Reward Teaching for Federated Multi-armed Bandits

Chengshuai Shi*, Wei Xiong†, Cong Shen*, and Jing Yang‡
*University of Virginia, Charlottesville, VA 22904, USA
†The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong
‡The Pennsylvania State University, University Park, PA 16802, USA

2023 IEEE International Symposium on Information Theory (ISIT) | 978-1-6654-7554-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/ISIT54713.2023.10206444

***Abstract*—Most existing federated multi-armed bandits (FMAB) designs are based on the presumption that clients will implement the new design to collaborate with the server. In reality, however, it may not be possible to modify the client protocols. Motivated by this limitation, this work focuses on clients who always maximize their individual cumulative rewards, and introduces a novel idea of *reward teaching*, where the server guides the clients towards global optimality through implicit local reward adjustments. Under this framework, the server faces two *tightly coupled* tasks of bandit learning and target teaching, whose combination is non-trivial and challenging. A novel algorithm, called *Teaching-After-Learning (TAL)*, is proposed, which encourages and discourages clients' explorations separately. General performance analyses of TAL on regret and cost are first established when the clients' strategies satisfy certain requirements. To particularize the results, clients with UCB or ε-greedy strategies are then considered, where novel technical approaches are developed to analyze their *warm-start* behaviors. The obtained guarantees concretely demonstrate that when facing these client strategies, TAL achieves logarithmic regrets while only incurring logarithmic adjustment costs, which is order-optimal w.r.t. a natural lower bound.**

I. Introduction

Federated multi-armed bandits (FMAB) [2]–[8] is a recently proposed framework that introduces the core principles of federated learning (FL) [9] into multi-armed bandits (MAB) [10]. In particular, FMAB often considers a system of one global server and multiple *heterogeneous* local clients with the goal of having the clients converge to the *global* optimality.

One practical difficulty of realizing existing FMAB designs is to implement new protocols for both the server and clients [3], [4], [11]. Specifically, the server and clients are required to follow the carefully crafted designs collaboratively. In real-world applications, it is relatively easy to update the server's protocols for FMAB. However, given the typically large number of clients, it is often not realistic to assume that all of their protocols can be updated, as it would result in a significant infrastructure cost. For example, in cognitive radio systems (which is a common motivating application for FMAB), mobile devices (i.e., clients) are often configured to optimize their individual communication qualities following their built-in protocols. It is often hard and expensive to modify these devices to follow the FMAB designs, especially since such updates are often needed for both software and hardware.

This work removes this limitation by *designing mechanisms only on the server's side*. Especially, the clients can still follow the original routines to optimize their individual performances (as in the aforementioned cognitive radio example) and no change of their protocols is required. Towards this end, a novel **"reward teaching"** approach is proposed: the server implicitly adjusts the local rewards perceived by the clients to indirectly influence their decision-making. This idea is practical for cognitive radio, as it is widely adopted in standard communication protocols for the server to measure rewards (e.g., throughput) and send designed signals to mobile devices.

The seemingly simple idea of reward teaching brings considerable challenges for the server strategy. In particular, the server faces the following two tasks *simultaneously*: **bandit learning** and **target teaching**. On one hand, the server has to learn the *unknown* global model through the clients' actions, which are based on local observations and may not align with the server's objectives. Thus, reward adjustments should be carefully placed to have the clients explore with respect to (w.r.t.) the global information (instead of their local ones). On the other hand, even if the global model is learned successfully, the corresponding learning history has a cumulative effect on guiding the clients towards the learned target, as all historical (adjusted) rewards are considered in clients' future decision-making. As a result, while having been studied individually (e.g., learning in MAB and teaching in data-poisoning MAB), the combination of the aforementioned two tasks is novel and challenging as they are *tightly coupled*.

The contributions of this work are summarized as follows.

- **A reward-teaching framework.** A novel idea of reward teaching is proposed to let the server design reward signals to guide clients with their own local strategies. This idea is practically appealing for FMAB systems as existing client protocols do not have to be modified – only the reward signals they receive are adjusted. From another perspective, it also provides a method to handle non-naive FMAB clients.
- **Client-strategy-agnostic algorithm design.** A phased approach, coined *"Teaching-After-Learning" (TAL)*, is first proposed. It addresses the challenge of teaching in an unknown environment by separately encouraging and discouraging explorations in two phases. It is worth noting that the design of TAL is agnostic to the clients' local strategies.
- **Client-strategy-dependent analysis.** Theoretical regret and cost guarantees of TAL are first established when the clients' local strategies satisfy some general properties. Particularizing these properties to UCB1 and ε-greedy [12] strategies

A full version of this paper can be found in [1]. The work of CSs was supported in part by the US National Science Foundation (NSF) under awards 2029978, 2143559, 2002902, Virginia Commonwealth Cyber Initiative, and the Bloomberg Data Science Ph.D. Fellowship. The work of JY was supported in part by the US NSF under awards 2030026, 2114542, and 1956276.

at clients reveals that TAL can achieve a logarithmic regret while only incurring a logarithmic adjustment cost, which is order-optimal w.r.t. a natural lower bound. Moreover, the novel technical approaches to analyzing *warm-start* bandit clients may be of independent merit.

