aligned}
\mathop{\mathbb{E}}_{p^{(j)}} \left[\sum_{j=1}^{B} \sum_{i=1}^{N} p_i^{(j)} \tilde{L}_i^{(j)} - \sum_{j=1}^{B} \tilde{L}_{i^*}^{(j)} \right] &= \sum_{j=1}^{B} \sum_{i=1}^{N} p_i^{(j)} \mathop{\mathbb{E}}_{p^{(j)}} \left[\tilde{L}_i^{(j)} \right] - \sum_{j=1}^{B} \mathop{\mathbb{E}}_{p^{(j)}} \left[\tilde{L}_{i^*}^{(j)} \right] \\
&= \sum_{j=1}^{B} \sum_{i=1}^{N} p_i^{(j)} L_i^{(j)} - \sum_{j=1}^{B} L_{i^*}^{(j)}.
\end{aligned}$$

Taking the result back to the LHS of the inequality leads to

$$\begin{aligned}
\sum_{j=1}^{B} \sum_{i=1}^{N} p_i^{(j)} L_i^{(j)} - \sum_{j=1}^{B} L_{i^*}^{(j)} &\le \mathop{\mathbb{E}}_{p^{(j)}} \left[\frac{\ln N}{\eta} + \frac{\eta}{2} \sum_{j=1}^{B} \sum_{i=1}^{N} p_i^{(j)} \left(\tilde{L}_i^{(j)}\right)^2 \right] \\
&= \frac{\ln N}{\eta} + \frac{\eta}{2} \sum_{j=1}^{B} \sum_{i=1}^{N} p_i^{(j)} \cdot \mathop{\mathbb{E}}_{p^{(j)}} \left[\left(\tilde{L}_i^{(j)}\right)^2 \right] \\
&= \frac{\ln N}{\eta} + \frac{\eta}{2} \sum_{j=1}^{B} \sum_{i=1}^{N} p_i^{(j)} \cdot p_i^{(j)} \cdot \left(\frac{L_i^{(j)}}{p_i^{(j)}} \right)^2 \\
&= \frac{\ln N}{\eta} + \frac{\eta}{2} \sum_{j=1}^{B} \sum_{i=1}^{N} \left(L_i^{(j)}\right)^2 \\
&\le \frac{\ln N}{\eta} + \frac{\eta N}{2} \sum_{j=1}^{B} b_j^2. \qquad (L_i^{(j)} \in [0, b_i])
\end{aligned}$$

Setting $\eta = \frac{\sqrt{2 \ln N}}{\|b\|_2 \sqrt{N}}$ bounds the pseudo-regret by $\|b\|_2 \cdot \sqrt{2N \ln N} = \mathcal{O}(\|b\|_2 \cdot \sqrt{N \log N})$. □

F.2 Lower Bound

Similar to the second point of the proof of Theorem 6, we first prove a lower bound for any arbitrary batch structure $\mathcal{B}$, then construct a family of hard instances that gives a tight worst-case lower bound.

Proof of Theorem 6 (Point 3). Recall the notation that $b_{\max} = \max_{j \in [B]} b_j$. Let $K = \min\left\{N, \max\left\{2, \lfloor \|b\|_2^2 / b_{\max}^2 \rfloor\right\}\right\}$. As $\|b\|_2^2 \ge b_{\max}^2$ and $N \ge 2$, we have

$$K \ge 2; \; K \le 2\|b\|_2^2 / b_{\max}^2.$$

We construct K hard environments, indexed by $\theta \in [K]$. Fixing a sufficiently small absolute constant $\gamma = 1/16$, we define $\delta_j = \gamma \frac{\sqrt{K} b_j}{\|b\|_2}$ for $j \in [B]$. It is easy to see that

$$\delta_j \le \gamma \frac{\sqrt{K} b_{\max}}{\|b\|_2} \le \gamma\sqrt{2} < \frac{1}{4}.$$

