Memory-Constrained No-Regret Learning in Adversarial Multi-Armed Bandits

Xiao Xu and Qing Zhao, *Fellow, IEEE,*

***Abstract*—An adversarial multi-armed bandit problem with memory constraints is studied where the memory for storing arm statistics is only in a sublinear order of the number of arms. A hierarchical learning framework that offers a sequence of operating points on the tradeoff curve between the regret order and memory complexity is developed. Its sublinear regret orders are established under both weak regret and shifting regret notions. This work appears to be the first on memory-constrained bandit problems in the adversarial setting.**

***Index Terms*—Adversarial multi-armed bandits, no-regret learning, memory complexity.**

I. Introduction

FIRST posed in [1] for the application of clinical trials, the multi-armed bandit (MAB) problem has been studied under various models and across diverse application domains [2]. The name of the problem comes from likening an archetypical single-player online learning problem to playing a multi-armed slot machine (known as a bandit for its ability of emptying the player's pocket). Each arm, when pulled, generates rewards according to an unknown stochastic model or in an adversarial fashion. Only the reward of the chosen arm is revealed after each play. The objective of the player is an arm selection policy that maximizes the cumulative reward over T plays. The bandit feedback model where an arm can only be observed after it is played induces the tradeoff between exploration (to gather information from less explored arms) and exploitation (to maximize immediate reward by prioritizing arms with a good reward history).

Depending on the generative model of arm rewards, bandit problems can be categorized into the stochastic and the adversarial settings. In the former, rewards from successive plays of an arm obey a given, albeit unknown, stochastic model. In the latter, rewards are assigned by an adversary. Regardless of the reward models, a commonly adopted performance measure of an arm selection policy is regret, defined as the cumulative reward loss against a properly defined benchmark policy that assumes hindsight vision or certain clairvoyant knowledge about the underlying generative model of arm rewards. The difference between the regret measures in the stochastic and the adversarial settings is in the adopted benchmark policies.

A canonical model for stochastic bandits assumes that rewards from each arm are drawn i.i.d. from a fixed distribution. In this case, the benchmark policy in the regret definition is one that assumes the knowledge of the stochastic model, hence plays the arm with the greatest mean throughout the time horizon. The regret is measured in expectation taken over the random process of reward realizations induced by the arm selection policy. Representative studies include [3]–[6].

The adversarial bandit problem, first studied in [7], is closely related to the problem of learning in repeated unknown games. In the game setting, a player's reward of taking a particular action (i.e., playing a particular arm) is jointly determined by the payoff function of the game and the actions taken by all opponents. From the perspective of a single player, the reward can be viewed as assigned by an adversary aggregating the interactions with all opponents in the game [8]. Connections between certain system-level objectives of the game (e.g., convergence to equilibria) and the regret performance of a single player against a collective adversary have been revealed [9]–[11]. See a recent survey on distributed leaning in multi-agent systems [12].

Various benchmark policies have been considered for regret measures in the adversarial setting. In particular, weak regret is defined against a benchmark policy that plays the best (fixed) arm in terms of the cumulative reward in hindsight [13]. The weak regret notion corresponds to the external regret in the game setting. A stronger regret notion is the shifting regret, where the benchmark policy is allowed to switch arms over time but limited by a hardness constraint on the number of switchings.

A policy is said to achieve no-regret learning if, for every sequence of rewards assigned by the adversary, the adopted regret measure has a sublinear growth rate with T. In other words, the policy offers, asymptotically as $T \to \infty$, the same average reward as the specific benchmark adopted in the corresponding regret measure. A number of learning algorithms have been developed to achieve no-regret learning under various regret notions [13], [14]. It has been shown that randomization in arm selection is necessary for achieving no-regret learning [15].

A. Main Results

Memory complexity has not been considered in adversarial bandits. Existing learning policies require a memory space with size linear in the number K of arms to store arm reward statistics. Such a linear order of memory complexity may render these learning policies impractical in applications

Xiao Xu and Qing Zhao are with the School of Electrical and Computer Engineering, Cornell University, Ithaca, NY, 14850 USA (e-mails: {xx243, qz16}@cornell.edu).

This work was supported by the National Science Foundation under Grant CCF-1815559. An earlier version was posted on arXiv at https://arxiv.org/abs/2002.11804.

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org., provided by the author. The material includes appendices of proof details. This material is 300KB in size.

involving a large action/arm space, for example, recommendation systems and dynamic routing in urban transportation and computer networks.

In this paper, we study the memory-constrained adversarial bandit problem where a learning policy is only given M words of memory for storing input values and necessary variables, where M is in a sublinear order of K. The memory constraint entails that past reward observations, except for a diminishing fraction, need to be either forgotten or summarized with certain succinct statistics. No-regret learning hence hinges on not only a balance between exploration and exploitation, but also a balance between what to remember and what to forget.

In this work, we develop a hierarchical learning framework that offers a sequence of operating points on the tradeoff curve between the regret order and memory complexity. Referred to as *HLMC (Hierarchical Learning with Memory Constraints)*, the proposed learning framework partitions the arms into multi-level groups and the time horizon into multi-level epochs through a tree-structured hierarchy. The depth of the tree is chosen to trade off regret order with memory complexity: a deeper tree leads to a lower memory complexity at the price of a higher regret order. Using aggregated statistics for arm groups at all levels, the HLMC framework recursively selects arm groups (referred to as super arms) across epochs (referred to as super time steps) according to the tree hierarchy. Within each epoch, a memory-unconstrained learning policy can be employed to govern the selection of arm groups at the corresponding level. This hierarchical learning framework decouples the design issue of to-remember-or-to-forget induced by the memory constraint from the exploration-exploitation tradeoff induced by the bandit feedback. It hence provides a general framework for extending memory-unconstrained learning policies to memory-constrained settings.

We establish the regret performance and memory complexity of HLMC as a function of D, the depth of the adopted tree hierarchy. In particular, for $D = 2$, HLMC consists of a leaf level of K individual arms and a higher level of $\Theta(\sqrt{K})$ arm groups, each consisting of $\Theta(\sqrt{K})$ arms. In this case, the memory required by HLMC consists of two parts: one for storing group statistics used by the group-level selection strategy, the other for arm statistics within the selected group for arm selection. We show that the memory complexity of HLMC is $\Theta(\sqrt{K})$. In terms of regret performance, we show that no-regret learning is achieved by HLMC under both weak regret and shifting regret when suitable memory-unconstrained policies are employed as learning routines at each level. Specifically, with a sublinear-order memory complexity of $\Theta(\sqrt{K})$, HLMC offers a weak regret of $O(T^{3/4}K^{1/4})$ and a shifting regret of $O(T^{3/4}V^{1/4}K^{1/4})$ up to logarithmic factors, where V is the hardness constraint on the benchmark policy.

In the general case with a D-level hierarchy ($D \geq 2$), the memory required by HLMC consists of D parts for storing group statistics at all D levels. We show that the memory complexity is of order $\Theta(DK^{1/D})$ with a weak regret order of $O(DT^{1-\frac{1}{2D}}K^{\frac{1}{2D}})$ up to a logarithmic factor. The tradeoff between regret order and memory complexity of HLMC is therefore quantified through the discrete depth D of the adopted hierarchy, which can be designed in accordance with the size of the available memory space. At the two ends of the spectrum is $D = \lceil \log_2 K \rceil$ and $D = 1$. In the former, HLMC achieves no-regret learning under the notion of weak regret with a memory complexity that is only logarithmic in K. In the latter, the problem degenerates to the memory-unconstrained setting, and HLMC reduces to a memory-unconstrained learning routine.

B. Related Work

There is a growing body of work on adversarial bandits, in both the canonical form [7], [13], [14] and various variants arising in specific applications (see, for example, [16], [17]). However, memory constraints have not been considered. Most related to this work are two recent studies on memory-constrained *stochastic* bandit models [18], [19]. In the stochastic setting in [18], [19], rewards from each arm are drawn i.i.d. from a *fixed* distribution. Based on the sample sizes and the gaps in the sample mean, suboptimal arms can be identified up to a desired level of accuracy and subsequently eliminated from memory. Indeed, the key idea of the two algorithms proposed in [18] [19] is based on best arm identification techniques (see [20] for examples). Specifically, the memory constraint is dealt with by exploring and comparing a subset of arms over a period of time and successively eliminating suboptimal arms.

The above learning policies for memory-constrained stochastic bandits, however, do not apply to the adversarial setting. Being deterministic, they incur linear regret orders against adversaries. This is also confirmed in our numerical studies in Sec. VI. The fundamental difference between a memory-constrained adversarial bandit problem and its stochastic counterpart is that the best arm in hindsight of an adversarially chosen reward sequence can not be reliably inferred from partial observations. As a result, no arms can be reliably eliminated from consideration at any point in the learning horizon without causing significant regret. In the proposed HLMC, the memory constraint is dealt with by storing succinct aggregated arm statistics rather than completely forgetting certain set of arms.