II. Problem Formulation

A. Federated Multi-armed Bandits

Local and global models. Following [2]–[4], a standard FMAB system of M local models and one global model is considered. With the same set of K arms shared by all the models, at each time step $t \in [T]$, each arm $k \in [K]$ is associated with a local reward $X_{k,m}(t) \in [0,1]$ for each local model $m \in [M]$ and a global reward $Y_k(t) \in [0,1]$ for the global model. These rewards of each arm k are all independently sampled with unknown expectations denoted as $\mu_{k,m} := \mathbb{E}[X_{k,m}(t)], \forall m \in [M]$ and $\nu_k := \mathbb{E}[Y_k(t)]$. In general, the local arm utilities are model-dependent, i.e., $\mu_{k,m}$ may not equal to $\mu_{k,n}$ for $n \neq m$. The optimal local arm for each local model m is denoted as $k_{*,m} := \arg\max_{k\in[K]} \mu_{k,m}$ with $\mu_{*,m} := \mu_{k_{*,m},m}$, and the optimal global arm as $k_\dagger := \arg\max_{k\in[K]} \nu_k$ with $\nu_\dagger := \nu_{k_\dagger}$.

As in [2]–[4], we consider the setting where each arm k's mean reward on the global model is the average of its mean rewards on the local models, i.e., $\nu_k := \mathbb{E}[Y_k(t)] = \frac{1}{M}\sum_{m\in[M]} \mu_{k,m}$. A global-local misalignment may occur as the global optimality may not align with each local optimality, i.e., $k_\dagger$ may not be the same as $k_{*,m}$ for all or part of $m \in [M]$.

Clients and server. In FMAB, there exist M clients and one server. At time t, each client $m \in [M]$ selects an arm $\pi_m(t)$ (referred to as "local actions") and then observes its local reward $X_{\pi_m(t),m}(t)$ on local model m. Additionally, each client m's action $\pi_m(t)$ would also generate a reward $Y_{\pi_m(t)}(t)$ from the global model. It would be helpful to interpret the local and global rewards as the individual-level and system-level impacts from the clients' actions.

The server in FMAB does not perform any arm-pulling action herself. Instead, she focuses on guiding the local actions to optimize their incurred *global rewards*. However, the global rewards are not directly observable by the server and the clients, which is often a result of practical measurement limitations [3]. Instead, the server is assumed to be able to observe the local actions and the corresponding local rewards, i.e., $\{\pi_m(t), X_{\pi_m(t),m}(t) : m \in [M]\}$.

To optimize global performance, previous FMAB studies require clients to work collaboratively following new local protocols. Instead, this work considers that clients are fully committed to interacting with their *own* local models (i.e., client m with local model m). Then, the clients would naturally adopt their own MAB policies to maximize their local rewards. This setting is practically appealing as in many applications (e.g., the cognitive radio example in Sec. I), the local clients are inherently configured to perform local policies to optimize their local performance (e.g., IoT devices). Specifically, at time t, each client m *individually* makes an arm-pulling decision $\pi_m(t)$ based on her own history observed on local model m, i.e., $H_m(t-1) := \{\pi_m(\tau), X_{\pi_m(\tau),m}(\tau) : \tau \in [t-1]\}$.

B. Reward Teaching

As mentioned, each client would select suitable actions w.r.t. her own local rewards, which however may not necessarily meet the server's preference due to the global-local model misalignment. To address this challenge, the following reward-teaching mechanism is introduced for the server to indirectly influence the clients' action selections.

Specifically, after observing $\{X_{\pi_m(t),m}(t) : m \in [M]\}$, the server can adjust each client m's local reward $X_{\pi_m(t),m}(t)$ to $X'_{\pi_m(t),m}(t)$ by an amount of $\sigma_m(t)$, i.e., $X'_{\pi_m(t),m}(t) := X_{\pi_m(t),m}(t) + \sigma_m(t)$, which is then revealed to the client (instead of $X_{\pi_m(t),m}(t)$). Note that one implicit constraint is that the adjusted rewards must still be in $[0,1]$, which is the system limitation. If this constraint is satisfied, the clients are assumed to be unable to detect the reward adjustments. The adjusted rewards lead to an adjusted history of $H'_m(t) := \{\pi_m(\tau), X'_{\pi_m(\tau),m}(\tau) : \tau \in [t]\}$ for client m, which ideally can shape her future actions in favor of the server.