For every $\theta \in [K]$, we define an oblivious randomized environment P_θ as follows. For every batch $j \in [B]$ and arm $i \in [N]$, independently sample

$$X_{j,i} \sim \begin{cases} \text{Bernoulli}\left(\frac{1}{2} - \delta_j\right), & i = \theta, \\ \text{Bernoulli}\left(\frac{1}{2}\right), & i \neq \theta. \end{cases}$$

Within batch j, set the loss of arm i to be constant across all b_j rounds, i.e., $l_i^{(t)} = X_{j,i};\ t = t_{j-1}+1, \ldots, t_j$. Thus the cumulative loss of arm i in batch j is $L_i^{(j)} = b_j X_{j,i}$. Since the environment is oblivious and the random variables associated with batch j are independent of the history before batch j, under P_θ we have

$$\mathbb{E}\left[L_{i_j}^{(j)}\right] = \Pr_\theta[i_j = \theta] b_j (\frac{1}{2} - \delta_j) + \Pr_\theta[i_j \neq \theta] \frac{b_j}{2} = \frac{b_j}{2} - b_j \delta_j \Pr_\theta[i_j = \theta],$$

where, recall that i_j is the arm selected by the learner in batch j. On the other hand,

$$\mathbb{E}_\theta\left[L_\theta^{(j)}\right] = b_j(\frac{1}{2} - \delta_j),$$

while every arm $i \neq \theta$ has expected loss $b_j/2$. Therefore, the pseudo-regret under P_θ is

$$\begin{aligned} \overline{\text{Reg}}_\theta &= \sum_{j=1}^{B} \left(\frac{b_j}{2} - b_j \delta_j \Pr[i_j = \theta] \right) - \sum_{j=1}^{B} b_j \left(\frac{1}{2} - \delta_j \right) \\ &= \sum_{j=1}^{B} b_j \delta_j \left(1 - \Pr_\theta[i_j = \theta] \right) \\ &= \sum_{j=1}^{B} b_j \delta_j - \sum_{j=1}^{B} b_j \delta_j \Pr_\theta[i_j = \theta] \end{aligned}$$

For the first term, we have

$$\sum_{j=1}^{B} b_j \delta_j = \gamma \frac{\sqrt{K}}{\|b\|_2} \sum_{j=1}^{B} b_j^2 = \gamma \sqrt{K} \|b\|_2.$$

Now we show that, for at least one $\theta \in [K]$, the learner selects arm θ incorrectly on a constant fraction of this weighted mass. Define the null environment P_0, under which $X_{j,i} \sim \text{Bern}(1/2)$ independently for every $j \in [B]$ and $i \in [N]$. We compare P_0 and P_θ, Since the learner uses the same decision rule under P_0 and P_θ, the only difference between the two transcript distributions occurs in batches in which arm θ is selected. By the chain rule for KL divergence,

$$D_{\text{KL}}(P_0 \| P_\theta) = \sum_{j=1}^{B} \Pr_0[i_j = \theta] \, \text{kl}(1/2 \| 1/2 - \delta_j),$$

where $\text{kl}(p\|q)$ denotes the binary relative entropy. For $0 \le \delta_j \le 1/4$,

$$\text{kl}(1/2 \| 1/2 - \delta) = -\frac{1}{2} \log(1 - 4\delta^2) \le \frac{8}{3} \delta_j^2.$$

Hence, $D_{\text{KL}}(P_0 \| P_\theta) \le \frac{8}{3} \sum_{j=1}^{B} \Pr_0[i_j = \theta] \, \delta_j^2$. By Pinsker's inequality, for every $j \in [B]$,

$$\Pr_\theta[i_j = \theta] \le \Pr_0[i_j = \theta] + \sqrt{\frac{1}{2} D_{\text{KL}}(P_0 \| P_\theta)}.$$

Taking this back, we have

$$\overline{\mathrm{Reg}}_\theta \geq \gamma\sqrt{K}\|b\|_2 - \sum_{j=1}^{B} b_j\delta_j \Pr_0[i_j = \theta] - \gamma\sqrt{K}\|b\|_2\sqrt{\frac{1}{2}D_{\mathrm{KL}}(P_0\|P_\theta)}.$$