Another type of memory constraint that has been studied in the MAB literature is temporal across time steps: a policy can only make decisions based on the reward outcomes of the m most recent plays. This problem was first considered in [21] where a two-armed bandit problem with Bernoulli rewards was studied. It was later shown in [22] that there exists a policy with $m = 2$ that achieves an asymptotically optimal average reward in the two-armed bandit instance. The decision process with temporal memory constraints was further modeled as a finite-state machine in [23], where the past reward history was aggregated as a finite-valued statistic. The objective considered in these studies was the asymptotic convergence of the empirical average reward. Analysis on the convergence rate or the regret order, however, was lacking. The objective of minimizing regret with temporal memory constraints was considered in [24] under the full-information feedback setting (i.e., the rewards of all arms that the player could have played are revealed after every time step). A learning algorithm achieving no-regret learning with $O(m^K)$ states (each arm statistic can take $O(m)$ values) was developed. However, the full-information feedback setting is fundamentally different from the bandit setting studied in this paper. Moreover, the proposed learning algorithm needs to store a statistic of every arm and the total number of states is exponential in K.

II. Problem Formulation

We consider an adversarial bandit problem with a finite arm set $\mathcal{A} = \{1, 2, ..., K\}$. At each time $t = 1, 2, ..., T$, a player chooses one arm to play. The reward $r_{i,t} \in [0, 1]$ of playing an arm i at time t is assigned by an adversary. We assume that the adversary is *oblivious*, i.e., the assignment of the reward at time t is independent of the player's past actions. Equivalently, an oblivious adversary determines the sequence of reward vectors $((r_{1,t}, ..., r_{K,t}))_{t=1}^T$ ahead of time. We assume that the player can only observe the reward of the selected arm at each time.

The objective of the player is an online learning policy π that specifies a sequential arm selection rule at each time t based on the observation history. We assume that the policy can only use M ($M = o(K)$ as $K \to \infty$) words of memory space to store input values and necessary parameters. We follow the memory model studied in [19] where each of the variables used by the policy takes 1 word of memory[1] and thus, a policy with memory size M can only store M statistics at any given time to summarize the reward history of arms.

The performance of policy π is measured by regret, which is defined as the reward loss against the best benchmark action sequence $a^T = (a_1, ..., a_T)$ with the greatest cumulative reward, i.e.,

$$R_\pi(T) = \max_{a^T \in \mathcal{A}^T} \sum_{t=1}^{T} r_{a_t,t} - \sum_{t=1}^{T} r_{\pi_t,t}, \tag{1}$$

where $\mathcal{A}^T$ is the set of all possible action sequences with length T and π_t is the arm selected by policy π at time t. When there is no ambiguity, the notation is simplified to $R(T)$.

As the regret $R(T)$ can be randomized due to the potential randomness of the arm selection policy π, we consider two types of *no-regret learning* conditions in this paper. A policy π is said to achieve no-regret learning *in expectation* if, for every sequence of rewards $((r_{1,t}, ..., r_{K,t}))_{t=1}^T$, the expected regret $E_\pi[R(T)] = o(T)$ as $T \to \infty$, where the expectation is taken over the possible randomness of π. The second condition states that a policy π achieves no-regret learning *with high probability* if, for every sequence of rewards and every given $\delta \in (0, 1)$, the regret $R(T) = o(T)$ as $T \to \infty$ with probability at least $1 - \delta$.

It is not difficult to see that achieving no-regret learning, either in expectation or with high probability, is impossible if the benchmark sequence is chosen arbitrarily [13]. Therefore, certain restrictions on the benchmark sequence is necessary to make the problem feasible. In this paper, we consider two types of regret notions with different restrictions on the benchmark sequence. The first regret notion is the so-called *weak regret* where the benchmark sequence consists of a single arm, i.e.,

$$R_{\text{w}}(T) = \max_{i \in \mathcal{A}} \sum_{t=1}^{T} r_{i,t} - \sum_{t=1}^{T} r_{\pi_t,t}. \tag{2}$$

A stronger regret notion is the so-called *shifting regret* where the benchmark sequence is constrained by its *hardness*. Specifically, the hardness of a sequence $a^T = (a_1, ..., a_T)$ measures the total number of arm switchings over time, i.e.,

$$H(a^T) \triangleq 1 + \sum_{t=1}^{T-1} \mathbb{I}(a_t \neq a_{t+1}), \tag{3}$$

where $\mathbb{I}(\cdot)$ is the indicator function. The shifting regret with a hardness constraint V is defined as

$$R_{\text{s}}(T, V) = \max_{a^T : H(a^T) \leq V} \sum_{t=1}^{T} r_{a_t,t} - \sum_{t=1}^{T} r_{\pi_t,t}. \tag{4}$$

It is clear that the shifting regret is a stronger notion than the weak regret: no-regret learning under the former implies no-regret learning under the latter, but not vice versa.

To achieve no-regret learning under various regret notions, a number of learning routines have been developed in the memory-unconstrained setting. Representative algorithms include EXP3, EXP3.P, and EXP3.S that achieve no-regret learning under the notion of weak regret in expectation, with high probability, and under the notion of shifting regret in expectation, respectively. We summarizes the details of these algorithms in Appendix A.

III. Hierarchical Learning with Memory Constraints

In this section, we propose a general learning structure: *HLMC (Hierarchical Learning with Memory Constraints)* for the memory-constrained adversarial bandit problem. We first present the general framework of HLMC with a multi-level hierarchy on the partitions of the arms and the time horizon. Then we use a representative case with a two-level hierarchy to illustrate its details.

A. A General Framework with Multi-Level Hierarchy

The key to the balance between what to remember and what to forget induced by memory constraints is to summarize past reward observations through certain succinct statistics. This motivates partitions of the arms into tree-structured groups and the time horizon into tree-structure epochs through a D-level hierarchy. At every level of the hierarchy, reward observations from arms within a group during an epoch is aggregated as a single group statistic. Using these group statistics, the HLMC structure carries out a recursive learning procedure that successively selects and zooms into an arm group during every corresponding epoch according to the tree hierarchy. See Fig. 1 for an example of HLMC with a three-level hierarchy.

Through the design of the depth D of the adopted hierarchy, HLMC achieves different operating points on the tradeoff curve between the regret order and memory complexity. Intuitively, a deeper hierarchy requires a smaller memory space

[1] The number of bits in a word depends on how real numbers are stored in the memory, which is out of the scope of this paper.

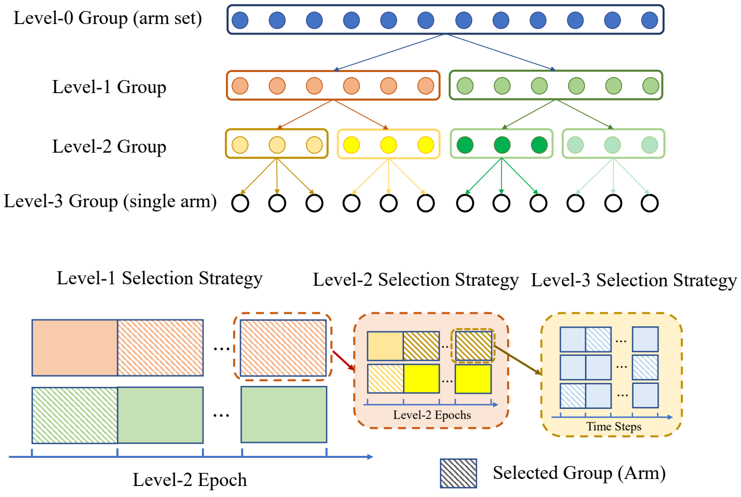

Fig. 1: HLMC with a three-level hierarchy: the arm set is partitioned into two level-1 groups and every level-1 group is partitioned into two level-2 groups (see Sec. V for detailed discussions on the number of groups at each level). Every level-2 group consists of 3 arms (level-3 groups). The time horizon is partitioned in a similar way into multi-level epochs. At the beginning of every level-ℓ epoch ($1 \le \ell \le 3$), a level-ℓ arm group is selected according to a level-ℓ strategy. Within this epoch, a level-$(\ell+1)$ strategy is conducted to select level-$(\ell+1)$ arm groups with in the selected level-ℓ group across level-$(\ell+1)$ epochs.

for storing reward statistics, but incurs a higher regret order. See Sec. V for detailed discussions.

It should be noted that HLMC is a general learning framework that decouples the tradeoff between what to remember and what to forget from the one between exploration and exploitation. The solution to the former is the design of the aggregated group statistics and the recursive learning structure. For the latter, different learning routines developed in the memory-unconstrained setting can be plugged in for group selection at each level with the goal of minimizing various notions of regret.

B. A Representative Case with Two-Level Hierarchy

We use $D = 2$ as a representative case to present the details of HLMC. In the two-level hierarchy, the set $\mathcal{A}$ of arms is partitioned into equal-sized groups $\{\mathcal{A}_\ell\}_{\ell=1}^{L}$ where

$$\mathcal{A}_\ell = \{1 + N(\ell - 1), ..., \min(N\ell, K)\}, \tag{5}$$

$N = \lceil \sqrt{K} \rceil$ is the group size (note that the number of arms in the last group may be smaller than N), and $L = \lceil \frac{K}{N} \rceil$ is the number of groups. The time horizon is partitioned into equal-length epochs $\{\mathcal{T}_s\}_{s=1}^{S}$ where

$$\mathcal{T}_s = [1 + \Delta(s - 1), \min(\Delta S, T)], \tag{6}$$

$\Delta \in \mathbb{N}^+$ is the epoch length to be determined later, and $S = \lceil \frac{T}{\Delta} \rceil$ is the number of epochs. Note that the length of the S-th epoch may be smaller than Δ.