It is worth mentioning that such reward adjustments are practical for FMAB applications. In the cognitive radio example, it is common for the base station to first measure the communication quality (via pilot signals) and then send *designed* feedback to the devices; this is the case in both cellular and WiFi. Adjusting rewards can be achieved via either sending modified feedback signals or modifying the allocated resources (e.g., bandwidth) to boost or reduce client performance, which is standard in modern communication protocols. The devices, on the other hand, are oblivious to such adjustments thanks to their built-in protocols.

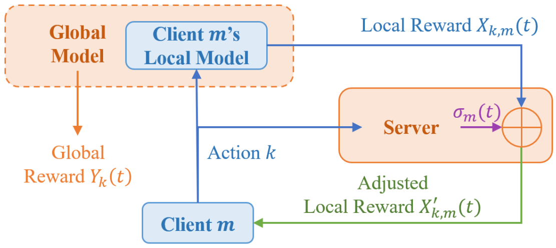

Fig. 1. The reward-teaching process with client m (among the overall M clients) and action $\pi_m(t) = k$.

The reward-teaching process is summarized as the following steps, which is also illustrated in Fig. 1:

- Each client m chooses $\pi_m(t)$ using history $H'_m(t-1)$;
- The server observes $\{\pi_m(t), X_{\pi_m(t),m}(t) : m \in [M]\}$;
- The server adjusts $X_{\pi_m(t),m}(t)$ into $X'_{\pi_m(t),m}(t)$ by the amount of $\sigma_m(t)$ for each client $m \in [M]$;
- Each client m observes the adjusted $X'_{\pi_m(t),m}(t)$.

C. Learning Objectives

Following previous FMAB studies, we focus on the global view of the server, which leads to a two-fold objective. First, the server's main goal is to maximize the cumulative global rewards and can be characterized by minimizing the notion of *global regret* defined as $R_F(T) := \sum_{m\in[M]} R_m(T)$, where $R_m(T)$ is the regret incurred by client m's actions w.r.t. the global model (instead of her local model) defined as $R_m(T) := T\nu_\dagger - \mathbb{E}[\sum_{t\in[T]} Y_{\pi_m(t)}(t)]$. The expectation is w.r.t. both the reward generations and the client-system interactions.

Also, the server's adjustments on local rewards are quantified by the *cumulative cost* defined as $C_F(T) := \sum_{m\in[M]} C_m(T)$, where $C_m(T)$ denotes the overall cost spent on client m and is defined as $C_m(T) := \mathbb{E}[\sum_{t\in[T]} |\sigma_m(t)|]$. The subscripts F refer to the global model (i.e., the federation).

Intuitively, there exists a trade-off between these two objectives: with more adjustments on rewards, i.e., larger $C_F(T)$, the server can have bigger impacts on the clients' selections of actions, which ideally would decrease the regret $R_F(T)$. It is thus important to strike a balance between these two objectives, which is the focus of the remainder of this paper.

D. Client Strategies

To facilitate discussion, we denote client m's local strategy as Π_m. Note that while performing their own policies, the clients are assumed not to be strategically against the server, which is reasonable for most of the real-world applications of FMAB, e.g., autonomous but not fully flexible mobile devices in cognitive radio [3]. In addition, we denote $N_{k,m}(t)$ as the number of pulls by client m on arm k by time t, and $N^{-1}_{k,m}(\tau)$ refers to time step t such that $N_{k,m}(t) = \tau$.

III. Two Coupled Tasks and Design Objectives

In this section, two tightly coupled tasks faced by the reward-teaching server, bandit learning and target teaching, are elaborated. A reasonable design objective is also proposed.

Bandit learning. One major distinction between learning in FMAB and in classical MAB [10], [13] is that the server can only gather information through clients' local actions. Previous FMAB studies tackled this challenge by proposing new protocols for clients to naively follow [2]–[4], [11]. In contrast, in this work, such information collection can only be indirectly guided via carefully designed rewards.

Target teaching. To understand teaching, a special case is first considered where the optimal arm $k_\dagger$ is known by the server. Then, the goal is to assign adjustments to have the clients pull the *pre-specified* arm $k_\dagger$ as much as possible, which is mathematically the same as the *data-poisoning MAB* problem [14]–[17]. However, in this work, the identity of $k_\dagger$ is not available in advance.

Combination leads to a tight coupling. While both tasks have been separately investigated (to some extent), the reward-teaching server faces a combination of them. On one hand, even if the server can perfectly learn the global model, she still needs to teach it to the clients. On the other hand, to teach correctly, sufficient information should be learned by the server. The resulted *tight coupling* is the major challenge of the design. Specifically, the learning attempt has cumulative effects on teaching, which in return relies on the learned target. One consequent major difficulty is the analysis of the "warm-start" behaviors of bandit algorithms.