Now we average the above inequality over $\theta \in [K]$. As the learner selects at most one arm in each batch, $\sum_{\theta=1}^{K}\Pr[i_j = \theta] \leq 1$, thus

$$\frac{1}{K}\sum_{\theta=1}^{K}\sum_{j=1}^{B} b_j\delta_j \Pr[i_j = \theta] \leq \frac{\sum_{j=1}^{B} b_j\delta_j}{K} = \frac{\gamma\sqrt{K}\|b\|_2}{K}.$$

For the last term, the Cauchy-Schwarz inequality gives

$$\begin{aligned}
\frac{1}{K}\sum_{\theta=1}^{K}\sqrt{\frac{1}{2}D_{\mathrm{KL}}(P_0\|P_\theta)} &\leq \sqrt{\frac{1}{K}\sum_{\theta=1}^{K}\frac{1}{2}D_{\mathrm{KL}}(P_0\|P_\theta)} \\
&\leq \sqrt{\frac{1}{K}\sum_{\theta=1}^{K}\frac{1}{2}\sum_{j=1}^{B}\Pr_0[i_j = \theta]\frac{8}{3}\delta_j^2} \\
&= \sqrt{\frac{4}{3K}\sum_{j=1}^{B}\delta_j^2\sum_{\theta=1}^{K}\Pr_0[i_j = \theta]} \\
&\leq \sqrt{\frac{4}{3K}\sum_{j=1}^{B}\delta_j^2}.
\end{aligned}$$

Substituting δ_j, we have

$$\sqrt{\frac{4}{3K}\sum_{j=1}^{B}\delta_j^2} = \sqrt{\frac{4}{3K}\gamma^2 K\frac{\sum_{j=1}^{B} b_j^2}{\|b\|_2^2}} = \frac{2\gamma}{\sqrt{3}}.$$

Combining together,

$$\begin{aligned}
\frac{1}{K}\sum_{\theta=1}^{K}\overline{\mathrm{Reg}}_\theta &\geq \frac{1}{K}\sum_{\theta=1}^{K}\gamma\sqrt{K}\|b\|_2 - \frac{\gamma\sqrt{K\|b\|_2}}{K} - \gamma\sqrt{K}\|b\|_2\frac{2\gamma}{3} \\
&= \gamma\sqrt{K}\|b\|_2\left(1 - \frac{1}{K} - \frac{2\gamma}{\sqrt{3}}\right).
\end{aligned}$$

As $K \geq 2$ and $\gamma = 1/16$, it is easy to see that $\frac{1}{K}\sum_{\theta=1}^{K}\overline{\mathrm{Reg}}_\theta = \Omega(\sqrt{K\|b\|_2^2})$. From the definition of K,

$$K \geq \frac{1}{2}\min\left\{N, \frac{\|b\|_2^2}{b_{\max}^2}\right\}.$$

Therefore,

$$\sqrt{K\|b\|_2^2} = \Omega\left(\sqrt{\|b\|_2^2\min\{N, \|b\|_2^2/b_{\max}^2\}}\right) = \Omega\left(\min\left\{\|b\|_2\sqrt{N}, \frac{\|b\|_2^2}{b_{\max}}\right\}\right).$$

Finally, we consider a hard instance where $b_{\max} \leq \mathcal{O}(\|b\|_2/\sqrt{N})$. Under the $\min$ operation above, this hard instance yields a worst-case lower bound of $\Omega\left(\|b\|_2\sqrt{N}\right)$, which ends the proof. □

G Reduction from Scale-Free Online Learning

We perform the reduction by treating each batch as a single meta-round. In batch j, the learner uses the decision produced by the online algorithm throughout $I_j = \{t_{j-1} + 1, \ldots, t_j\}$, and feeds the aggregate loss over the batch back to the algorithm at time t_j.

G.1 OCO AND EXPERTS

We use the algorithm, namely, SOLO FTRL, and theorems proposed in Orabona & Pál (2018).