By treating each group $\mathcal{A}_\ell$ as a "super arm" ℓ and each epoch $\mathcal{T}_s$ a "super time-step" s, we reduce the group selection problem to a classic memory-unconstrained adversarial bandit problem. Specifically, with $M \ge L$, existing learning strategies developed for memory-unconstrained adversarial bandits can be adopted using L words of memory space to select groups across epochs, without violating the memory constraint. The reward of playing a "super arm" ℓ_s at a "super time-step" s is defined as the average reward per play obtained from the corresponding arm group $\mathcal{A}_{\ell_s}$ during the corresponding epoch $\mathcal{T}_s$, i.e.,

$$y_{\ell_s,s} = \frac{1}{|\mathcal{T}_s|} \sum_{t \in \mathcal{T}_s} r_{i_t,t}, \tag{7}$$

where $i_t \in \mathcal{A}_{\ell_s}$ is the arm selected at time t.

The group-level strategy uses an aggregated statistic of every arm group, which is stored throughout the time horizon, for group selection across epochs. Once a group $\mathcal{A}_{\ell_s}$ is selected at the beginning of every epoch $\mathcal{T}_s$, an arm-level learning routine is employed on $\mathcal{A}_{\ell_s}$ based on individual statistics of arms within the group. These arm statistics are updated at every time step in $\mathcal{T}_s$ and are forgotten at the end of the epoch. After each epoch, the average reward $y_{\ell_s,s}$ per play is used to update the aggregated statistic of the selected group $\mathcal{A}_{\ell_s}$. The details of HLMC with a two-level hierarchy is summarized in Algorithm 1.

Algorithm 1 HLMC with a Two-Level Hierarchy

```
Input: $T$ the time length, $\mathcal{A}$ the set of $K$ arms, and $\Delta > 0$ the epoch length.
Obtain arm group partition $\{\mathcal{A}_\ell\}_{\ell=1}^{L}$ according to (5).
Obtain epoch partition $\{\mathcal{T}_s\}_{s=1}^{S}$ according to (6).
Initialize and store the statistics of every arm group.
for $s = 1, 2, ..., S$ do
    Select arm group $\ell_s$ according to the group-level selection strategy.
    Initialize and store the statistics of every arm in $\mathcal{A}_{\ell_s}$.
    Initialize $y_{\ell_s,s} = 0, \tau = 0$.
    for $t \in \mathcal{T}_s$ do
        Play arm $i_t$ according to the arm-level selection strategy and receive reward $r_{i_t,t}$.
        Update arm statistics in the memory using $r_{i_t,t}$.
        Update $y_{\ell_s,s} = \frac{y_{\ell_s,s}\tau + r_{i_t,t}}{\tau+1}$, $\tau = \tau + 1$.
    Update all group statistics in the memory using $y_{\ell_s,s}$.
```

IV. Memory Complexity and Regret Performance in the Two-Level Case

In this section, we analyze the memory complexity and regret performance of the proposed HLMC learning framework in the two-level case. We notice that in HLMC, the group-level strategy requires L words of memory to store a statistic of every arm group. Once a group is selected, the statistics of all arms within the selected group should also be stored. Hence, N additional words of memory are needed. As a result, the total memory size required by the HLMC framework is $N + L$, which is of order $\Theta(\sqrt{K})$.

In terms of regret performance, it is clear that the regret order in T achieved by HLMC depends on the specific learning routines employed at both group and arm levels. In the following three subsections, we discuss minimizing weak regret in expectation, with high probability, and minimizing shifting regret in expectation, respectively through plugging in different learning routines to the two levels.

A. Minimizing Weak Regret in Expectation

We first show that adopting EXP3 at both group and arm levels in the HLMC framework with learning rates γ_1 and γ_2 respectively guarantees a sublinear regret order in T under the notion of expected weak regret.

Theorem 1. *For any T and K, if the input parameter $\Delta = \left\lceil \sqrt{\frac{TN\ln N}{L\ln L}} \right\rceil$ (where N, L are defined in Sec. III-B), adopting EXP3 at both group and arm levels with learning rates $\gamma_1 = \sqrt{\frac{L\ln L}{2S}}$ and $\gamma_2 = \sqrt{\frac{N\ln N}{2\Delta}}$ guarantees that, for every assignment of the reward sequence, the expected weak regret of HLMC is upper bounded by:*

$$\mathbb{E}_{HLMC}\left[R_w(T)\right] \leq (4+2\sqrt{2})T^{\frac{3}{4}}K^{\frac{1}{4}}(\ln K)^{\frac{1}{2}}. \tag{8}$$

To obtain the upper bound in Theorem 1, we decompose the expected weak regret into two parts by introducing an intermediate term $C'_{\max}$ as follows: for every fixed reward sequence, let $i_{\max}$ be the best arm with the greatest cumulative reward over the entire time horizon and $\mathcal{A}_{\ell_{\max}}$ the arm group to which $i_{\max}$ belongs. We define $C'_{\max}$ as the expected cumulative reward obtained by running the arm-level EXP3 algorithm with learning rate γ_2 on $\mathcal{A}_{\ell_{\max}}$ during all epochs, i.e.,

$$C'_{\max} = \sum_{s=1}^{S} \mathbb{E}_{\text{Arm-EXP3}(\mathcal{A}_{\ell_{\max}})}\left[\sum_{t\in\mathcal{T}_s} r_{i_t,t}\right], \tag{9}$$

where $\mathbb{E}_{\text{Arm-EXP3}(\mathcal{A}_{\ell_{\max}})}[\cdot]$ denotes the expectation taken over the randomness of the arm-level EXP3 algorithm when conducted on group $\mathcal{A}_{\ell_{\max}}$. Then the expected weak regret of HLMC is decomposed as:

$$\mathbb{E}_{\text{HLMC}}\left[R_{\text{w}}(T)\right] = \underbrace{(C'_{\max} - C_{\text{HLMC}})}_{R_1(T)} + \underbrace{(C_{\max} - C'_{\max})}_{R_2(T)}, \tag{10}$$

where

$$\begin{aligned} C_{\text{HLMC}} &= \mathbb{E}_{\text{HLMC}}\left[\sum_{t=1}^{T} r_{i_t,t}\right], \\ C_{\max} &= \sum_{t=1}^{T} r_{i_{\max},t}. \end{aligned} \tag{11}$$

Note that in the decomposition, $R_1(T)$ corresponds to the group-level reward loss due to not selecting $\mathcal{A}_{\ell_{\max}}$ at every epoch, and $R_2(T)$ corresponds to the arm-level reward loss due to playing suboptimal arms in $\mathcal{A}_{\ell_{\max}}$ assuming that group $\mathcal{A}_{\ell_{\max}}$ is selected at all epochs.

We first upper bound the group-level reward loss $R_1(T)$. Noticing that the arm selection process during every epoch is independent of the group and arm selection history in the past, we can thus rewrite the expected reward of the HLMC policy as follows:

$$\begin{aligned} &\mathbb{E}_{\text{HLMC}}\left[\sum_{t=1}^{T} r_{i_t,t}\right] \\ =&\mathbb{E}_{\text{Group-EXP3}}\left[\sum_{s=1}^{S} \mathbb{E}_{\text{Arm-EXP3}(\mathcal{A}_{\ell_s})}\left[\sum_{t\in\mathcal{T}_s} r_{i_t,t}\right]\right], \end{aligned} \tag{12}$$

where $\mathbb{E}_{\text{Group-EXP3}}[\cdot]$ denotes the expectation taken over the randomness of the group-level EXP3 algorithm, and $\mathcal{A}_{\ell_s}$ is the group selected at epoch s. To ease the analysis, we assume without losing generality that all epochs have an equal length Δ. We further define

$$x_{\ell,s} = \mathbb{E}_{\text{Arm-EXP3}(\mathcal{A}_\ell)}\left[\frac{1}{|\mathcal{T}_s|}\sum_{t\in\mathcal{T}_s} r_{i_t,t}\right]. \tag{13}$$

It is not difficult to see that

$$R_1(T) = \Delta\left(\sum_{s=1}^{S} x_{\ell_{\max},s} - \mathbb{E}_{\text{Group-EXP3}}\left[\sum_{s=1}^{S} x_{\ell_s,s}\right]\right). \tag{14}$$

It is then clear that upper bounding $R_1(T)$ is equivalent to upper bounding the weak regret of applying the group-level EXP3 algorithm to the adversarial bandit problem constructed by the reduction in Sec. III. Specifically, the reward of selecting a group $\mathcal{A}_\ell$ at epoch $\mathcal{T}_s$ is defined as $y_{\ell,s}$ according to (7) where i_t is randomly selected by the arm-level EXP3 algorithm. Therefore, $y_{\ell,s}$ is a random reward with mean $x_{\ell,s}$. The group selection problem is reduced to a classic memory-unconstrained adversarial bandit problem with noisy observations. It should be noted that after fixing an assignment of the reward sequence $((r_{1,t},...,r_{K,t}))_{t=1}^{T}$, the expected reward $x_{\ell,s}$ is fixed. Meanwhile, the realization of $y_{\ell,s}$ is independent across ℓ, s and is independent of the arm (group) selection history up to epoch s. We obtain the following result on applying the group-level EXP3 algorithm to the reduced bandit problem.