Design objective. For the cost, with a known target arm, [18], [19] prove lower bounds that *with UCB1 and ε-greedy clients (defined in Sec. V), it is necessary to spend a cost $C_m(T) = \Omega(\log(T))$ to obtain a regret $R_m(T) = O(\log(T))$.* Thus, with M *independent* FMAB clients, a cost of $C_F(T) = \Omega(M\log(T))$ is required to obtain a regret of $R_F(T) = O(M\log(T))$ while knowing arm $k_\dagger$, which naturally holds for the more stringent case of not knowing the target $k_\dagger$. For the regret, UCB1 and ε-greedy clients can be shown to be conservative [18] as each client m would pull each arm at least $\Omega(\log(T))$ times regardless of the rewards; thus $R_m(T) = \Omega(\log(T))$ and $R_F(T) = \Omega(M\log(T))$.

With these results, the following design goal, order-wise tight w.r.t. both criteria, is established:

> ***Goal:*** *Design algorithms to achieve both*
> $R_F(T) = O(M\log(T))$ *and* $C_F(T) = O(M\log(T))$.

IV. Algorithm Design

To address the coupled tasks of bandit learning and target teaching, one idea is to first learn the server's target and then teach the clients to converge to it, which leads to the proposed "Teaching-After-Learning" (TAL) algorithm (presented in Alg. 1). Specifically, it starts with the learning phase to identify the optimal global arm. Then, in the teaching phase, the server guides the clients toward the learned global optimality. Note that although there is a separation of phases, the teaching phase must handle clients that accumulate observations from the learning phase (i.e., "warm-start" clients).

In the learning phase, TAL uniformly adjusts each client m's observed rewards to γ_1, i.e., $\sigma_m(t) \leftarrow \gamma_1 - X_{\pi_m(t),m}(t)$, where $\gamma_1 \in [0,1]$ is a to-be-specified input parameter. Intuitively, this uniform reward adjustment encourages sufficient (or ideally, uniform) explorations among all arms, since their rewards are all γ_1's. Thus, the server can collect enough information on each arm to identify her optimal arm $k_\dagger$.

This identification is designed to proceed in epochs indexed by counter ψ to ensure statistical independence. If at time t, each client m has pulled each arm k at least $F(\psi) := \sum_{\tau\in[\psi]} f(\tau)$ times, where $f(\psi) := \frac{1}{M}\cdot 2^{2\psi+3}\log(2KT^2)$, the server updates upper and lower confidence bounds (UCB and LCB) for each arm $k\in[K]$ using its rewards collected between its $F(\psi-1)+1$ and $F(\psi)$ pulls (i.e., overall $f(\psi)$ pulls) by each client as follows:

$$\mathrm{UCB}_k(\psi), \mathrm{LCB}_k(\psi) := \sum\nolimits_{m\in[M]} \hat{\mu}_{k,m}(\psi)/M \pm \mathrm{CB}(\psi), \qquad (1)$$

where $\hat{\mu}_{k,m}(\psi) := \frac{1}{f(\psi)}\sum_{\tau=F(\psi-1)+1}^{F(\psi)} X_{k,m}(N^{-1}_{k,m}(\tau))$ and $\mathrm{CB}(\psi) := \sqrt{\log(2KT^2)/(2Mf(\psi))} = 2^{-\psi-2}$. Note that with the estimation of $\mu_{k,m}$ from local samples, the first term in Eqn. (1) is essentially an estimation $\hat{\nu}_k(\psi)$ of ν_k. The confidence bound $\mathrm{CB}(\psi)$ is specifically designed s.t. $\mathrm{LCB}(\psi) \le \nu_k \le \mathrm{UCB}(\psi)$ holds for each arm k and each epoch ψ in the learning phase with high probability.

The learning phase ends in epoch ψ if the confidence interval of one arm $k_\ddagger$ dominates that of all other arms, i.e.,

$\mathrm{LCB}_{k_\ddagger}(\psi) \geq \mathrm{UCB}_k(\psi), \forall k \neq k_\ddagger$, which is recognized as the optimal arm. Otherwise, a new epoch $\psi + 1$ begins. With the designed confidence bound, this identification is guaranteed to be correct with high probability.

With the identified arm $k_\ddagger$, the server utilizes the following adjustments to guide the clients in the teaching phase:

$$\sigma_m(t) \leftarrow \begin{cases} \gamma_2 - X_{\pi_m(t),m}(t) & \text{if } \pi_m(t) \neq k_\ddagger \\ 0 & \text{if } \pi_m(t) = k_\ddagger \end{cases}, \qquad (2)$$

where γ_2 is another to-be-specified input parameter and typically should be small. In other words, if the client does not pull arm $k_\ddagger$, her reward is adjusted to a small value γ_2 to discourage explorations; otherwise, the original reward of arm $k_\ddagger$ is kept unchanged to save adjustments.