Lemma 9 (Corollary 1 of Orabona & Pál (2018)). *Let $\mathcal{K}$ be a nonempty bounded closed convex set equipped with a norm $\|\cdot\|$, and let $\psi : \mathcal{K} \to \mathbb{R}_+$ be lower semicontinuous and 1-strongly convex with respect to $\|\cdot\|$. Define $S = \sup_{x\in\mathcal{K}} \psi(x)$. Then SOLO FTRL with regularizer*

$$R(w) = \frac{\sqrt{2.75}}{\sqrt{\sup_{v\in\mathcal{K}} f(v)}} f(w)$$

satisfies

$$\mathrm{Reg} \le 13.3 \sqrt{\sup_{v\in\mathcal{K}} f(v) \sum_{j=1}^{B} \|z_j\|_*^2},$$

for any sequence of loss vector $z_1, \dots, z_B$, where $\|\cdot\|_*$ is the dual norm.

OCO. Let $x^{(j)} \in \mathcal{K}$ denote the action selected at meta-round j and played throughout batch j, i.e., $x^{(t)} = x^{(j)}$ for every $t \in I_j$. For each $t \in I_j$, recall $g_t \in \partial f_t\left(x^{(j)}\right)$ is a subgradient at the decision used in batch j, and $z_j = \sum_{t\in I_j} g_t$. For every $u \in \mathcal{K}$, convexity gives

$$\sum_{t=1}^{T} \left(f_t\left(x^{(t)}\right) - f_t(u)\right) \le \sum_{j=1}^{B} \langle z_j, x^{(j)} - u\rangle.$$

Thus, the batched OCO problem is reduced to OLO on the meta-losses z_j. Applying Lemma 9 gives

$$\mathrm{Reg} \le C \sqrt{\frac{D^2}{2} \sum_{j=1}^{B} \|z_j\|_2^2} = \mathcal{O}\left(DG\sqrt{\sum_{j=1}^{B} b_j^2}\right).$$

Experts Problem. Let $l_t \in [0,1]^N$ be the expert-loss vector at round t, and let $z_j = \sum_{t\in I_j} l_t$. Then $\|z_j\|_\infty \le b_j$, and the regret in the batched experts problem is the OLO regret

$$\mathrm{Reg} = \sum_{j=1}^{B} \langle p_j, z_j\rangle - \min_{p\in\Delta_N} \sum_{j=1}^{B} \langle p, z_j\rangle.$$

Equip Δ_N with the l_1-norm and take

$$f(p) = \sum_{i=1}^{N} p_i \log p_i + \log N.$$

By Pinsker's inequality, f is 1-strongly convex with respect to $\|\cdot\|_1$, and $\sup_{p\in\Delta_N} f(p) = \log N$. Since the dual norm of l_1 is l_∞, Lemma 9 gives

$$\mathrm{Reg} \le C \sqrt{\log N \sum_{j=1}^{B} \|z_j\|_\infty^2} \le \mathcal{O}\left(\|b\|_2 \sqrt{\log N}\right).$$

G.2 ADVERSARIAL BANDITS

We use UMAB-NN proposed in Chen & Zhang (2023), as well as its theorem guarantees.

Lemma 10 (Theorem 1 of Chen & Zhang (2023)). *Consider the oblivious adversarial bandit problem with N arms and nonnegative loss vectors $z_1, \dots, z_B \in \mathbb{R}_+^N$. The UMAB-NN algorithm satisfies*

$$\mathbb{E}[\mathrm{Reg}] = \tilde{O}\left(\sqrt{N \sum_{j=1}^{B} \|z_j\|_\infty^2}\right).$$

At the beginning of batch j, the bandit algorithm samples an arm i_j, which is played throughout I_j. At the end of the batch, the learner observes z_j, i_j, the aggregate loss of the selected arm. Since $l_t \in [0,1]^N$, it is easy to see that $z_j \in \mathbb{R}_+^N$ and $\|z_j\|_\infty \le b_j$. Applying Lemma 10 gives

$$\mathbb{E}[\mathrm{Reg}] = \tilde{O}\left(\sqrt{N}\|b\|_2\right).$$

H Reduction from Delayed-Feedback Models

Similar to Appendix G, we show that existing delayed-feedback results also recover the same order of upper bounds for the OCO and the experts problem. The reduction is conducted by viewing the feedback generated at each round of a batch as being delayed until the end of that batch. Specifically, for every $t \in I_j$, we view the feedback generated at round t as being delayed until the end of batch j, and define $\tau_t = t_j - t$. We use the delayed FTRL-PROX analysis of Joulani et al. (2016). The following lemma is a specialization of their Theorem 4 to linear losses.