Lemma 1. *By choosing $\gamma_1 = \sqrt{\frac{L\ln L}{2S}}$, the group-level EXP3 algorithm guarantees that, for every assignment of the reward sequence $((r_{1,t},...,r_{K,t}))_{t=1}^{T}$,*

$$\max_{1\leq\ell\leq L} \sum_{s=1}^{S} x_{\ell,s} - \mathbb{E}_{\textit{Group-EXP3}}\left[\sum_{s=1}^{S} x_{\ell_s,s}\right] \leq 2\sqrt{2SL\ln L}, \tag{15}$$

where ℓ_s is the arm group selected by the group-level EXP3 algorithm at epoch s.

Proof. See Appendix B in the supplementary material. □

For the arm-level reward loss $R_2(T)$, we notice that

$$R_2(T) = \sum_{s=1}^{S}\left(\sum_{t\in\mathcal{T}_s} r_{i_{\max},t} - \mathbb{E}_{\text{Arm-EXP3}(\mathcal{A}_{\ell_{\max}})}\left[\sum_{t\in\mathcal{T}_s} r_{i_t,t}\right]\right). \tag{16}$$

It suffices to upper bound each term in the summation, that is, the weak regret of conducting the arm-level EXP3 algorithm on group $\mathcal{A}_{\ell_{\max}}$ during each epoch $\mathcal{T}_s$. The regret bound has been shown in Lemma 3.

Theorem 1 is then proved by applying Lemma 1 and Lemma 3 to $R_1(T)$ and $R_2(T)$, respectively.

Proof of Theorem 1. Combining (14) with Lemma 1, and (16) with Lemma 3, we can derive that

$$\begin{aligned} R_1(T) &\leq 2\Delta\sqrt{2SL\ln L} = 2\sqrt{2T\Delta L\ln L}, \\ R_2(T) &\leq 2S\sqrt{2\Delta N\ln N} = 2\sqrt{\frac{2T^2}{\Delta}N\ln N}. \end{aligned} \tag{17}$$

By choosing $\Delta = \left\lceil\sqrt{\frac{TN\ln N}{L\ln L}}\right\rceil$, we obtain the upper bound in Theorem 1. □

It should be noted that although the proposed learning policy requires the knowledge of the total time length T for choosing input parameters to achieve no-regret learning, the issue of unknown T can be easily addressed by the doubling technique as used in the classic memory-unconstrained setting [13]. Specifically, the algorithm operates in stages, with the stage length doubles at each time. In stage r with length 2^r, the algorithm operates under a known-horizon setting with the horizon length $T = 2^r$. It is not difficult to show that the same regret order still holds.

B. Minimizing Weak Regret with High Probability

We further show that by adopting EXP3.P at both group and arm levels in the HLMC framework with parameters $(\eta_1, \gamma_1, \beta_1)$ and $(\eta_2, \gamma_2, \beta_2)$ respectively, the weak regret of HLCM has a sublinear growth rate in T with high probability.

Theorem 2. *For any T, K and every $\delta \in (0, 1)$, if $\Delta = \left\lceil\sqrt{\frac{TN\ln(2KT/\delta)}{L\ln(2L/\delta)}}\right\rceil$ (where N, L are defined in Sec. III-B), and the EXP3.P algorithm is adopted at both the group level with $\beta_1 = \sqrt{\frac{\ln(2L/\delta)}{LS}}, \eta_1 = 0.95\sqrt{\frac{\ln L}{LS}}, \gamma_1 = 1.05\sqrt{\frac{L\ln L}{S}}$, and the arm level with $\beta_2 = \sqrt{\frac{\ln(2KS/\delta)}{N\Delta}}, \eta_2 = 0.95\sqrt{\frac{\ln N}{N\Delta}}, \gamma_2 = 1.05\sqrt{\frac{N\ln N}{\Delta}}$, then for any assignment of the reward sequence, the weak regret of HLCM is upper bounded by*

$$R_w(T) \leq 12.5T^{\frac{3}{4}}K^{\frac{1}{4}}(\ln(2KT/\delta))^{\frac{1}{2}}, \tag{18}$$

with probability at least $1 - \delta$.

Theorem 2 is proved via a similar structure with that used in analyzing the expected weak regret of HLMC in Sec. IV-A. Specifically, the weak regret is decomposed as:

$$\begin{aligned} R_{\mathrm{w}}(T) &= \sum_{s=1}^{S}\sum_{t\in\mathcal{T}_s} r_{i_{\max},t} - \sum_{s=1}^{S}|\mathcal{T}_s|y_{\ell_{\max},s} \\ &\quad + \sum_{s=1}^{S}|\mathcal{T}_s|y_{\ell_{\max},s} - \sum_{s=1}^{S}\sum_{t\in\mathcal{T}_s} r_{i_t,t} \\ &= R_1(T) + R_2(T), \end{aligned} \tag{19}$$

where $i_{\max}$ is the arm with the greatest cumulative reward in hindsight, $\ell_{\max}$ is the group index of $i_{\max}$, $y_{\ell_{\max},s}$ is the average reward obtained by running the arm-level EXP3.P algorithm on $\mathcal{A}_{\ell_{\max}}$ during epoch s, and i_t is the arm selected by HLMC at time t.

We first upper bound $R_1(T)$, which corresponds to the arm-level reward loss due to playing suboptimal arms in $\mathcal{A}_{\ell_{\max}}$ assuming that $\mathcal{A}_{\ell_{\max}}$ is selected at all epochs. It suffices to upper bound

$$\sum_{t\in\mathcal{T}_s} r_{i_{\max},t} - |\mathcal{T}_s|y_{\ell_{\max},s}, \tag{20}$$

for every s. It is clear that (20) is equivalent to the weak regret of applying the arm-level EXP3.P algorithm to $\mathcal{A}_{\ell_{\max}}$ during epoch $\mathcal{T}_s$, which is upper bounded in Lemma 4.

To upper bound $R_2(T)$, which corresponds to the group-level reward loss due to not selecting $\mathcal{A}_{\ell_{\max}}$ at all epochs, we rewrite $R_2(T)$ as

$$R_2(T) = \Delta\left(\sum_{s=1}^{S} y_{\ell_{\max},s} - \sum_{s=1}^{S} y_{\ell_s,s}\right) \tag{21}$$

where ℓ_s is the group selected by the group-level EXP3.P algorithm at epoch s (we assume without loss of generality that every epoch has equal length Δ).

As argued in Sec. IV-A, the realization of $y_{\ell,s}$ is independent across ℓ, s and is independent of the past group selection history. Once we fixed a sequence of realizations of $((y_{1,s}, ...y_{L,s}))_{s=1}^{S}$, Lemma 4 can be applied to upper bound the group-level regret $R_2(T)$ with high probability.

Proof of Theorem 2. For every $\delta > 0$ and every assignment of the reward sequence, we apply Lemma 4 to all groups $\ell = 1, ..., L$ and all epochs $s = 1, ..., S$ by choosing $\delta_0 = \frac{\delta}{2LS}$. Then using the union bound, we obtain that with probability at least $1-\delta/2$, the upper bound on (20) in Lemma 4 holds for every groups ℓ and every epoch s. As a result, the arm-level regret $R_1(T)$ is upper bounded as:

$$\begin{aligned} R_1(T) &\leq 5.15S\sqrt{N\Delta\ln(2NLS/\delta)} \\ &= 5.15\sqrt{\frac{T^2}{\Delta}N\ln\left(\frac{2KS}{\delta}\right)}, \end{aligned} \tag{22}$$

with probability at least $1 - \delta/2$.

Moreover, we apply Lemma 4 again to the group-level selection strategy by choosing $\delta_0 = \delta/2$. We obtain that with probability at least $1 - \delta/2$,

$$\begin{aligned} R_2(T) &\leq 5.15\Delta\sqrt{LS\ln(2L/\delta)} \\ &= 5.15\sqrt{T\Delta L\ln(2L/\delta)}, \end{aligned} \tag{23}$$

for every realization of $((y_{1,s}, ...y_{L,s}))_{s=1}^{S}$. The upper bound on $R_{\mathrm{w}}(T)$ in Theorem 2 is obtained by choosing $\Delta = \left\lceil\sqrt{\frac{TN\ln(2KT/\delta)}{L\ln(2L/\delta)}}\right\rceil$ and combining (22) and (23) using the union bound. □

C. Minimizing Shifting Regret in Expectation

To achieve no-regret learning under a stronger regret notion: shifting regret, we consider applying EXP3.S at the group level of HLMC. At the arm-level, we still adopt the EXP3 algorithm for arm selection. It should be noted that the arm-level strategy in the HLMC framework is restarted at the beginning of every epoch, which guarantees quick elimination of the past experience. Therefore, the hierarchical structure automatically adapts to the variation of the benchmark sequence by relying more on recent observations. In the following theorem, we provide an upper bound on the expected shifting regret of HLMC when EXP3.S and EXP3 are adopted at the group and the arm levels, respectively.