From Alg. 1, it can be observed that TAL is a pure server protocol and agnostic to the clients' local strategies – the only interaction with the clients is the adjusted rewards.

Algorithm 1 TAL (with input $\gamma_1, \gamma_2 \in [0, 1]$)

```
Initialize: $F \leftarrow 1$ (i.e., the learning phase); $\psi \leftarrow 1$; $k_\ddagger \leftarrow 0$
for $t \leq T$ do
    Observe $\{\pi_m(t), X_{\pi_m(t),m}(t) : m \in [M]\}$
    if $F = 1$ & $N_{k,m}(t) \geq F(\psi), \forall m \in [M], k \in [K]$ then
        Update $\{\mathrm{UCB}_k(\psi), \mathrm{LCB}_k(\psi) : k \in [K]\}$ as Eqn. (1)
        if $\exists j \in [K], \mathrm{LCB}_j(\psi) \geq \mathrm{UCB}_k(\psi), \forall k \neq j$ then
            Set $k_\ddagger \leftarrow j$; $F \leftarrow 2$ (i.e., the teaching phase)
        else Set $\psi \leftarrow \psi + 1$
        end if
    end if
    if $F = 1$ then $\sigma_m(t) \leftarrow \gamma_1 - X_{\pi_m(t),m}(t), \forall m \in [M]$
    else if $F = 2$ then Set $\sigma_m(t)$ as Eqn. (2), $\forall m \in [M]$
    end if
    Set $X'_{\pi_m(t),m}(t) \leftarrow X_{\pi_m(t),m}(t) + \sigma_m(t), \forall m \in [M]$
    Reveal $X'_{\pi_m(t),m}(t)$ to each client $m \in [M]$
end for
```

V. Performance Analysis

We first provide a general analysis of TAL under a few identified properties for clients' strategies. Then, we consider clients with UCB1 or ε-greedy algorithms and show that TAL achieves order-optimal performance with these clients.

Some useful notations are introduced as follows: $\Delta_k := \nu_\dagger - \nu_k, \forall k \neq k_\dagger$, $\Delta_{\min} = \Delta_{k_\dagger} := \min_{k \neq k_\dagger} \Delta_k$, $\Delta_{\max} := \max_{k \in [K]} \Delta_k$, and $\mu_{\dagger,m} := \mu_{k_\dagger,m}$. Moreover, $\delta_{k,m}(\gamma) := \mathbb{E}[|\gamma - X_{k,m}(t)|]$ and $\psi_{\max} := \lceil \log_2(1/\Delta_{\min}) \rceil$.

We first define sufficiently exploring algorithms for the learning phase in TAL, which states that a bandit algorithm would sufficiently explore when facing uniform rewards.

Definition 1 (Sufficiently Exploring Algorithms). *Consider a K-armed bandit environment where rewards from arms in a set $\mathcal{I} \subseteq [K]$ are always a fixed constant $\gamma \in [0, 1]$. In this environment, a bandit algorithm Π is said to be $(\mathcal{I}, \gamma, \underline{\eta}, \overline{\eta})$-sufficiently exploring if it would pull each arm in the set $\mathcal{I}$ at least $\underline{\eta}(\tau; \gamma, \mathcal{I})$ and at most $\overline{\eta}(\tau; \gamma, \mathcal{I})$ times when total τ pulls have been performed on set $\mathcal{I}$.*

If local strategies are sufficiently exploring, enough information can be collected in the learning phase to identify the global optimal arm, as stated in the following lemma, where $\underline{\eta}^{-1}(N; \gamma, [K])$ denotes the value τ s.t. $\underline{\eta}(\tau; \gamma, [K]) = N$.

Lemma 1 (Learning Phase in TAL). *If Π_m is $([K], \gamma_1, \underline{\eta}_m, \overline{\eta}_m)$-sufficiently exploring for all $m \in [M]$, with probability (w.p.) at least $1 - 1/T$, the learning phase ends with $k_\ddagger = k_\dagger$ by time step T_1, and the regret and cost in the learning phase of TAL are bounded, respectively, as*

$$R_{F,1}(T) \leq \sum\nolimits_{m \in [M]} \sum\nolimits_{k \neq k_\dagger} \Delta_k \cdot \overline{\eta}_m(T_1; \gamma_1, [K]);$$
$$C_{F,1}(T) \leq \sum\nolimits_{m \in [M]} \sum\nolimits_{k \in [K]} \delta_{k,m}(\gamma_1) \cdot \overline{\eta}_m(T_1; \gamma_1, [K]),$$

where $T_1 \leq \max_{m \in [M]} \{\underline{\eta}_m^{-1}(F(\psi_{\max}); \gamma_1, [K])\}$.

The sufficiently exploring lower bound (i.e., $\underline{\eta}$) ensures sufficient information collection, while the corresponding upper bound (i.e., $\overline{\eta}$) guarantees performance.

