# Bounded Memory Adversarial Bandits with Composite Anonymous Delayed Feedback*

**Zongqi Wan**[1,2], **Xiaoming Sun**[1,2] and **Jialin Zhang**[1,2†]

[1]Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[2]University of Chinese Academy of Sciences, Beijing, China
{wanzongqi20s,sunxiaoming,zhangjialin}@ict.ac.cn

## Abstract

We study the adversarial bandit problem with composite anonymous delayed feedback. In this setting, losses of an action are split into $d$ components, spreading over consecutive rounds after the action is chosen. And in each round, the algorithm observes the aggregation of losses that come from the latest $d$ rounds. Previous works focus on oblivious adversarial setting, while we investigate the harder non-oblivious setting. We show non-oblivious setting incurs $\Omega(T)$ pseudo regret even when the loss sequence is bounded memory. However, we propose a wrapper algorithm which enjoys $o(T)$ policy regret on many adversarial bandit problems with the assumption that the loss sequence is bounded memory. Especially, for $K$-armed bandit and bandit convex optimization, we have $\tilde{\mathcal{O}}(T^{2/3})$ policy regret bound. We also prove a matching lower bound for $K$-armed bandit. Our lower bound works even when the loss sequence is oblivious but the delay is non-oblivious. It answers the open problem proposed in [Wang *et al.*, 2021], showing that non-oblivious delay is enough to incur $\tilde{\Omega}(T^{2/3})$ regret.

## 1 Introduction

Multi-armed bandit is a widely studied problem. It can be formulated by a multi-rounds game between two players, an adversary and a learner. In $t$-th round, the adversary assigns each action $a \in [K]$ a loss $l_t(a)$, and simultaneously, the learner chooses an action $a_t \in [K]$. It then incurs loss $l_t(a_t)$. The learner can observe the loss this round of the action it just chose (i.e., $l_t(a_t)$), but can not observe the loss of other actions. This is so-called *bandit feedback*. Learner's goal is to minimize its total loss, We usually choose a posteriori best fixed action as the comparison, so that the goal is to minimize the *expected regret*, defined as

$$\mathbb{E}[R_T] = \mathbb{E}\left[\sum_{t=1}^{T} l_t(a_t) - \min_{y\in\mathcal{A}} \sum_{t=1}^{T} l_t(y)\right]$$

*The full version of this paper can be found in arXiv:2204.12764

†Contact Author

Bandits problem has a wide range of applications in the industry, including medical trials, recommendation systems, computational ads., tree search algorithms..([Kocsis and Szepesvári, 2006; Chapelle *et al.*, 2014; Villar *et al.*, 2015; Silver *et al.*, 2016; Lei *et al.*, 2017]). In the standard formulation of the bandit problem, each round the learner observes the precise feedback immediately, and adjusts its strategy afterward according to the immediate feedback. However, in many real-world situations, this assumption can not be satisfied. The total impact of an action may not be observed immediately. In contrast, the impact may spread over an interval after the action has been played. For instance, consider the advertising problem. People do not always click on the website or buy the product after seeing the ads immediately, and the feedback(i.e., click number) the recommender observed may be the aggregation of the impact of several ads recommended before.

To address the above scenarios, [Pike-Burke *et al.*, 2018] proposed a stochastic bandit model with anonymous feedback. In their setting, round $t$ is related with a delay time $d(t)$, which is drawn from some i.i.d distribution, and the feedback learner can observe at the end of round $t$ is the aggregation $\sum_{s+d(s)=t} l_s(a_s)$. They showed that there is a learner achieving $\tilde{\mathcal{O}}\left(\sqrt{KT} + K\mathbb{E}(d)\right)$ expected regret. [Cesa-Bianchi *et al.*, 2018] generalized the model to adversarial bandits, where the loss in their model is a composition of constant parts, $l_t(a) = \sum_{s=0}^{d-1} l_t^{(s)}(a)$ where $l_t^{(s)}(a)$ means the part of loss which will delay $s$ rounds. The feedback learner observes at the end of $t$-th round is $l_t^o(a_t) = \sum_{s=0}^{d-1} l_{t-s}^{(s)}(a_{t-s})$. In their model, the loss sequence and delay are both oblivious, which means they can not be adjusted according to the learner's action history. They proposed a mini-batch wrapper algorithm, converting a standard bandit algorithm to the one that can handle composite anonymous delayed feedback. They applied this wrapper on EXP3 algorithm[Auer *et al.*, 2002], achieving $\tilde{\mathcal{O}}(\sqrt{dKT})$ expected regret of multi-armed bandit problem with composite anonymous feedback. Their wrapper can also be applied on bandit convex optimization problem [Flaxman *et al.*, 2005], in which the action set is a convex body, and loss functions are bounded convex functions on this convex body.They applied their wrapper on the algorithm proposed in [Saha and Tewari, 2011],

achieving $\tilde{\mathcal{O}}(d^{1/3}(KT)^{2/3})$ regret. Subsequently, [Wang *et al.*, 2021] studied the situation that delay is determined by a non-oblivious adversary. That is, though the loss sequence is oblivious, an adversary can split the loss into parts according to the learner's action history. In this setting, they modified standard EXP3 algorithm so that it achieves $\tilde{\mathcal{O}}((d+\sqrt{K})T^{2/3})$ regret for $K$-armed bandit. Different from other previous algorithms, it does not require any prior knowledge of delay $d$. Though they believed that their algorithm is asymptotic optimal for the adversary with oblivious loss and non-oblivious delay, we only have a $\Omega(\sqrt{T})$ regret lower bound from the classical multi-armed bandit problem. How to derive a matching regret lower bound is one of the future research problems in their work.