Theorem 3. *For any T, K, and V, assume that $T \geq VK$. If the input parameter $\Delta = \left\lceil \sqrt{\frac{TN\ln N}{VL\ln(TL)}} \right\rceil$ (where N, L are defined in Sec. III-B), adopting EXP3.S at the group level with $\gamma_1 = \sqrt{\frac{VL\ln(LS)}{S}}$, $\alpha = 1/S$, and EXP3 at the arm level with $\gamma_2 = \sqrt{\frac{N\ln N}{2\Delta}}$ guarantees that, for every assignment of the reward sequence, the expected shifting regret of HLMC with a hardness constraint V on the benchmark action sequence is upper bounded by:*

$$\mathbb{E}_{\mathit{HLMC}}[R_s(T,V)] \leq (6\sqrt{2}+1)T^{\frac{3}{4}}V^{\frac{1}{4}}K^{\frac{1}{4}}(\ln(KT))^{\frac{1}{2}}. \quad (24)$$

Corollary 1. *If $V = o(T)$ as $T \to \infty$, the HLMC algorithm achieves no-regret learning in expectation under the notion of shifting regret with hardness constraint V.*

To upper bound the expected shifting regret of HLMC against an arbitrary benchmark action sequence a^T with a hardness constraint V, the key technique is to construct an alternative benchmark sequence b^T such that: (i) $H(b^T) \leq V$, (ii) the cumulative reward achieved by b^T is close to that achieved by a^T, and (iii) the actions specified by b^T are invariant within each epoch. Using such a sequence b^T, it suffices to show that the expected shifting regret of HLMC against b^T has a sublinear growth rate in T.

We follow the same proof structure with that used for analyzing the expected weak regret in Sec. IV-A. First note that the constructed sequence b^T is time-invariant within each epoch. Therefore, the arm-level regret analysis in Lemma 3 directly carries over. At the group-level, the reduction to a memory-unconstrained adversarial bandit problem with noisy observations is still legitimate since the group specified by the benchmark sequence is fixed within each epoch. Based on the reduction and Lemma 5, we obtain the following result on applying the EXP3.S algorithm to the group level.

Lemma 2. *By choosing $\gamma_1 = \sqrt{\frac{LV\ln(LS)}{S}}$ and $\alpha = 1/S$, the group-level EXP3.S algorithm guarantees that, for every assignment of the reward sequence $((r_{1,t},...,r_{K,t}))_{t=1}^T$ and every benchmark sequence of arm groups $h^S = (h_1,...,h_S)$ where $H(h^S) \leq V$,*

$$\sum_{s=1}^{S} x_{h_s,s} - \mathbb{E}_{\mathit{Group\text{-}EXP3.S}}\left[\sum_{s=1}^{S} x_{\ell_s,s}\right] \leq 4\sqrt{VLS\ln(LS)}, \quad (25)$$

where ℓ_s is the arm group selected at epoch s.

Proof. See Appendix C in the supplementary material. □

The upper bound in Theorem 3 on the expected shifting regret of HLMC against any arbitrary benchmark action sequence with a hardness upper bound V is obtained by combining Lemma 3 and Lemma 2 together.

Proof of Theorem 3. For an arbitrary benchmark action sequence a^T such that $H(a^T) \leq V$, we first construct an alternative benchmark sequence b^T as follows: suppose the time horizon is partitioned into V segments:

$$[T_1, T_2), [T_2, T_3), ..., [T_V, T_{V+1}), \quad (26)$$

where $T_1 = 1, T_{V+1} = T+1$, and a_t is fixed for all $t \in [T_v, T_{v+1})$ (let j_v denote that arm and h_v denote the group it belongs to). Suppose T_v belongs to epoch s_v. The alternative benchmark sequence b^T is defined as

$$b_t = j_v, \text{ if } s(t) \in [s_v, s_{v+1}), \quad (27)$$

where $s(t)$ is the epoch to which time t belongs.

One can check that the action specified by b^T is fixed within each epoch and $H(b^T) \leq V$. Moreover, b^T differs from a^T only in the epochs when an action switch happens in a^T, i.e., $\{s_v\}_{v=1}^V$. Therefore,

$$\sum_{t=1}^{T}(r_{a_t,t} - r_{b_t,t}) \leq V\Delta. \quad (28)$$

We decompose the expected shifting regret against a^T as:

$$\begin{aligned}
&\mathbb{E}_{\text{HLMC}}[R_{a^T}(T)] \\
=&\sum_{t=1}^{T}(r_{a_t,t} - r_{b_t,t}) + \left(\sum_{t=1}^{T} r_{b_t,t} - \sum_{v=1}^{V}\sum_{s=s_v}^{s_{v+1}-1}|\mathcal{T}_s|x_{h_v,s}\right) \\
&+\left(\sum_{v=1}^{V}\sum_{s=s_v}^{s_{v+1}-1}|\mathcal{T}_s|x_{h_v,s} - \mathbb{E}_{\text{HLMC}}\left[\sum_{t=1}^{T} r_{i_t,t}\right]\right) \\
=&R_1(T) + R_2(T) + R_3(T).
\end{aligned} \quad (29)$$

Note that $R_1(T) \leq V\Delta$. For $R_2(T)$, we have

$$\begin{aligned}
R_2(T) =& \sum_{v=1}^{V}\sum_{s=s_v}^{s_{v+1}-1}\sum_{t\in\mathcal{T}_s} r_{b_t,t} - \sum_{v=1}^{V}\sum_{s=s_v}^{s_{v+1}-1}|\mathcal{T}_s|x_{h_v,s} \\
\leq& 2S\sqrt{2\Delta N\ln N},
\end{aligned} \quad (30)$$

where the last inequality uses Lemma 3.

For $R_3(T)$, we can show that

$$\begin{aligned}
R_3(T) =& \sum_{v=1}^{V}\sum_{s=s_v}^{s_{v+1}-1}|\mathcal{T}_s|x_{h_v,s} - \mathbb{E}_{\text{Group-EXP3.S}}\left[\sum_{s=1}^{S}\Delta x_{\ell_s,s}\right] \\
\leq& 4\Delta\sqrt{VLS\ln(LS)},
\end{aligned} \quad (31)$$

where the last inequality uses Lemma 2.

Combining the above inequalities together and choosing $\Delta = \left\lceil \sqrt{\frac{TN\ln N}{VL\ln(TL)}} \right\rceil$, we can derive that

$$\mathbb{E}_{\text{HLMC}}[R_{a^T}(T)] \leq 6\sqrt{2}T^{\frac{3}{4}}V^{\frac{1}{4}}K^{\frac{1}{4}}(\ln(KT))^{\frac{1}{2}} + \sqrt{TVK\ln K}. \quad (32)$$

Notice that if $T \geq VK$, the first term on the RHS of (32) dominates. Since a^T is chosen arbitrarily with a hardness upper bounded V, we obtain the conclusion in Theorem 2. □

It should be noted that to achieve the upper bound established in Theorem 3, the knowledge of V is required in selecting input parameters. When V is unknown, we show in the following theorem that no-regret learning under shifting regret can still be achieved by HLMC in expectation under certain conditions.

Theorem 4. *By selecting $\Delta = \left\lceil \sqrt{\frac{TN\ln N}{L\ln(TL)}} \right\rceil$ and $\gamma_1 = \sqrt{\frac{L\ln(LS)}{S}}$ (the other parameters are identical to those specified in Theorem 3), the expected shifting regret of HLMC with a hardness constraint V on the benchmark action sequence is upper bounded by:*

$$\mathbb{E}_{HLMC}[R_s(T,V)] \leq \sqrt{2}(V+5)T^{\frac{3}{4}}K^{\frac{1}{4}}(\ln(KT))^{\frac{1}{2}}. \quad (33)$$

If $V = o(T^{1/4})$ as $T \to \infty$, no-regret learning is achieved by HLMC in expectation under shifting regret with a hardness constraint V, even if V is unknown.

Proof. The proof is similar to that of Theorem 3 and thus, we omit the details. □

V. Memory Complexity and Regret Performance in General Cases

As discussed in Sec. III, HLMC achieves different operating points on the tradeoff curve between the regret order and memory complexity through selecting different depth D of the adopted hierarchy. To show this, we provide performance analysis of HLMC in the general case with $D \geq 2$. For simplicity, we present detailed analysis for the case with $D = 3$. All claims and results can be easily generalized to cases with more than three levels.