Then, for the teaching phase, since the cumulative observations from the learning phase are inherited to the client strategies, we can view the clients as "warm-started". The following notion of warm-start pulls is introduced, which measures the warm-start behavior of an algorithm.

Definition 2 (Warm-start Pulls). *In a K-armed bandit environment $\mathcal{B}$, if a reward sequence $H = \{H_k : k \in [K]\}$ is input to a bandit algorithm Π, where H_k is a reward sequence for arm k, warm-start pulls on arm k is defined as $\iota_k(T; H, \mathcal{B}, \Pi) := \mathbb{E}_\Pi[\sum_{t \in [T]} \mathbb{1}\{\pi(t) = k\} | H, \mathcal{B}]$, which represents the expected pulls performed by Π on each arm k during T steps in environment $\mathcal{B}$ with prior input H.*

Using this notion of warm-start pulls, the following guarantee on the teaching phase can be established.

Lemma 2 (Teaching Phase in TAL). *If the event in Lemma 1 occurs, the regret and cost in the teaching phase of TAL are bounded, respectively, as*

$$R_{F,2}(T) \leq \sum_{m \in [M]} \max_{H_m \in \mathcal{H}_m} \sum_{k \neq k_\dagger} \Delta_k \cdot \iota_k(T; H_m, \mathcal{B}_m, \Pi_m);$$
$$C_{F,2}(T) \leq \sum_{m \in [M]} \max_{H_m \in \mathcal{H}_m} \sum_{k \neq k_\dagger} \delta_{k,m}(\gamma_2) \cdot \iota_k(T; H_m, \mathcal{B}_m, \Pi_m),$$

where $\mathcal{B}_m$ denotes an environment with constant rewards as γ_2 for arm $k \neq k_\dagger$ and stochastic rewards with expectation $\mu_{\dagger,m}$ for arm $k_\dagger$. The set $\mathcal{H}_m$ is defined with each element of it as a reward sequence $H_m = \{H_{k,m} : k \in [K]\}$ where $H_{k,m} \in \{\{\gamma_1\}^\tau : \tau \in [\underline{\eta}_m(T_1; \gamma_1, [K]), \overline{\eta}_m(T_1; \gamma_1, [K])]\}$.

Note that $\mathcal{B}_m$ characterizes the environment of client m in the teaching phase while $\mathcal{H}_m$ represents the cumulative observation inherited from the learning phase. As long as the warm-start pulls on the sub-optimal arms are low, the regret and cost in the teaching phase can be bounded.

Finally, the overall performance guarantee can be obtained.

Theorem 1 (Overall Performance of TAL). *Under the assumption in Lemma 1, with $R_{F,1}(T), R_{F,2}(T)$ defined in Lemma 1 and $C_{F,1}(T), C_{F,2}(T)$ in Lemma 2, the regret and cost of TAL* *are bounded, respectively, as* $R_F(T) \leq R_{F,1}(T) + R_{F,2}(T) + O(M)$ *and* $C_F(T) \leq C_{F,1}(T) + C_{F,2}(T) + O(M)$.

A. UCB Clients

The popular UCB-type algorithms are first considered to particularize the general performance guarantee. In particular, we focus on the celebrated UCB1 algorithm [12] while noting that the analysis generalizes to other UCB variants [20], [21]. Especially, at time t, the UCB1 algorithm for client m chooses arm $\pi_m(t) = \arg\max_{k\in[K]}\{\hat{\mu}'_{k,m}(t-1) + \sqrt{2\log(t)/N_{k,m}(t-1)}\}$, which the perceived sample mean $\hat{\mu}'_{k,m}(t) := \sum_{\tau\in[N_{k,m}(t)]} X'_{k,m}(N^{-1}_{k,m}(\tau))/N_{k,m}(t)$.

First, the sufficiently exploring assumption in Lemma 1 is verified in Lemma 3. This is intuitive as with constant rewards, the sample means are the same while additional pulls decrease the confidence bound in UCB1.

Lemma 3. *For any* $\gamma \in [0,1]$ *and set* $\mathcal{I} \subseteq [K]$, *UCB1 is* $(\mathcal{I}, \gamma, \underline{\eta}, \overline{\eta})$*-sufficiently exploring with* $\underline{\eta}(\tau;\gamma,\mathcal{I}) = \lfloor \tau/|\mathcal{I}| \rfloor$ *and* $\overline{\eta}(\tau;\gamma,\mathcal{I}) = \lceil \tau/|\mathcal{I}| \rceil$.

Then, the performance of TAL in the learning phase (in Lemma 1) can be bounded by recognizing $T_1 = O(K\log(KT)/(M\Delta^2_{\min}))$, which further specifies the reward sequence set $\mathcal{H}_m$ in Lemma 2 and leads to the following lemma on the warm-start pulls of UCB1.