The existing works on composite anonymous feedback setting all assume that the loss sequence is oblivious, which does not always hold in the real world. For example, consider the case that one is involved in a repeated game with other players, it is natural that others will adjust their strategies according to his action history. This will make the loss of each pure strategy non-oblivious. In $K$-armed bandit and bandit convex optimization problem without delay, even if we consider non-oblivious loss, we still have $\tilde{\Theta}(\sqrt{T})$ pseudo regret. However, things become very different when we consider composite anonymous delayed feedback.

**Contribution.** We studied the bandits with composite anonymous feedback under *non-oblivious* setting. In our model, we allow both non-oblivious delay and loss sequences. Since when the loss sequence is non-oblivious, the common regret can be generalized to different performance metrics. We first discuss which performance metric to use in our setting. We show that any learner can not achieve sublinear *external pseudo regret* under our non-oblivious setting. Inspired by [Arora *et al.*, 2012], we turn to a more reasonable metric called *policy regret*. In non-oblivious setting, policy regret is $\Omega(T)$ even without delayed feedback. So [Arora *et al.*, 2012] considered a weaker adversary which has *bounded memory*. They proved $o(T)$ policy regret bounds for many bandit problems with bounded memory assumption. Especially, they proved $\mathcal{O}(T^{2/3})$ policy regret bounds for $K$-armed bandit. Different from the pseudo regret metric, we find that the policy regret does not get worse with the introduction of delay. We prove that the simple mini-batch wrapper can generate $o(T)$ *policy regret* algorithms for many bandit problems with non-oblivious delays and *bounded memory* loss sequence in composite anonymous feedback setting. Especially, it can generate $\mathcal{O}(T^{2/3})$ policy regret algorithms for $K$-armed bandits and bandit convex optimization in our setting. Moreover, this mini-batch wrapper does not require any prior knowledge of $d$. Meanwhile, pseudo regret is still $\Omega(T)$ even when we restrict the loss sequence to be bounded memory. Since policy regret is the same as common regret in the oblivious setting, our upper bound can be seen as a generalization of the result in [Wang *et al.*, 2021]. Furthermore, we prove a matching lower bound for our model. In fact, we prove a stronger lower bound. Even if the loss sequence is generated by an oblivious adversary, any learner can only obtain $\tilde{\Omega}(T^{2/3})$ regret. Our lower bound answered the problem proposed in [Wang *et al.*, 2021], showing that non-oblivious delay on its own is enough to cause $\tilde{\Theta}(T^{2/3})$ regret.

To summarize the above results, our study provides a complete answer to the non-oblivious composite anonymous feedback setting.

**More Related Work.** Delay setting was first considered in [Gergely Neu *et al.*, 2010], in which they assumed each feedback is delayed by a constant $d$, and they posed a $\tilde{\mathcal{O}}(\sqrt{dKT})$ regret algorithm. [Joulani *et al.*, 2013] generalized this result to the partial monitoring setting. [Quanrud and Khashabi, 2015] first considered the non-uniform delay setting, where the delay size of each round can be different. Let $D$ be the sum of all delay sizes, they proved a $\mathcal{O}(\sqrt{(D+T)\log K})$ regret bound in *full feedback* online learning setting. In recent years, a series of works have generalized this work to the bandit feedback setting and keep developing more instance-dependent upper bound([Li *et al.*, 2019; Thune *et al.*, 2019; Bistritz *et al.*, 2019; Zimmert and Seldin, 2020; Gyorgy and Joulani, 2021; Cella and Cesa-Bianchi, 2020]). All above works assumed non-anonymous and separated feedback. That is, the learner observes every single feedback rather than their aggregation. Besides, [Li *et al.*, 2019] studied an unknown delay setting where the learner does not know which round the feedback comes from. Their feedback is also separated, and the learner knows which action is related to the feedback.

Policy regret and bounded memory assumption are proposed in [Arora *et al.*, 2012]. They are used in many online learning and bandit literature, including [Anava *et al.*, 2015; Heidari *et al.*, 2016; Arora *et al.*, 2018; Jaghargh *et al.*, 2019].

## 2 Model Setting

Adversarial bandit problem is a repeated game between a learner and an adversary. There are $T$ rounds in this game. In $t$-th round, the learner chooses an action $a_t \in \mathcal{A}$, the adversary chooses a function $l_t \in \mathcal{L}$ mapping $\mathcal{A}$ to a normalized bounded interval $[0, 1]$. Then the learner incurs loss $l_t(a_t)$. The learner's target is to minimize its expected cumulative regret.

Here we formulate two classical bandit problems into the instance of adversarial bandit problem.

**Example 1** ($K$-armed bandit)**.** *$K$-armed bandit is a special case of adversarial bandit problem where $\mathcal{A} = [K] \triangleq \{1, 2, \cdots, K\}$. And $\mathcal{L}$ is the set of all functions mapping $[K]$ to $[0, 1]$.*

**Example 2** (bandit convex optimization)**.** *Bandit convex optimization is also a particular case of adversarial bandit problem when $\mathcal{A}$ is a convex body of $\mathbb{R}^K$. And $\mathcal{L}$ contains all $L$-Lipschitz convex function where $L$ is a constant.*

**Adversary setting.** Our setting allows the adversary to be non-oblivious, which means that it can choose functions $l_t$ according to the action history $a_1, a_2, \cdots, a_{t-1}$ of the learner. To formalize this setting, we can think that $l_t$ takes the whole action histories sequence $A_t = (a_1, a_2, \cdots, a_t)$ as its input. Under this viewpoint, we can assume that $l_t$ is determined before the game starts. And $l_t(A_t) \in \mathcal{L}$ means if we fix $a_1, \cdots, a_{t-1}$, it belongs to $\mathcal{L}$.