We first introduce some notations and specify some parameters used in the algorithm as well as the analysis. The three levels in the hierarchy are referred to as the group, subgroup, and arm levels, respectively. In the first level, the arm set $\mathcal{A}$ is evenly partitioned into $N_1 = \lceil K^{1/3} \rceil$ groups $\{\mathcal{A}_\ell\}_{\ell=1}^{N_1}$. Within each group $\mathcal{A}_\ell$, arms are further evenly partitioned into $N_2 = \lceil K^{1/3} \rceil$ subgroups $\{\mathcal{B}_h^\ell\}_{h=1}^{N_2}$ in the second level. In the last level, each subgroup $\mathcal{B}_h^\ell$ consists of $N_3 = \lceil \frac{K}{N_1 N_2} \rceil$ arms (the size of the last subgroup within each group may be smaller than N_3). We assume without losing generality that the size of every group (subgroup) is identical. Similarly, the time horizon $\mathcal{T}$ is evenly partitioned into S_1 epochs $\{\mathcal{T}_s\}_{s=1}^{S_1}$ and every epoch $\mathcal{T}_s$ is evenly partitioned into S_2 subepochs $\{\mathcal{I}_\tau^s\}_{\tau=1}^{S_1}$. We assume that every sub-epoch consists of S_3 time steps (S_1, S_2, S_3 will be specified later). It is clear that $T = S_1 S_2 S_3$.

The HLMC framework consists of three selection strategies at the group, subgroup, and arm levels. At the beginning of every epoch $\mathcal{T}_s$, the group-level strategy selects a group $\mathcal{A}_{\ell_s}$. The statistics of all sub-groups within $\mathcal{A}_\ell$ are stored in the memory until the end of $\mathcal{T}_s$. During $\mathcal{T}_s$, the subgroup-level strategy selects a subgroup $\mathcal{B}_{h_\tau}^{\ell_s}$ at the beginning of every subepoch $\mathcal{I}_\tau^s$ and the statistics of arms within $\mathcal{B}_{h_\tau}^{\ell_s}$ are stored in the memory until the end of $\mathcal{I}_\tau^s$. The arm-level strategy is conducted on the selected subgroup to play arms at every time step during the corresponding subepoch.

It is clear that the size of the memory space required by HLMC with a three-level hierarchy is $N_1 + N_2 + N_3$. Therefore, the memory complexity of HLMC is in the order of $\Theta(K^{1/3})$. More generally, if we adopt a D-level hierarchy where each level d ($d = 1, 2, ..., D$) consists of $N_d = \lceil K^{1/D} \rceil$ level-d groups, the memory complexity of HLMC is of order $\Theta(DK^{1/D})$. It should be noted that a level-d group should contain at least 2 level-(d+1) groups. As a result, the depth D is upper bounded by $\lceil \log_2 K \rceil$ and the minimum memory complexity of the HLMC framework is of order $\Theta(\log_2 K)$.

We show that HLMC with a three-level hierarchy achieves no-regret learning in expectation under the notion of weak regret, if we adopt EXP3 at all three levels. Using a similar approach with that in analyzing the regret performance in the two-level case, we prove an upper bound on the expected weak regret of HLMC in the following theorem.

Theorem 5. *For any T and K, by choosing $S_i = \left\lceil \frac{T^{1/3}(N_i \ln N_i)^{2/3}}{(\prod_{j\neq i} N_j \ln N_j)^{1/3}} \right\rceil$ and applying EXP3 with parameter $\gamma_i = \sqrt{\frac{N_i \ln N_i}{2S_i}}$ at every level $i = 1, 2, 3$, the expected weak regret of HLMC with a three-level hierarchy against every assignment of the reward sequence is upper bounded by*

$$\mathbb{E}_{HLMC}[R_w(T)] \leq 12T^{5/6}K^{1/6}(\ln K)^{1/2}. \quad (34)$$

Proof. See Appendix D in the supplementary material. □

For general HLMC with a D-level hierarchy ($2 \leq D \leq \lceil \log_2 K \rceil$), the following corollary on the expected weak regret can be directly derived.

Corollary 2. *If EXP3 is applied to all D levels of the general HLMC framework, the expected weak regret is of order*

$$O(DT^{1-\frac{1}{2D}}K^{\frac{1}{2D}}) \quad (35)$$

up to a logarithmic factor, as $T \to \infty$.

Proof. The proof is similar to the ones of Theorem 1 and 5 and thus, we omit the details. □

Corollary 2 indicates that the tradeoff between the regret order and memory complexity of HLMC depends on the depth D of the adopted hierarchy: a deeper hierarchy incurs a higher regret order with a smaller memory complexity. We further establish a memory-dependent regret upper bound of HLMC by adaptively selecting D based on the size M of the available memory space. In particular, we define the minimum depth $D^*(M)$ of a legitimate hierarchy when M words of memory are available:

$$D^*(M) = \min\{D \in \mathbb{N}^+ : D\lceil K^{1/D} \rceil \leq M\}. \quad (36)$$

Thus, the minimum regret achieved by HLMC with M words of memory is of order

$$O\left(D^*(M)T^{1-\frac{1}{2D^*(M)}}K^{\frac{1}{2D^*(M)}}\right). \quad (37)$$

In one extreme case when $M = \Theta(\log_2 K)$, the size of the available memory space matches the minimum complexity of the deepest hierarchy where $D^*(M) = \lceil \log_2 K \rceil$. In this case, the regret order achieved by HLMC is still sublinear in T. In the other extreme case when $M \geq K$ (i.e., the memory-unconstrained case), it is clear that $D^*(M) = 1$ and HLMC with a single-level hierarchy reduces to an existing learning routine for memory-unconstrained adversarial bandits.

One may notice that the memory-dependent regret order of HLMC does not improve when M increases but $D^*(M)$ is unchanged, since the dependency of the regret order with respect to the available memory is quantified. However, in practice, a larger memory space may help in achieving a smaller regret if arms are adaptively partitioned according to M, even if D is fixed. We take the two-level case as an example: given M words of memory, we let $N = \lceil \frac{M-\sqrt{M^2-4K}}{2} \rceil$ and $L = \lceil \frac{K}{N} \rceil$. As long as $M \geq 2\sqrt{K}$, the arm partition is legitimate and one can verify that $N + L \leq M$. It is not difficult to check that the theoretical regret orders established in Sec. IV still hold under the adaptive arm partition. We further show in Sec. VI-C through numerical examples that under certain conditions, the regret performance of HLMC using adaptive arm partitions in the two-level hierarchy improves as M increases.

VI. Numerical Examples

In this section, we illustrate the regret performance of the proposed HLMC learning structure numerically through simulations. All the experiments are run 10 times using a Monte Carlo method on Python 3.7.

A. Weak Regret Minimization

We conduct two experiments to compare the regret performance of HLMC with baseline ones under the notion of weak regret. Given that this is the first work on memory-constrained adversarial bandits, we consider two baselines: UCB-M (proposed in [19] for memory constrained stochastic bandits) and EXP3 (for classic adversarial bandits without memory constraints).

We first notice that the only randomness of UCB-M comes from the random shuffle of arm indices before playing arms, which provides no improvement on the performance in the stochastic setting. Without the random shuffle step, UCB-M is purely deterministic and thus, we can easily construct a reward sequence such that UCB-M incurs a regret linear in T. Specifically, in the first experiment, we consider the following setup: let $K = 100$, $M = 20$, and $T = 10^7$. In accordance with the UCB-M policy, we partition the time horizon into phases with exponentially growing lengths $2^i h_0 b_0$ $(i = 0, 2, ...)$. Each phase is further partitioned evenly into h_0 sub-phases with length $2^i b_0$. We select $h_0 = \lceil \frac{K-1}{M-1} \rceil$ and $b_0 = M(M+2)$. For each phase, we assign arm rewards as follows: during each subphase $u = 0, 2, ..., h_0 - 1$, we let arm $(M(u+1) \text{ mod } K)$ offer reward 1 and the other arms offer reward 0. Since UCB-M selects arm groups with size M in a round-robin fashion, it is clear that arms selected by UCB-M offers 0 reward at almost all time steps. The weak regret of UCB-M is clearly linear in T. For HLMC, we adopt a two-level hierarchy and apply EXP3 to both group and arm levels. The simulation results on the expected weak regret are presented in Fig. 2.

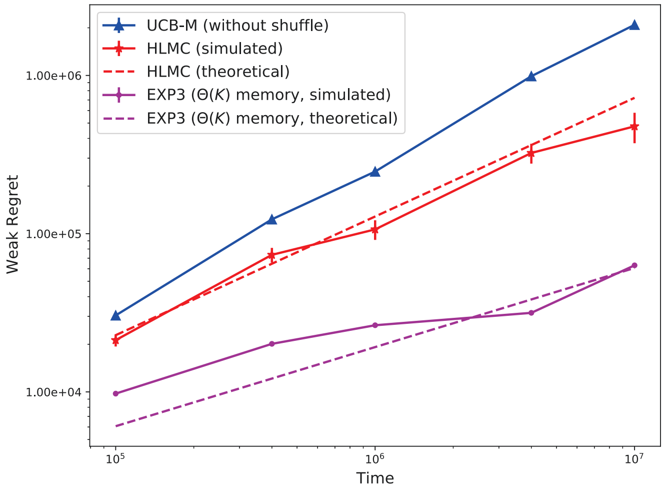

Fig. 2: Comparison of the weak regret of UCB-M (without random shuffle of arm indices), HLMC, and EXP3: $K = 100$, $M = 20$, and $T = 10^7$. The time horizon is partitioned into phases with exponentially growing lengths $2^i h_0 b_0$ $(i = 0, 2, ...)$. Each phase is partitioned evenly into h_0 sub-phases with length $2^i b_0$. During each subphase u, arm $(M(u+1) \text{ mod } K)$ offers reward 1 and the other arms offer reward 0.