Lemma 4. *If* $\gamma_1 \geq \mu_{\dagger,m} > \gamma_2$ *and* Π_m *is UCB1, for all* $k \neq k_\dagger$, *it holds that* $\max_{H_m\in\mathcal{H}_m}\{\iota_k(T;H_m,\mathcal{B}_m,\Pi_m)\} = O\Big(\frac{(\gamma_1-\gamma_2)T_1}{K(\mu_{\dagger,m}-\gamma_2)} + \frac{\log(KT)}{(\mu_{\dagger,m}-\gamma_2)^2}\Big)$.

Proving this lemma is non-trivial and may be of independent interest in understanding the warm-start behavior of UCB1. Essentially, the result can be interpreted as first offsetting the "warm-start" history (the first term) and then converging to arm $k_\dagger$ (the second term) in a environment $\mathcal{B}_m$, whose rewards for arm $k \neq k_\dagger$ are constant γ_2's and rewards for arm $k_\dagger$ have an expectation $\mu_{\dagger,m}$ (see Lemma 2).

It is noted that Lemma 4 first requires $\gamma_1 \geq \mu_{\dagger,m}$, which maintains the optimism for the estimation of arm $k_\dagger$ on each local model m. The other requirement $\mu_{\dagger,m} > \gamma_2$ is intuitive as otherwise the local client m would not converge to arm $k_\dagger$. Since there is no prior information about $\mu_{\dagger,m}$. a feasible and sufficient solution is to set $\gamma_1 = 1$ while $\gamma_2 = 0$, which leads to the following theorem.

Theorem 2 (TAL with UCB1 clients)**.** *For TAL with* $\gamma_1 = 1$ *and* $\gamma_2 = 0$, *if all clients run UCB1 locally and* $\mu_{\dagger,m} \neq 0$ *for all* $m \in [M]$, *it holds that*

$$R_F(T) = O\Big(\sum_{m\in[M]}\sum_{k\neq k_\dagger}\Big[\frac{\Delta_k\log(KT)}{\mu_{\dagger,m}M\Delta^2_{\min}} + \frac{\Delta_k\log(KT)}{\mu^2_{\dagger,m}}\Big]\Big);$$

$$C_F(T) = O\Big(\sum_{m\in[M]}\sum_{k\in[K]}\frac{(1-\mu_{k,m})\log(KT)}{M\Delta^2_{\min}} + \sum_{m\in[M]}\sum_{k\neq k_\dagger}\Big[\frac{\mu_{k,m}\log(KT)}{\mu_{\dagger,m}M\Delta^2_{\min}} + \frac{\mu_{k,m}\log(KT)}{\mu^2_{\dagger,m}}\Big]\Big).$$

We note that the regret and cost are both of order $O(M\log(T))$; thus TAL is order-optimal w.r.t. both criteria in this scenario according to Sec. III. Moreover, the regret shows two dominating terms, which are from Lemma 4. In fact, there is another non-dominating (thus hidden) term from Lemma 1 for the learning phase. A similar three-part form is shared by the cost: the first term is from the learning phase while the last two terms are from the teaching phase.

B. ε-greedy Clients

We further consider clients running the well-known ε-greedy algorithm [22]. Especially, the ε-greedy algorithm for client m chooses arm $\pi_m(t) = \arg\max_{k\in[K]}\hat{\mu}'_{k,m}(t-1)$ with probability $1-\varepsilon_m(t)$; otherwise, arm $\pi_m(t)$ is selected uniformly random from $[K]$, where the exploration probability $\varepsilon_m(t) \in [0,1]$ is taken as $\varepsilon_m(t) = O(K/t)$, following [12].

First, the sufficiently-exploring property is verified.

Lemma 5. *For any* $\gamma \in [0,1]$, *if ties among arms are broken uniformly at random, with probability at least* $1-1/T$, *ε-greedy is* $([K], \gamma, \underline{\eta}, \overline{\eta})$*-sufficiently exploring with* $\underline{\eta}(\tau;\gamma,[K]) = O(\tau/K - \log(KT))$ *and* $\overline{\eta}(\tau;\gamma,[K]) = O(\tau/K + \log(KT))$.

Due to the randomness in ε-greedy, it is complicated to analyze its warm-start pulls in general. Instead, the following lemma focuses on $\gamma_1 = \gamma_2 = 0$.

Lemma 6. *If* Π_m *is ε-greedy and* $\mu_{\dagger,m} > \gamma_1 = \gamma_2 = 0$, *with probability at least* $1-1/T$, *it holds that* $\max_{H_m\in\mathcal{H}_m}\{\sum_{k\neq k_\dagger}\iota_{k,m}(T;H_m,\mathcal{B}_m,\Pi_m)\} = O(K\log(KT)/\mu^2_{\dagger,m})$.