**Delay setting.** In this paper, we consider the composite anonymous delayed feedback setting proposed by [Cesa-Bianchi *et al.*, 2018]. In this setting, the adversary can split the loss function into $d$ components arbitrarily, where $d$ is a constant. That is $l_t(A_t) = \sum_{s=0}^{d-1} l_t^{(s)}(A_t)$. We also allow this splitting process to be non-oblivious, which means $l_t^{(s)}$ can be chosen according to the action histories $A_t$. At the end of $t$-th round, after the algorithm has made its decision this round, the algorithm can only observe $l_t^o(A_t) = \sum_{s=0}^{d-1} l_{t-s}^{(s)}(A_{t-s})$, but can not figure out how $l_t^o(A_t)$ is composed.

**Pseudo-regret and policy regret.** In non-oblivious setting, the most common metric of the performance of a learner is *external pseudo-regret*. Defined as

$$R_T^{pseudo} = \sum_{t=1}^{T} l_t(A_t) - \min_{y \in \mathcal{A}} \sum_{t=1}^{T} l_t(A_{t-1}, y) \tag{1}$$

Though *external pseudo-regret* is widely used, its meaning is quite strange for the non-oblivious setting because if the learner actually chooses $y$ every round, the loss he gets is not $\sum_{t=1}^{T} l_t(A_{t-1}, y)$ but $\sum_{t=1}^{T} l_t(y, y, \cdots, y)$. This fact inspires people to design more meaningful metrics in non-oblivious setting. [Arora *et al.*, 2012] proposed a new metric called *policy regret* which is defined as

$$R_T^{policy} = \sum_{t=1}^{T} l_t(A_t) - \min_{y \in \mathcal{A}} \sum_{t=1}^{T} l_t(y, \cdots, y) \tag{2}$$

Policy regret captures the fact that learner's different action sequences will cause different loss sequences. We believe that policy regret is a more reasonable metric compared with external pseudo regret. Both pseudo-regret and policy regret are the same as standard regret definition when the loss is oblivious.

[Arora *et al.*, 2012] shows that policy regret has a $\Omega(T)$ lower bound against the non-oblivious adversary in $K$-armed bandits without delayed feedback. So they consider a weaker adversary which has bounded memory.

**Definition 1** ($m$-bounded memory)**.** *A non-oblivious loss sequence $l_t$ is called $m$-*bounded memory *if for any action sequence $a_1, a_2, \cdots, a_t$ and $a'_1, a'_2, \cdots, a'_{t-m-1}$, $l_t(a_1, a_2, \cdots, a_t) = l_t(a'_1, a'_2, \cdots, a'_{t-m-1}, a_{t-m}, \cdots, a_t)$ holds. We call a non-oblivious loss sequence is* bounded memory *iff $m$ is a constant.*

From the definition, 0-bounded memory loss sequence is an oblivious loss sequence. [Arora *et al.*, 2012] proved a $\tilde{\mathcal{O}}(K^{1/3}T^{2/3})$ policy regret upper bound for no delay setting under the assumption that the adversary is bounded memory. In this paper, we also assume that the loss sequence is bounded memory. However, we do not restrict the memory of delay adversary. As an example, the model in [Wang *et al.*, 2021] can be seen as the same as our model with 0-bounded memory loss sequence.

In our non-oblivious composite anonymous feedback setting, external pseudo-regret has a $\Omega(T)$ lower bound even when the loss sequence is generated by an 1-bounded memory adversary, which means this setting is not learnable under the external pseudo-regret metric. Formally, we prove the following theorem.

**Algorithm 1** Mini-batch wrapper

**Input:** Black-box bandit algorithm $\mathcal{B}$, batch size $\tau$, time horizon $T$

$j \leftarrow 1$
**while** not end **do**
  **if** there are less than $\tau$ rounds **then**
    choose arbitrary action in remaining rounds.
  **else**
    query $\mathcal{B}$ for the next action $z_j \in \mathcal{A}$
    choose $z_j$ for consecutive $\tau$ rounds, collect feedback $l_t^o$ for $t \in [(j-1)\tau + 1, j\tau]$
    feed $\mathcal{B}$ with $\hat{l}_j = \min\left\{\frac{1}{\tau}\sum_{t=(j-1)\tau+1}^{j\tau} l_t^o, 1\right\}$ as the feedback of action $z_j$
    $j \leftarrow j + 1$

**Theorem 1.** *In composite anonymous feedback setting, there is a 2-armed bandit with non-oblivious adversary, such that any learner incurs $\Omega(T)$ expected pseudo-regret. Moreover, the loss sequence is 1-bounded memory.*

As we can see from the next section, though our setting is not learnable under the pseudo-regret metric, it is learnable under the policy regret metric with the assumption that loss sequence is bounded memory.

## 3 Upper Bound

In this section, we prove that by applying a mini-batch wrapper, one can convert any standard non-oblivious bandit algorithm to an algorithm that can handle non-oblivious adversary with composite anonymous delayed feedback.

Intuitively, when the learner chooses an action different from the last round, the feedback it observes in the interval of $d$ rounds after that can not reflect the true losses of the actions it chooses. In other words, the learner suffers from observing inaccurate feedback during a $d$ rounds interval immediately when it switches its chosen action. To obtain accurate feedback for decisions, a learner can not switch its chosen action frequently. A natural approach is to apply a mini-batch wrapper on a standard bandit algorithm. That is, we divide all $T$ rounds into $\lceil T/\tau \rceil$ batches where $\tau$ is the batch size. Each batch contains $\tau$ consecutive rounds except the last round, which may contain fewer than $\tau$ rounds. At the beginning of the $j$-th batch, we receive an action $z_j \in \mathcal{A}$ from the black-box algorithm and keep choosing $z_j$ during this batch. At the end of the $j$-th batch, we feed the average loss observed in this batch to the black-box algorithm. Since the black-box algorithm can only receive a $[0, 1]$ loss, we feed the minimal between average loss and 1. See Algorithm 1 for the pseudo-code.