From Fig. 2, we can observe that HLMC outperforms the UCB-M policy under the constructed adversarial environment. The error bar indicates that the proposed learning policy is robust with low variance. Note that although the EXP3 algorithm achieves the best performance, it requires $\Theta(K)$ memory size, which is infeasible in the memory-constrained setting. We also plot the theoretical upper bounds on the regret of HLMC and EXP3 (i.e., $2T'^{\frac{3}{4}}K^{\frac{1}{4}}(\ln K)^{\frac{1}{2}}$ and $2\sqrt{T'K\ln K}$ where $T' = T/5$ due to the fact that the cumulative reward of the best arm is $T/5$ instead of T in this experiment[2]), which verify that the expected weak regret of HLMC has the same order with the theoretical upper bounded established in Theorem 1.

We further use another example to show that even with the random shuffle step, UCB-M still fails to avoid a linear regret in T against adversaries. We consider the same experiment setup with a different reward assignment. Specifically, the phase and subphase partitions are the same with those in the first experiment. During each subphase $u = 0, 2, ..., h_0 - 1$, we let arm 1 offer $(u \text{ mod } 2)$ reward and the other arms offer $\epsilon = 1 \times 10^{-4}$ rewards. It is not difficult to check that after every time arm 1 is selected by UCB-M and offers reward 1, it will offer 0 reward in the next subphase and will be excluded from memory. Therefore, significant regret is incurred in the subphase after next, when arm 1 offers 1 reward again. Over the entire time horizon, UCB-M suffers a linear regret order in T. Moreover, we added another baseline: EXP3-M by changing the UCB subroutine in UCB-M to the EXP3 subroutine. The simulation results are presented in Fig. 3, which again verify the advantage of HLMC against UCB-M and EXP3-M. It should be noted that even with the random shuffle step or a subroutine developed for classic adversarial bandits during every epoch, the UCB-M and EXP3-M algorithms still suffer significant regret due to the fact that the algorithmic structure of the two algorithms fails to

[2] The choice of the constant in front of T, K does not change the regret order. To demonstrate that the theoretical regret bound and the simulated results have the same order, we set the constant equal to 2.

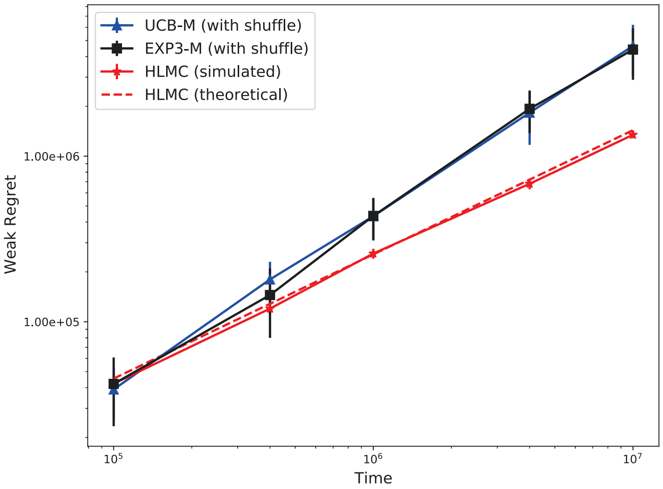

Fig. 3: Comparison of the weak regret of UCB-M (with random shuffle of arm indices), EXP3-M (with random shuffle of arm indices) and HLMC: the time partition is the same with that in Fig. 2. During each subphase u, arm 1 offers (u mod 2) reward and the other arms offer $\epsilon = 10^{-4}$ reward.

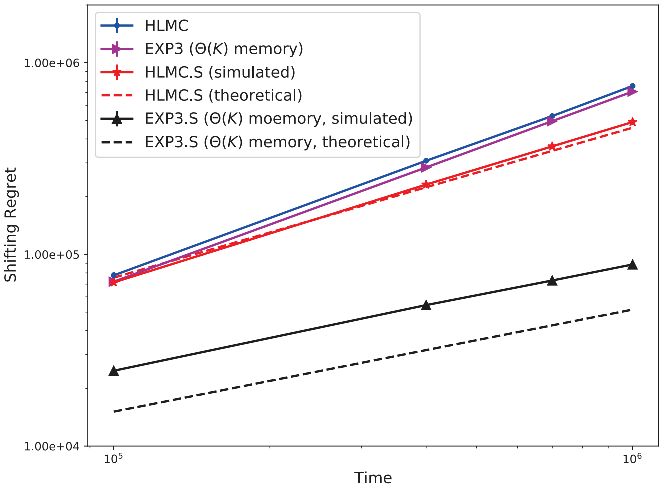

Fig. 4: Comparison of the shifting regret of HLMC.S, HLMC, EXP3, and EXP3.S: $K = 16, M = 8$, and $T = 10^6$. The time horizon is partitioned evenly into $V = 10$ phases. In phase $v = 0, 1, ..., V-1$, arm $i_v = (vN \bmod K)$ offer reward 1 and the other arms offer reward 0.

balance between what to remember and what to forget in the adversarial setting. Besides, the random shuffle step in UCB-M and EXP3-M introduces high variance with little improvement on the expected weak regret. The comparison between the theoretical upper bounds and the simulated results also verifies the correctness of our analysis in Theorem 1.

B. *Shifting Regret Minimization*

We further conduct an experiment to show the regret performance of HLMC with a two-level hierarchy under the notion of shifting regret. As discussed in Sec. IV-C, by adopting EXP3.S at the group level, HLMC achieves a sublinear scaling of shifting regret in T. In this experiment, we compare the performance of HLMC adopting EXP3.S at the group level and EXP3 at the arm level (referred to as HLMC.S in this subsection), HLMC adopting EXP3 at both group and arm levels (referred to as HLMC in this subsection), EXP3, and EXP3.S. The experiment is set up as follows: let $K = 16, M = 8$, and $T = 10^6$. The time horizon is partitioned evenly into $V = 10$ phases. In phase $v = 0, 1, ..., V-1$, we let arm $i_v = (vN \bmod K)$ offer reward 1 and the other arms offer reward 0 (N is the group size defined in the HLMC framework, which equals 4 in this experiment). It is clear that the best benchmark policy in the shifting regret definition with hardness V is to play the best arm i_v within every phase v. The simulation results are presented in Fig. 4.

It can be observed from Fig. 4 that HLMC.S designed for shifting regret minimization outperforms HLMC and EXP3 for weak regret minimization. Adopting EXP3.S at the group level of the HLMC framework improves the regret performance under the notion of shifting regret. Moreover, the error bar verifies the robustness of the proposed policies. It should be noted that although EXP3.S outperforms HLMC and HLMC.S, it requires $\Theta(K)$ memory space, which is inapplicable in the memory-constrained setting.

C. *Impact of Available Memory on Regret Performance*

In this subsection, we show the impact of the size of available memory space on the regret performance of HLMC. We use the same experiment setup with that in the first experiment in Sec. VI-A. We compare the weak regret of HLMC with $M = 14, 20, 50, 80$. Specifically, when $M = 14$, the HLMC framework requires a three-level hierarchy with $N_1 = 5, N_2 = 5$, and $N_3 = 4$. When $M = 20, 50, 80$, HLMC adopts two-level hierarchies with $N = \lceil \frac{M - \sqrt{M^2 - 4K}}{2} \rceil$ and $L = \lceil K/N \rceil$.

The results in Fig. 5 show that the regret performance of HLMC improves as the size of the memory space increases. In particular, adopting a hierarchy with fewer levels improves the regret order as indicated in Corollary 2. Even with the same number of levels, a smaller regret can be achieved with a larger memory space. Intuitively, as M increases, the epoch length Δ decrease. Since the reward sequence assigned in the experiment is stable within a short period but varies vastly in the long run (it has been argued in [25] that such a reward assignment is justified in various real-world applications), the arm-level regret is dominated by the group-level regret and the latter decreases with the epoch length. We also plot the theoretical upper bounds on the regret of HLMC with different levels of hierarchies. The comparison between the theoretical the simulated results verifies our analysis.

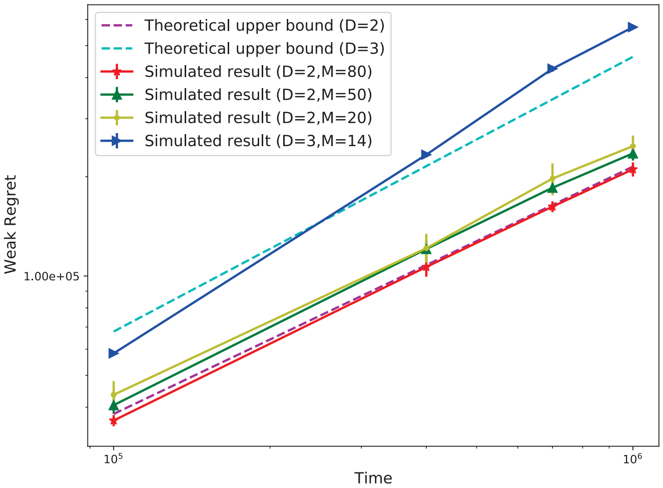

Fig. 5: Comparison of the weak regret of HLMC with $M = 14, 20, 50, 80$ memory space: the experiment setup is the same with that in Fig. 2. When $M = 14$, the HLMC framework adopts a three-level hierarchy with $N_1 = 5, N_2 = 5$, and $N_3 = 4$. When $M = 20, 50, 80$, HLMC adopts two-level hierarchies with $N = \lceil \frac{M-\sqrt{M^2-4K}}{2} \rceil$ and $L = \lceil K/N \rceil$.