Lemma 11 (Specialization of Theorem 4 of Joulani et al. (2016))**.** *Let $\mathcal{K}$ be a nonempty closed convex set equipped with a norm $\|\cdot\|$, and let $\psi : \mathcal{K} \to \mathbb{R}_+$ be 1-strongly convex with respect to $\|\cdot\|$. Define $S = \sup_{x\in\mathcal{K}} \psi(x)$. Consider delayed online linear optimization with loss vectors $g_1, \dots, g_T$ satisfying $\|g_t\|_* \le G$. Let $\widetilde{\tau}_s$ denote the update delay of the s-th feedback processed by the SOLID reduction of Joulani et al. (2016). For any non-increasing sequence $\eta_1, \dots, \eta_T > 0$, FTRL-PROX with cumulative regularizer $r_{0:s}(x) = \frac{\psi(x)}{\eta_s}$ satisfies*

$$\mathrm{Reg} \le \frac{S}{\eta_T} + \frac{G^2}{2} \sum_{s=1}^{T} \eta_s \left(1 + 2\widetilde{\tau}_s\right).$$

Moreover, $\sum_{s=1}^{T} \widetilde{\tau}_s = \sum_{t=1}^{T} \tau_t$.

Let $a_s = 1 + 2\widetilde{\tau}_s$, $A_s = \sum_{r=1}^{s} a_r$. Setting $\eta_s = \frac{\sqrt{S}}{G\sqrt{A_s}}$ is batch-oblivious, since A_s depends only on feedback already received. Using $\sum_{s=1}^{T} a_s/\sqrt{A_s} \le 2\sqrt{A_T}$, Lemma 11 gives $\mathrm{Reg} \le 2G\sqrt{SA_T}$. For our batched feedback pattern,

$$\sum_{t=1}^{T} \tau_t = \sum_{j=1}^{B} \sum_{t=t_{j-1}+1}^{t_j} (t_j - t) = \frac{1}{2} \sum_{j=1}^{B} b_j(b_j - 1).$$

Therefore,

$$A_T = T + 2\sum_{t=1}^{T} \tau_t = \sum_{j=1}^{B} b_j^2 = \|b\|_2^2.$$

OCO. Recall that $g_t \in \partial f_t(x^{(j)})$ for every $t \in I_j$, and let $z_j = \sum_{t\in I_j} g_t$. If only the aggregate first-order feedback z_j is returned at the end of batch j, we may equivalently feed the delayed algorithm b_j copies of $\bar{g}_j = \frac{z_j}{b_j}$, all arriving at time t_j. Since $\|z_j\|_2 \le Gb_j$, we have $\|\bar{g}_j\|_2 \le G$. Moreover, for every $u \in \mathcal{K}$, convexity gives

$$\sum_{t=1}^{T} \Big(f_t(x^{(t)}) - f_t(u)\Big) \le \sum_{j=1}^{B} \langle z_j, x^{(j)} - u\rangle = \sum_{j=1}^{B} \sum_{t\in I_j} \langle \bar{g}_j, x^{(j)} - u\rangle.$$

Thus, the batched OCO regret is upper bounded by the regret of the corresponding delayed OLO instance. Equip $\mathcal{K}$ with the Euclidean norm and take, for any fixed $x_0 \in \mathcal{K}$,

$$\psi(x) = \frac{1}{2}\|x - x_0\|_2^2.$$

Then ψ is 1-strongly convex and $S \le \frac{D^2}{2}$. Applying Lemma 11 and $A_T = \|b\|_2^2$ gives

$$\mathrm{Reg} \le 2G\sqrt{\frac{D^2}{2}\|b\|_2^2} = \mathcal{O}\left(DG\|b\|_2\right).$$