By applying the mini-batch wrapper, times of the learner's action switching can be controlled by $\mathcal{O}(T/\tau)$. In each batch, the learner suffers the inaccurate feedback for constant rounds so that it only curses a constant feedback error, which will add to the regret finally. If we set the batch size to be a relatively

large quantity such that $T/\tau = o(T)$, it is possible that we control the regret to $o(T)$.

**Theorem 2.** *Suppose we have a bandit algorithm $\mathcal{B}$ for no delay setting which achieves $R(J)$ expected pseudo regret when the time horizon is $J$ and the loss sequence is generated by a non-oblivious adversary. Assume $\tau > \max\{d, m\}$ and the loss sequence is $m$-bounded, then the mini-batch wrapper (Algorithm 1) can achieve policy regret as follows for the composite anonymous feedback setting.*

$$\mathbb{E}[R_T^{policy}] \leq \tau R(\lfloor T/\tau \rfloor) + \mathcal{O}(\max\{m, d\}T/\tau) + \mathcal{O}(\tau)$$

*Proof Sketch.* Without loss of generality, we can assume $T/\tau$ is an integer, otherwise it only produces an extra $\mathcal{O}(\tau)$ term in the expected policy regret bound. Since $z_j$ is the action of $j$-th batch, we have $a_{(j-1)\tau+1} = a_{(j-1)\tau+2} = \cdots = a_{j\tau} = z_j$. Let $Z_j$ be the sequence $(z_1, z_2, \cdots, z_j)$. Due to the pseudo regret bound of the black box algorithm, we have

$$\mathbb{E}\left[\max_{y\in\mathcal{A}} \sum_{j=1}^{T/\tau} \left(\hat{l}_j(Z_j) - \hat{l}_j(Z_{j-1}, y)\right)\right] \leq R(T/\tau)$$

In each batch, the black-box algorithm receives at most $d$ total loss from the previous batch, so $\frac{1}{\tau}\sum_{t=(j-1)\tau+1}^{j\tau} l_t^o$ can not exceed 1 by $\mathcal{O}(d/\tau)$. And this gives

$$\mathbb{E}\left[\max_{y\in\mathcal{A}} \sum_{j=1}^{T/\tau} \sum_{t=(j-1)\tau+1}^{j\tau} \left(l_t^o(A_t) - l_t^o(A_{(j-1)\tau}, y^{t-(j-1)\tau})\right)\right] \leq \tau R(T/\tau) + \mathcal{O}(d)$$

Where $y^t$ is the $t$-repetition $(y, \cdots, y)$ sequence of $y$. Next, we aim to replace $l_t^o$ with $l_t$. It's easy to see $\left|\sum_{t=1}^T l_t(A_t) - \sum_{t=1}^T l_t^o(A_t)\right| \leq \mathcal{O}(d)$. For another term, our goal is to bound the distance between $L_j^o \triangleq \sum_{t=(j-1)\tau+1}^{j\tau} l_t^o\left(A_{(j-1)\tau}, y^{t-(j-1)\tau}\right)$ and $L_j \triangleq \sum_{t=(j-1)\tau+1}^{j\tau} l_t\left(A_{(j-1)\tau}, y^{t-(j-1)\tau}\right)$. For $j$-th batch, if $t$ is in the first $d-1$ rounds in this batch, $l_t^o(A_{(j-1)\tau}, y^{t-(j-1)\tau})$ may contain some loss coming from the previous batch, which is not in $L_j$. This quantity is at most 1. On the other hand, if $t$ is in the last $d-1$ rounds this batch, $l_t(A_{(j-1)\tau}, y^{t-(j-1)\tau})$ contains some loss delayed to the next batch. That means $|L_j - L_j^o| \leq \mathcal{O}(d)$. We have

$$\mathbb{E}\left[\max_{y\in\mathcal{A}} \sum_{j=1}^{T/\tau} \sum_{t=(j-1)\tau+1}^{j\tau} \left(l_t(A_t) - l_t(A_{(j-1)\tau}, y^{t-(j-1)\tau})\right)\right] \leq \tau R(T/\tau) + \mathcal{O}(d) + \mathcal{O}\left(\frac{dT}{\tau}\right)$$

Note that it's enough to replace $l_t(A_{(j-1)\tau}, y^{t-(j-1)\tau})$ with $l_t(y^t)$ to fit the form of policy regret. Since $l_t$ is $m$-bounded memory, if $t - (j-1)\tau \geq m$ then $l_t(y^t) = l_t(A_{(j-1)\tau}, y^{t-(j-1)\tau})$. In each batch, there are at most $m$ rounds do not satisfy this inequality. They will cause a difference at most $m$ between $\sum_{t=(j-1)\tau+1}^{j\tau} l_t(A_{(j-1)\tau}, y^{t-(j-1)\tau})$ and $\sum_{t=(j-1)\tau+1}^{j\tau} l_t(y^t)$. This gives

$$\mathbb{E}\left[\max_{y\in\mathcal{A}} \sum_{j=1}^{T/\tau} \sum_{t=(j-1)\tau+1}^{j\tau} \left(l_t(A_t) - l_t(y^t)\right)\right] \leq \tau R(T/\tau) + \mathcal{O}(d) + \mathcal{O}\left(\max\{m, d\}T/\tau\right)$$

Add the extra $\mathcal{O}(\tau)$ regret due to the assumption that $T/\tau$ is an integer, we get the final bound. □

By applying the mini-batch wrapper on some bandit algorithms, we obtain algorithms that can handle composite anonymous delayed feedback setting. Firstly, we apply Theorem 2 on $K$-armed bandit problem. [Auer *et al.*, 2002] proposed a well known algorithm $EXP3$ which guarantees $\tilde{\mathcal{O}}(\sqrt{KT})$ expected pseudo regret. Employing this algorithm, we have the following corollary.