D. Distributed Dynamic Spectrum Access in the Presence of Jamming

In this subsection, we consider the application of distributed dynamic spectrum access in the presence of jamming in multi-agent wireless communication systems. There are 1000 distributed agents competing for $K = 20$ channels (arms) and an attacker that is jamming the channels. The transmission rate of a channel is modeled as the reward of the corresponding arm. The quality of a channel depends on whether it is jammed by the attacker and how many distributed agents are accessing the channel simultaneously. Specifically, we assume that if a channel is not jammed, it offers reward 10 and all accessing agents evenly share the reward, i.e., every agent receives $10/n_t$ reward where n_t is the number of agents selecting the unjammed channel at time t. For the jammed channels, an agent can only receive 1 reward if there is no collision (if there are more than two agents selecting the same arm, no agent can receive reward from this arm). In this experiment, we consider an attacker that jams all but one channel at every time step t and the unjammed channel changes at the beginning of every phase and circulates among the K channels.

From the perspective of every agent, the problem can be modeled as a memory-constrained adversarial bandit problem studied in this paper. Due to limited memory on distributed wireless devices, every agent can store at most $M = 10$ statistics of arm rewards. We compare the per-agent average reward where each agent adopts HLMC with that adopting UCB-M (with and without random shuffle). The simulation result is shown in Fig. 6, which again demonstrates the advantage of HLMC against UCB-M in the adversarial setting.

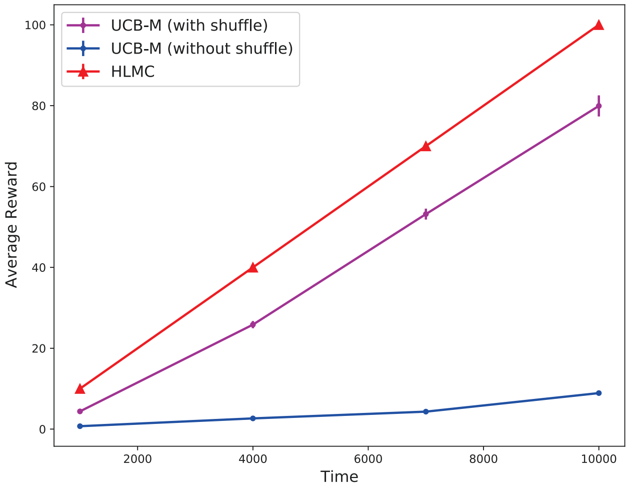

Fig. 6: Comparison of the average reward per agent by adopting HLMC and UCB-M (with and without random shuffle of arm indices) in dynamic spectrum access in the presence of jamming: 1000 distributed agents compete for $K = 20$ channels. An attacker adversarially jams all but one channel at every time step and the unjammed channel circulates among the K channels. For the unjammed channel, all accessing agents evenly share 10 reward. For the jammed channel, an agent can only receive 1 reward if there is no collision.

VII. Conclusions and Discussions

In this paper, we studied the problem of adversarial multi-armed bandits with memory constraints. We proposed a general hierarchical learning framework: HLMC that adopts a multi-level hierarchy to partition the arms into groups and the time horizon into epochs. The HLMC framework decouples the tradeoff between what to remember and what to forget induced by memory constraints from the one between exploration and exploitation due to bandit feedback. We showed in the two-level case that, by employing different existing learning routines developed for memory-unconstrained bandits at both levels of the hierarchy, HLMC achieves no-regret learning under various regret notions with a memory complexity sublinear in the number of arms. We further showed that through designing the depth of the adopted hierarchy, HLMC achieves different operating points at the tradeoff curve between the regret order and memory complexity. We conducted numerical experiments to verify the advantages of HLMC against existing baselines.

Several questions remain open in this problem. It is unclear whether $\Theta(\log K)$ is the minimum memory complexity required for achieving no-regret learning in the adversarial setting. Moreover, the current hierarchical partition of arm groups is pre-determined. It worth studying whether a dynamic (potentially stochastic) grouping strategy that depends on past observations can improve the regret performance. More importantly, whether the sequence of operating points offered by the proposed algorithm traces the Pareto front of this fundamental tradeoff between regret performance and memory complexity is an interesting open question that requires a separate full investigation. Another potential research direction is to find the best of both worlds, that is, a learning policy achieving the optimal regret orders in both stochastic and adversarial settings with memory constraints.

References

[1] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, no. 3/4, pp. 285–294, 1933.

[2] Q. Zhao, *Multi-Armed Bandits: Theory and Applications to Online Learning in Networks*. Morgan & Claypool Publishers, 2019.

[3] T. L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Advances in Applied Mathematics*, vol. 6, no. 1, pp. 4–22, 1985.

[4] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine Learning*, vol. 47, no. 2-3, pp. 235–256, 2002.

[5] A. Garivier and O. Cappé, "The KL-UCB algorithm for bounded stochastic bandits and beyond," in *Proceedings of the 24th Annual Conference on Learning Theory*, 2011, pp. 359–376.

[6] S. Vakili, K. Liu, and Q. Zhao, "Deterministic sequencing of exploration and exploitation for multi-armed bandit problems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 5, pp. 759–767, 2013.

[7] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "Gambling in a rigged casino: The adversarial multi-armed bandit problem," in *Proceedings of IEEE 36th Annual Foundations of Computer Science*. IEEE, 1995, pp. 322–331.

[8] N. Cesa-Bianchi and G. Lugosi, *Prediction, learning, and games*. Cambridge university press, 2006.

[9] H. P. Young, *Strategic learning and its limits*. OUP Oxford, 2004.

[10] T. Lykouris, V. Syrgkanis, and É. Tardos, "Learning and efficiency in games with dynamic population," in *Proceedings of the twenty-seventh annual ACM-SIAM symposium on Discrete algorithms*. SIAM, 2016, pp. 120–129.

[11] B. Duvocelle, P. Mertikopoulos, M. Staudigl, and D. Vermeulen, "Learning in time-varying games," *arXiv preprint arXiv:1809.03066*, 2018.

[12] X. Xu and Q. Zhao, "Distributed no-regret learning in multiagent systems: Challenges and recent developments," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 84–91, 2020.

[13] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire, "The nonstochastic multiarmed bandit problem," *SIAM Journal on Computing*, vol. 32, no. 1, pp. 48–77, 2002.

[14] J.-Y. Audibert and S. Bubeck, "Minimax policies for adversarial and stochastic bandits," in *Proceedings of the 22nd Annual Conference on Learning Theory*, 2009, pp. 217–226.

[15] S. Bubeck and N. Cesa-Bianchi, "Regret analysis of stochastic and nonstochastic multi-armed bandit problems," *Machine Learning*, vol. 5, no. 1, pp. 1–122, 2012.

[16] M. Bande and V. V. Veeravalli, "Adversarial multi-user bandits for uncoordinated spectrum access," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 4514–4518.

[17] N. M. Vural, H. Gokcesu, K. Gokcesu, and S. S. Kozat, "Minimax optimal algorithms for adversarial bandit problem with multiple plays," *IEEE Transactions on Signal Processing*, vol. 67, no. 16, pp. 4383–4398, 2019.

[18] D. Liau, Z. Song, E. Price, and G. Yang, "Stochastic multi-armed bandits in constant space," in *International Conference on Artificial Intelligence and Statistics*, 2018, pp. 386–394.

[19] A. R. Chaudhuri and S. Kalyanakrishnan, "Regret minimisation in multi-armed bandits using bounded arm memory," *arXiv preprint arXiv:1901.08387*, 2019.

[20] S. Bubeck, R. Munos, and G. Stoltz, "Pure exploration in multi-armed bandits problems," in *International Conference on Algorithmic Learning Theory*. Springer, 2009, pp. 23–37.

[21] H. Robbins, "A sequential decision problem with a finite memory," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 42, no. 12, p. 920, 1956.

[22] T. M. Cover, "A note on the two-armed bandit problem with finite memory," *Information and Control*, vol. 12, no. 5, pp. 371–377, 1968.

[23] T. Cover and M. Hellman, "The two-armed-bandit problem with time-invariant finite memory," *IEEE Transactions on Information Theory*, vol. 16, no. 2, pp. 185–195, 1970.

[24] C.-J. Lu and W.-F. Lu, "Making online decisions with bounded memory," in *International Conference on Algorithmic Learning Theory*. Springer, 2011, pp. 249–261.

[25] J. Zimmert, H. Luo, and C.-Y. Wei, "Beating stochastic and adversarial semi-bandits optimally and simultaneously," in *International Conference on Machine Learning*, 2019, pp. 7683–7692.