Experts Problem. Let $z_j = \sum_{t\in I_j} l_t$ be the aggregate expert-loss vector in batch j. As above, at time t_j we may feed the delayed algorithm b_j copies of $\bar{l}_j = \frac{z_j}{b_j}$. Since $l_t \in [0,1]^N$, we have $\bar{l}_j \in [0,1]^N$ and $\|\bar{l}_j\|_\infty \le 1$. This transformation preserves the regret:

$$\mathrm{Reg} = \sum_{j=1}^{B} \langle p_j, z_j\rangle - \min_{p\in\Delta_N} \sum_{j=1}^{B} \langle p, z_j\rangle = \sum_{j=1}^{B} \sum_{t\in I_j} \langle p_j, \bar{l}_j\rangle - \min_{p\in\Delta_N} \sum_{j=1}^{B} \sum_{t\in I_j} \langle p, \bar{l}_j\rangle.$$

Table 1: Comparison of guessed and actual parameters across epochs.

Epoch (r)	**Guessed** ($g^{(r)}$)	**Actual** ($a^{(r)}$)
1	b_1^2	b_2^2
2	$b_1^2 + b_2^2$	$b_3^2 + b_4^2$
3	$b_1^2 + \ldots + b_4^2$	$b_5^2 + \ldots + b_8^2$
$\cdots$	$\cdots$	$\cdots$
R	$\sum_{j=1}^{2^{R-1}} b_j^2$	$\sum_{j=2^{R-1}+1}^{2^R} b_j^2$

Equip Δ_N with the l_1-norm and take

$$\psi(p) = \sum_{i=1}^{N} p_i \log p_i + \log N.$$

By Pinsker's inequality, ψ is 1-strongly convex with respect to $\|\cdot\|_1$, and

$$\sup_{p \in \Delta_N} \psi(p) = \log N.$$

Since the dual norm of l_1 is l_∞, applying Lemma 11 with $G = 1$ gives

$$\mathrm{Reg} \le 2\sqrt{\log N}\,\|b\|_2 = \mathcal{O}\left(\|b\|_2\sqrt{\log N}\right).$$

I Naive Doubling Trick

As Intuitively shown in Table 1, for each doubling epoch r, Algorithm 5 predicts the sum of squared known batch to be the actual sum of squared batch in epoch r, i.e., $\sum_{j=1}^{2^{r-1}} b_j^2 = \sum_{j=2^{r-1}+1}^{2^r} b_j^2$.

Algorithm 5 Naive Doubling Trick under Adversarial Batched Setting

Play the uniform distribution until the first block ends (i.e., at time b_1);
$r \leftarrow 1$; ▷ start epoch 1
while $2^r \le B$ **do**
  $g^{(r)} \leftarrow \sum_{j=1}^{2^{r-1}} b_j^2$; ▷ guess squared 2-norm in epoch r
  $\eta^{(r)} \leftarrow \sqrt{\frac{8 \ln N}{g^{(r)}}}$; ▷ set learning rate
  Run Hedge on blocks $2^{r-1}+1, \ldots, 2^r$ with learning rate $\eta^{(r)}$;
  $r \leftarrow r + 1$; ▷ start a new epoch
end while

Example 1. *Let $m \ge 2$ be an integer power of 2. Consider a batch structure with $B = 2m$ batches:*

$$b_1 = \cdots = b_m = 1,\ b_{m+1} = \cdots = b_{2m} = m,$$

and note that $\|b\|_2 = \sqrt{m^3 + m} = \Theta(m^{3/2})$.