**Corollary 1.** *For $K$-armed bandit problem, if we apply Algorithm 1 on $EXP3$ algorithm, and set batch size $\tau = K^{-1/3}T^{1/3}$, we have the expected policy regret satisfies*

$$\mathbb{E}[R_T^{policy}] \leq \tilde{\mathcal{O}}(\max\{m, d\}K^{1/3}T^{2/3})$$

For bandit convex optimization, [Bubeck *et al.*, 2017] proposed an algorithm using kernel method, and it can guarantee $\tilde{\mathcal{O}}(K^{9.5}\sqrt{T})$ expected pseudo regret, where $K$ is the dimension of the action space $\mathcal{A}$. By employing this algorithm as our black-box algorithm $\mathcal{B}$, we have the following corollary.

**Corollary 2.** *For bandit convex optimization, if we apply Algorithm 1 on the algorithm in [Bubeck* et al.*, 2017], and set batch size $\tau = K^{-19/3}T^{1/3}$, we have the expected policy regret satisfies*

$$\mathbb{E}[R_T^{policy}] \leq \tilde{\mathcal{O}}(\max\{m, d\}K^{19/3}T^{2/3})$$

Other algorithms for composite anonymous feedback setting are also some kinds of mini-batch wrapper, such as CLW in [Cesa-Bianchi *et al.*, 2018] and ARS-EXP3 in [Wang *et al.*, 2021]. ARS-EXP3 uses increasing batch sizes on EXP3 algorithm. CLW is a wrapper algorithm that can be applied on many normal bandit algorithms, and it uses a constant batch size. As we discussed above, every action switching may incur extra constant regret. At first, it seems that CLW will incur $\Omega(T)$ regret since CLW performs $\Omega(T)$ action switches. However, thanks to the oblivious adversary they assumes, CLW can randomize the batch size to fool the oblivious adversary, such that they can still reach $\mathcal{O}(\sqrt{T})$ regret. Nonetheless, this randomizing technique does not work when delay is non-oblivious since the adversary here can read learner's random bits realized before the current round.

## 4 Lower Bound

In this section, we prove a matching lower bound for $K$-armed bandit. The main result of [Dekel *et al.*, 2014] actually implies a $\tilde{\Omega}(T^{2/3})$ policy regret lower bound for $K$-armed bandit without delayed feedback. However, their lower bound depends on constructing a non-oblivious loss sequence. Our lower bound here is stronger since the loss sequence in our

construction is oblivious. The lower bound shows that the non-oblivious delay adversary on its own is enough to incur $\tilde{\Theta}(T^{2/3})$ regret without the help of non-oblivious loss.

**Theorem 3.** *For $K$-arms bandit with non-oblivious delays and oblivious loss sequence, we have*

$$\mathbb{E}[R_T] = \tilde{\Omega}(K^{1/3}T^{2/3})$$

Note that we use the notation $R_T$ rather than $R_T^{policy}$, since in Theorem 3, the loss sequence is oblivious, and the policy regret is the same as normal regret.

According to Yao's minimax principle, to prove Theorem 3, it is enough to construct a distribution over some loss sequences and a deterministic delay adversary, such that any deterministic learner can only achieve $\tilde{\Omega}(K^{1/3}T^{2/3})$ expected regret. Before we discuss our detailed construction, we firstly describe our intuition briefly. We construct the loss sequence based on a random walk. The best arm always has the loss $\epsilon$ lower than the random walk, while others always have the loss equal to the random walk. This construction can force the learner to switch between arms since the learner only observes a random walk and obtain no information if he keeps choosing one arm. The delay adversary we construct makes the observed losses in one round are the same no matter which arm is chosen in this round. Therefore, the learner can only get information from the changes of observed loss between rounds. Our delay construction makes the changes of the observed loss as consistent as possible with the changes of the random walk. So it happens only a few times that the observed loss sequence deviates from the random walk. We bound the information learner can get in this deviation through a KL divergence argument. Thus, the total information learner get is so low that it can not help the learner achieve low regret. However, a random walk can drift so much that it jumps out of $[0, 1]$, making the construction of the loss sequence invalid. To address this problem, we borrow the idea of multi-scale random walk from [Dekel *et al.*, 2014]. Multi-scale random walk is a trade-off between random walk and i.i.d samples. Its drift can be bounded in an acceptable range while still maintaining the low information nature of the random walk.

To clarify our constructional proof, we describe the construction of the loss sequence in Section 4.1, and the construction of the delay adversary is in Section 4.2. We sketch the proof of Theorem 3 in Section 4.3.

### 4.1 Construction of Loss Sequence

In this section, we describe a stochastic process called *multi-scale random walk* proposed in [Dekel *et al.*, 2014]. It will be used to generate the loss sequence.

Let $\{\xi_t\}$ be i.i.d samples which obey gaussian distribution $\mathcal{N}(0, \sigma^2)$, where $\sigma^2$ is the variance to be determined later. Let $\rho : [T] \to \{0\} \cup [T]$ be the *parent function*. The function $\rho$ assigns each round $t$ a parent $\rho(t)$. We restrict that $\rho(t) < t$, and define

$$W_t = W_{\rho(t)} + \xi_t$$
$$W_0 = 0$$

Then $W_t$ is a stochastic process. We next define the width of this stochastic process.

**Definition 2** (cut and width)**.** *The* cut *of a parent function $\rho$ at $t$ is*

$$cut_\rho(t) = \{s \in [T] | \rho(s) \le t < s\}$$

*width of $\rho$ is $\omega(\rho) = \max_{t\in[T]} |cut_\rho(t)|$*

We then introduce a stochastic process called *multi-scale random walk*.