Finally, the overall performance guarantees of TAL with ε-greedy clients are presented in the following theorem.

Theorem 3 (TAL with ε-greedy clients)**.** *For TAL with* $\gamma_1 = \gamma_2 = 0$, *if clients run ε-greedy and break ties uniformly at random, and* $\mu_{\dagger,m} \neq 0, \forall m \in [M]$, *it holds that*

$$R_F(T) = O\Big(\frac{K\Delta_{\max}\log(KMT)}{\Delta^2_{\min}} + \sum_{m\in[M]}\frac{K\Delta_{\max}\log(KMT)}{\mu^2_{\dagger,m}}\Big),$$

$$C_F(T) = O\Big(\sum_{m\in[M]}\Big[\frac{K\mu_{*,m}\log(KMT)}{M\Delta^2_{\min}} + \frac{K\mu_{*,m}\log(KMT)}{\mu^2_{\dagger,m}}\Big]\Big).$$

The two parts in regret and cost are from the learning and teaching phases, respectively. As typically $M \ll T$, the goal of having regret and cost both of $O(M\log(T))$ is also achieved.

VI. Conclusions

A novel idea of reward teaching was proposed to have the server guide autonomous clients in an unknown FMAB environment via reward adjustments, which avoids any previously required changes to the clients' protocols. A novel client-strategy-agnostic algorithm, TAL, was proposed. It was designed with two phases to separately encourage and discourage explorations. General performance analysis was established when the clients' strategies satisfy certain requirements. Especially, for the representative UCB1 and ε-greedy clients, rigorous analyses showed that TAL strikes a balance between regret and adjustment cost (logarithmic in both metrics), which is order-optimal w.r.t. the natural lower bound.

References

[1] C. Shi, W. Xiong, C. Shen, and J. Yang, "Reward teaching for federated multi-armed bandits," *arXiv preprint arXiv:2305.02441*, 2023.

[2] Z. Zhu, J. Zhu, J. Liu, and Y. Liu, "Federated bandit: A gossiping approach," *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 5, no. 1, pp. 1–29, 2021.

[3] C. Shi and C. Shen, "Federated multi-armed bandits," in *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*, February 2021.

[4] K. S. Reddy, P. Karthik, and V. Y. Tan, "Almost cost-free communication in federated best arm identification," *arXiv preprint arXiv:2208.09215*, 2022.

[5] R. Huang, W. Wu, J. Yang, and C. Shen, "Federated linear contextual bandits," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[6] A. Dubey and A. Pentland, "Differentially-private federated linear bandits," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[7] C. Li and H. Wang, "Asynchronous upper confidence bound algorithms for federated linear bandits," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2022, pp. 6529–6553.

[8] T. Li, L. Song, and C. Fragouli, "Federated recommendation system via differential privacy," in *2020 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2020, pp. 2592–2597.

[9] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, Fort Lauderdale, FL, USA, Apr. 2017, pp. 1273–1282.

[10] T. Lattimore and C. Szepesvári, *Bandit algorithms*. Cambridge University Press, 2020.

[11] T. Li and L. Song, "Privacy-preserving communication-efficient federated multi-armed bandits," *IEEE Journal on Selected Areas in Communications*, 2022.

[12] 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.

[13] S. Bubeck and N. Cesa-Bianchi, "Regret analysis of stochastic and nonstochastic multi-armed bandit problems," *Foundations & Trends in Machine Learning*, 2012.

[14] K.-S. Jun, L. Li, Y. Ma, and X. Zhu, "Adversarial attacks on stochastic bandits," in *Advances in Neural Information Processing Systems*, 2018.

[15] F. Liu and N. Shroff, "Data poisoning attacks on stochastic bandits," in *International Conference on Machine Learning*. PMLR, 2019, pp. 4042–4050.

[16] E. Garcelon, B. Roziere, L. Meunier, J. Tarbouriech, O. Teytaud, A. Lazaric, and M. Pirotta, "Adversarial attacks on linear contextual bandits," in *Advances in Neural Information Processing Systems*, 2020.

[17] H. Wang, H. Xu, and H. Wang, "When are linear stochastic bandits attackable?" in *International Conference on Machine Learning*. PMLR, 2022, pp. 23 254–23 273.

[18] A. Rangi, L. Tran-Thanh, H. Xu, and M. Franceschetti, "Saving stochastic bandits from poisoning attacks via limited data verification," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 7, 2022, pp. 8054–8061.

[19] S. Zuo, "Near optimal adversarial attack on UCB bandits," *arXiv preprint arXiv:2008.09312*, 2020.

[20] J.-Y. Audibert, S. Bubeck *et al.*, "Minimax policies for adversarial and stochastic bandits." in *COLT*, vol. 7, 2009, pp. 1–122.

[21] 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.

[22] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT Press, 1998.