Consider an instance with $N = 2$ experts. For simplicity, let both experts' per-round losses to 0 in small blocks $(b_1, \ldots, b_m)$ so that the first half of the game only serves to fool the naive algorithm. For the remaining m blocks, the losses alternate between the two experts. Specifically, for each $k = 1, 2, \ldots, m$, block $j = m + k$ satisfies

$$\left(L_1^{(m+k)},\, L_2^{(m+k)}\right) = \begin{cases} (0,\, m), & \text{if } k \text{ is odd}, \\ (m,\, 0), & \text{if } k \text{ is even}. \end{cases}$$

Proof of Proposition 2. Let R be the index of the last epoch in Algorithm 5. We argue that by the construction of Example 1, the actual number of blocks a in epoch R is exactly the second half of the blocks, i.e., $a^{(R)} = \sum_{j=m+1}^{2m} m^2 = m^3$. On the other hand, Algorithm 5 guesses the number of blocks in the last epoch to be $g^{(R)} = \sum_{j=1}^{m} 1^2 = m$, which largely deviates from $a^{(R)}$. The

corresponding learning rate for epoch R is then set to $\eta^{(R)} = \sqrt{\frac{8 \ln 2}{m}}$. In the following, we omit the superscript and write η for the learning rate in the last epoch for simplicity.

Since both actions have zero losses in the first m blocks, we have $w_1^{(m+1)} = w_2^{(m+1)} = 1$ at the beginning of block $m+1$. Since $\left(L_1^{(m+1)}, L_2^{(m+1)}\right) = (0, m)$, Algorithm 5 incurs a loss of $\frac{m}{2}$, and updates the weights to $w_1^{(m+2)} = e^{-\eta \cdot 0} = 1$ and $w_2^{(m+2)} = e^{-\eta \cdot m} = e^{-\beta}$, where we use the shorthand $\beta := m \cdot \eta = \sqrt{8m \ln 2}$.

Then, in block $m+2$, the algorithm puts a much higher probability (of $p_1^{(m+2)} = \frac{1}{1+e^{-\beta}}$) on expert 1, and incurs a loss of $\frac{m}{1+e^{-\beta}}$. After that, the weights of the two experts are equal ($w_1^{(m+3)} = w_2^{(m+3)} = e^{-\beta}$), so the next block $m+3$ is identical to block $m+1$.

Repeating this pattern, there are $\frac{m}{2}$ blocks (namely, blocks $m+1, m+3, \ldots$) on which Algorithm 5 incurs a loss of $\frac{m}{2}$. On each of the remaining $\frac{m}{2}$ blocks, the loss is $\frac{m}{1+e^{-\beta}}$. Note that either experts has a total loss of $\frac{m}{2} \cdot m = m^2/2$ over the entire horizon. Therefore, the regret is given by

$$\frac{m}{2} \cdot \frac{m}{2} + \frac{m}{2} \cdot \frac{m}{1+e^{-\beta}} - \frac{m^2}{2} = Cm^2 = \Omega(m^2),$$

where $C = \frac{1}{4} \cdot \frac{1-e^{-\beta}}{1+e^{-\beta}} = \frac{1}{4} \cdot \frac{1-e^{-\sqrt{8m \ln 2}}}{1+e^{-\sqrt{8m \ln 2}}} = \Omega(1)$ for all sufficiently large m.

On the other hand, the desired regret bound from Theorem 5 is

$$\mathcal{O}(\|b\|_2 \sqrt{\log N}) = \mathcal{O}(m^{3/2})$$

on this instance. This completes the proof. □

I.1 Adaptive Doubling Trick Sidesteps the Hard Instance

We further instantiate Theorem 7 on the hard instance from Example 1, showing that the adaptive doubling trick avoids the linear-regret failure mode of the naive doubling trick.

Corollary 1. *For the adversarially batched experts instance constructed in Example 1, the naive doubling-trick adaptation of Hedge incurs expected regret $\Omega(T)$, whereas the adaptive doubling trick described in Algorithm 2 achieves a regret of $\mathcal{O}\left(T^{3/4}\right)$.*

Proof. We consider the case of Example 1, where Proposition 2 has shown that the naive doubling trick has a regret of $\Omega(T)$. Applying Theorem 7, we have

$$\mathrm{Reg}_N^B = \mathcal{O}\left(\|b\|_2 \sqrt{\log N}\right) = \mathcal{O}\left(\sqrt{m \cdot m^2 \log 2}\right) = \mathcal{O}\left(m^{\frac{3}{2}}\right).$$

Taking $T \asymp m^2$ back into the equation completes the proof. □