**Definition 3** (multi-scale random walk)**.** *Let parent function be $\rho^*(t) = t - 2^{\delta_t}$ where $\delta(t) = \max\{i \ge 0 | 2^i \text{ divides } t\}$. Then the stochastic process $W_t$ equipped with parent function $\rho^*$ is called multi-scale random walk.*

Multi-scale random walk has the special property that it is not too wide and its drift range is not too large. This property is formalized in following lemmas.

**Lemma 1** (Lemma 2 from [Dekel *et al.*, 2014])**.** *The width of multi-scale random walk is bounded by $\lfloor \log_2 T \rfloor + 1$.*

**Lemma 2** (Lemma 1 from [Dekel *et al.*, 2014])**.** *Let $W_t$ be the multi-scale random walk, then $\forall \delta \in (0, 1)$*

$$\mathbb{P}\left(\max_{t\in[T]} |W_t| \le \sigma\sqrt{2(\log T + 1)\log\frac{T}{\delta}}\right) \ge 1 - \delta$$

To specify the loss sequence of each arm, we define a uniform random variable $Z$ valued on $\{0\} \cup \mathcal{A}$. $Z$ tells us which arm is the best. When $Z = 0$, all arms have the same loss all the time. We define untruncated loss of arm $a \in \mathcal{A}$ be $l'_t(a) = W_t + \frac{3}{4} - \epsilon \cdot \mathbb{I}[Z = a]$. Where $\epsilon$ is also a parameter to be determined later and $W_t$ is the multi-scale random walk defined above. From now on, we will always use $W_t$ to represent the multi-scale random walk.

However, this loss may jump out of the bounded interval $[0, 1]$, so we truncate it to make sure that loss is in $[0, 1]$. For some technical reasons, we further require that loss is greater than $\frac{1}{2}$. The true loss sequence is $l_t(a) = \text{trunc}_{[\frac{1}{2},1]}(l'_t(a))$ where

$$\text{trunc}_{[a,b]}(x) = \begin{cases} a & x < a \\ b & x > b \\ x & \text{otherwise} \end{cases}$$

### 4.2 Construction of Delay Adversary

In this section, we describe our construction of delay adversary. The delay adversary is constructed so that it can mislead the learner. To achieve this goal, the delay adversary has two states: *low loss state* and *high loss state*. For $t$-th round, if delay adversary is at low loss state, it splits $l_t(a) = l_t^{(0)}(a) + l_t^{(1)}(a)$ for each arm $a \in \mathcal{A}$, so that

$$l_t^{(0)}(a) = \text{trunc}_{[\frac{1}{2},1]}(W_t + \frac{3}{4} - \epsilon) - l_{t-1}^{(1)}(a_{t-1}) \quad (3)$$

If delay adversary is at high loss state, it splits $l_t(a) = l_t^{(0)}(a) + l_t^{(1)}(a)$ for each arm $a \in \mathcal{A}$, so that

$$l_t^{(0)}(a) = \text{trunc}_{[\frac{1}{2},1]}(W_t + \frac{3}{4}) - l_{t-1}^{(1)}(a_{t-1}) \quad (4)$$

However, sometimes $l_t^{(0)}(a)$ computed by equation (3) and equation (4) might not lie in $[0, l_t(a)]$, which means the splitting is not valid. For example, if $l_{t-1}^{(1)}(a_{t-1}) = 0$, and the

delay adversary is at high loss state in round $t$, then consider the best arm $Z$, $l_t(Z) = \text{trunc}_{[\frac{1}{2},1]}(W_t + \frac{3}{4} - \epsilon)$ is strictly less than $\text{trunc}_{[\frac{1}{2},1]}(W_t + \frac{3}{4})$ when $|W_t| < \frac{1}{4}$, so $l_t^{(0)}(Z)$ computed according to (4) is greater than $l_t(Z)$, which is invalid. Similar situation happens when delay adversary is at low loss state and $l_{t-1}^{(1)}(a_{t-1}) > W_t + \frac{3}{4} - \epsilon, |W_t| < \frac{1}{4} - \epsilon$.

In order to avoid the above situation, the delay adversary will use the following procedure to switch its loss state wisely. Let $S_t$ be the state of delay adversary in round $t$. In round $t$, before splitting the true loss, delay adversary checks $l_{t-1}^{(1)}(a_{t-1})$, the loss delayed from the previous round. If $l_{t-1}^{(1)}(a_{t-1}) < \epsilon$ and $S_{t-1} =$ high loss, switch the state to low loss, i.e. $S_t =$ low loss. If $l_{t-1}^{(1)}(a_{t-1}) > \frac{1}{4} - \epsilon$ and $S_{t-1} =$ low loss, switch the state to high loss, i.e. $S_t =$ high loss. Otherwise, keep the state unchanged, $S_t = S_{t-1}$. This procedure keeps $l_t^{(1)}(a) \in [0, 1/4]$ for all $a$ and $t$. Remember we applied a $[1/2, 1]$ truncation on the true loss, this lead to $l_t^{(1)}(a) \le 1/4 \le 1/2 \le l_t(a)$, thus $l_t^{(1)} \in [0, l_t(a)], \forall t, a$. It shows that this state switching procedure is valid. Moreover, we let the delay adversary start from high loss state.

As we will show in Section 4.3, the times of state switches of the delay adversary is closely related to the information learner can get. The following lemma gives an upper bound to the times of state switches.

**Lemma 3.** *If $Z = i$, delay adversary performs at most $\frac{8\epsilon T_i}{1-8\epsilon}$ state switches, where $T_i$ denotes the number of rounds arm $i$ has been been selected.*

### 4.3 Proof Sketch of Theorem 3

The choice of a deterministic learner in $t$-th round is determined by the observed loss sequence $l_1^o(a_1), \cdots, l_{t-1}^o(a_{t-1})$ before $t$-th round. Let $\mathcal{F}_t$ be the $\sigma$-field generated by $l_1^o(a_1), \cdots, l_{t-1}^o(a_{t-1})$ and $\mathcal{F} \triangleq \mathcal{F}_T$. Let $\mathbb{P}$ be the distribution of observed loss sequence $\{l_t^o(a_t)\}_{t=1}^T$, $\mathbb{P}_t$ be the distribution of $l_t^o(a_t)$ conditioned on the observation history $l_1^o(a_1), \cdots, l_{t-1}^o(a_{t-1})$. Let $\mathbb{P}^i = \mathbb{P}(\cdot|Z = i)$ and $\mathbb{P}_t^i = \mathbb{P}_t(\cdot|\ Z = i)$. Our first step is to bound the total variation between $\mathbb{P}^i$ and $\mathbb{P}^0$, denoted by $\mathcal{D}_{TV}^{\mathcal{F}}(\mathbb{P}^0, \mathbb{P}^i)$. We show the total variation is upper bounded by the width of the parent function and the number of times the arm $i$ is chosen, see Lemma 4. Then, intuitively, if the total variation is small, which means that it is hard to distinguish distribution $\mathbb{P}^0$ and $\mathbb{P}^i$, the regret will be large.

**Lemma 4.** $\mathcal{D}_{TV}^{\mathcal{F}}(\mathbb{P}^0, \mathbb{P}^i) \le (\epsilon/\sigma)\sqrt{\frac{2\omega(\rho^*)\epsilon\mathbb{E}_{\mathbb{P}^0}[T_i]}{1-8\epsilon}}$, *where $\omega(\rho^*)$ is the width of $\rho^*$.*

*Proof Sketch.* By chain rule of KL divergence

$$\mathcal{D}_{KL}(\mathbb{P}^0||\mathbb{P}^i) = \sum_{t=1}^{T} \mathbb{E}_{\mathbb{P}^0}\left[\mathcal{D}_{KL}(\mathbb{P}_t^0||\mathbb{P}_t^i)\right]$$

For any fixed deterministic learner and condition on any realized observation sequence $l_1^o(a_1), l_2^o(a_2), \cdots, l_{t-1}^o(a_{t-1})$, let $S_t^i$ be the state of delay adversary when $Z = i$. The most important observation is $\mathcal{D}_{KL}(\mathbb{P}_t^0\|\mathbb{P}_t^i) \le \frac{\epsilon^2}{2\sigma^2}\mathbb{I}\{S_t^i \ne S_{\rho^*(t)}^i\}$. This means the learner obtains KL-divergence to distinguish the case $Z = 0$ and $Z = i$ only when $S_t^i \ne S_{\rho^*(t)}^i$. Moreover, we can bound $\sum_{t=1}^T \mathbb{I}\{S_t^i \ne S_{\rho^*(t)}^i\} \le \omega(\rho^*)\sum_{s=1}^T \mathbb{I}\{S_{s-1}^i \ne S_s^i\}$. Then we use Lemma 3 to get an upper bound of $\mathcal{D}_{KL}(\mathbb{P}^0\|\mathbb{P}^i)$. Then apply Pinsker's inequality, we get $\mathcal{D}_{TV}^{\mathcal{F}}(\mathbb{P}^0, \mathbb{P}^i) \le (\epsilon/\sigma)\sqrt{\frac{2\omega(\rho^*)\epsilon\mathbb{E}_{\mathbb{P}^0}[T_i]}{1-8\epsilon}}$. □

*Proof Sketch of Theorem 3.* To lower bound the regret, we first considers untruncated regret

$$\hat{R}_T = \sum_{t=1}^{T} l_t'(a_t) - \min_{a^* \in \mathcal{A}} \sum_{t=1}^{T} l_t'(a^*)$$

It can be lower bounded in terms of total variations between $\mathbb{P}^0$ and $\mathbb{P}^i$ for $i > 0$.

$$\mathbb{E}[\hat{R}_T] \ge \frac{\epsilon(K-1)T}{K+1} - \frac{\epsilon T}{K+1}\sum_{i=1}^{K} \mathcal{D}_{TV}^{\mathcal{F}}(\mathbb{P}^0, \mathbb{P}^i)$$

By choosing suitable $\epsilon$ and $\sigma$, we can prove $\mathbb{E}[\hat{R}_T] \ge \tilde{\Omega}(K^{1/3}T^{2/3})$ by using Lemma 4. Now we turn to true regret $R_T$. When $|W_t| \le \frac{1}{4} - \epsilon$ for any $t \in [T]$, $R_T = \hat{R}_T$. Lemma 2 tells us this happens with at least a constant probability if we choose $\sigma$ carefully. If $R_T \ne \hat{R}_T$, we use the trivial bound $R_T \ge 0$. Therefore $\mathbb{E}[R_T] \ge \tilde{\Omega}(K^{1/3}T^{2/3})$.

□

## 5 Conclusion

In this paper, we generalize the previous works of bandits with composite anonymous delayed feedback. We consider the non-oblivious loss adversary with bounded memory and the non-oblivious delay adversary. Though the external pseudo-regret incurs $\Omega(T)$ lower bound, for policy regret, we propose a mini-batch wrapper algorithm which can convert any standard non-oblivious bandit algorithm to the algorithm which fits our setting. By applying this algorithm, we prove a $\mathcal{O}(T^{2/3})$ policy regret bound on $K$-armed bandit and bandit convex optimization, generalizing the results of [Cesa-Bianchi *et al.*, 2018] and [Wang *et al.*, 2021]. We also prove the matching lower bound for $K$-armed bandit problem, and our lower bound works even when the loss sequence is oblivious.

## Acknowledgements

This work was supported in part by the Strategic Priority Research Program of Chinese Academy of Sciences under Grant No. XDA27000000, the National Natural Science Foundation of China Grants No. 61832003, 61872334.

## References

[Anava *et al.*, 2015] Oren Anava, Elad Hazan, and Shie Mannor. Online learning for adversaries with memory: price of past mistakes. In *NeurIPS*, pages 784–792, 2015.

[Arora *et al.*, 2012] Raman Arora, Ofer Dekel, and Ambuj Tewari. Online bandit learning against an adaptive adversary: from regret to policy regret. In *ICML*, pages 1747–1754, 2012.

[Arora *et al.*, 2018] Raman Arora, Michael Dinitz, Teodor V Marinov, and Mehryar Mohri. Policy regret in repeated games. In *NeurIPS*, volume 2018, pages 6732–6741, 2018.

[Auer *et al.*, 2002] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002.

[Bistritz *et al.*, 2019] Ilai Bistritz, Zhengyuan Zhou, Xi Chen, Nicholas Bambos, and Jose Blanchet. Exp3 learning in adversarial bandits with delayed feedback. In *NeurIPS*, 2019.

[Bubeck *et al.*, 2017] Sébastien Bubeck, Yin Tat Lee, and Ronen Eldan. Kernel-based methods for bandit convex optimization. In *STOC*, pages 72–85, 2017.

[Cella and Cesa-Bianchi, 2020] Leonardo Cella and Nicolò Cesa-Bianchi. Stochastic bandits with delay-dependent payoffs. In *AISTATS*, pages 1168–1177. PMLR, 2020.

[Cesa-Bianchi *et al.*, 2018] Nicolo Cesa-Bianchi, Claudio Gentile, and Yishay Mansour. Nonstochastic bandits with composite anonymous feedback. In *COLT*, pages 750–773, 2018.

[Chapelle *et al.*, 2014] Olivier Chapelle, Eren Manavoglu, and Romer Rosales. Simple and scalable response prediction for display advertising. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 5(4):1–34, 2014.

[Dekel *et al.*, 2014] Ofer Dekel, Jian Ding, Tomer Koren, and Yuval Peres. Bandits with switching costs: $T^{2/3}$ regret. In *STOC*, pages 459–467, 2014.

[Flaxman *et al.*, 2005] Abraham D Flaxman, Adam Tauman Kalai, and H Brendan McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. In *SODA*, pages 385–394, 2005.

[Gergely Neu *et al.*, 2010] András György Gergely Neu, Csaba Szepesvári, and András Antos. Online markov decision processes under bandit feedback. In *NeurIPS*, 2010.

[Gyorgy and Joulani, 2021] Andras Gyorgy and Pooria Joulani. Adapting to delays and data in adversarial multi-armed bandits. In *ICML*, pages 3988–3997, 2021.

[Heidari *et al.*, 2016] Hoda Heidari, Michael Kearns, and Aaron Roth. Tight policy regret bounds for improving and decaying bandits. In *IJCAI*, pages 1562–1570, 2016.

[Jaghargh *et al.*, 2019] Mohammad Reza Karimi Jaghargh, Andreas Krause, Silvio Lattanzi, and Sergei Vassilvtiskii. Consistent online optimization: Convex and submodular. In *AISTATS*, pages 2241–2250, 2019.

[Joulani *et al.*, 2013] Pooria Joulani, Andras Gyorgy, and Csaba Szepesvári. Online learning under delayed feedback. In *ICML*, pages 1453–1461, 2013.

[Kocsis and Szepesvári, 2006] Levente Kocsis and Csaba Szepesvári. Discounted ucb. In *2nd PASCAL Challenges Workshop*, volume 2, 2006.

[Lei *et al.*, 2017] Huitian Lei, Ambuj Tewari, and Susan A Murphy. An actor-critic contextual bandit algorithm for personalized mobile health interventions. *arXiv:1706.09090*, 2017.

[Li *et al.*, 2019] Bingcong Li, Tianyi Chen, and Georgios B Giannakis. Bandit online learning with unknown delays. In *AISTATS*, pages 993–1002, 2019.

[Pike-Burke *et al.*, 2018] Ciara Pike-Burke, Shipra Agrawal, Csaba Szepesvari, and Steffen Grunewalder. Bandits with delayed, aggregated anonymous feedback. In *ICML*, pages 4105–4113, 2018.

[Quanrud and Khashabi, 2015] Kent Quanrud and Daniel Khashabi. Online learning with adversarial delays. In *NeurIPS*, pages 1270–1278, 2015.

[Saha and Tewari, 2011] Ankan Saha and Ambuj Tewari. Improved regret guarantees for online smooth convex optimization with bandit feedback. In *AISTATS*, pages 636–642. JMLR Workshop and Conference Proceedings, 2011.

[Silver *et al.*, 2016] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

[Thune *et al.*, 2019] Tobias Sommer Thune, Nicolò Cesa-Bianchi, and Yevgeny Seldin. Nonstochastic multiarmed bandits with unrestricted delays. In *NeurIPS*, pages 6541–6550, 2019.

[Villar *et al.*, 2015] Sofía S Villar, Jack Bowden, and James Wason. Multi-armed bandit models for the optimal design of clinical trials: benefits and challenges. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 30(2):199, 2015.

[Wang *et al.*, 2021] Siwei Wang, Haoyun Wang, and Longbo Huang. Adaptive algorithms for multi-armed bandit with composite and anonymous feedback. In *AAAI*, volume 35, pages 10210–10217, 2021.

[Zimmert and Seldin, 2020] Julian Zimmert and Yevgeny Seldin. An optimal algorithm for adversarial bandits with arbitrary delays. In *AISTATS*, pages 3285–3294, 2020.