Sequential Decision-Making in Intelligent Multi-Agent Systems

---

A

Dissertation

Presented to

the faculty of the School of Engineering and Applied Science

University of Virginia

---

in partial fulfillment

of the requirements for the degree

Doctor of Philosophy

by

Chengshuai Shi

May 2024

APPROVAL SHEET

This

Dissertation

is submitted in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy

Author: Chengshuai Shi

This Dissertation has been read and approved by the examing committee:

Advisor: Cong Shen

Advisor:

Committee Member: Nikolaos Sidiropoulos

Committee Member: Scott T. Acton

Committee Member: Hongning Wang

Committee Member: Tariq Iqbal

Committee Member: Haifeng Xu

Committee Member:

Accepted for the School of Engineering and Applied Science:

Jennifer L. West, School of Engineering and Applied Science

May 2024

Abstract

Sequential decision-making models, especially multi-armed bandits (MAB) and reinforcement learning (RL) have found tremendous success in wide applications of cognitive radios, recommender systems, healthcare, and beyond. However, the majority of these previous studies are focused on single-agent scenarios, which may fail to capture many modern real-world multi-agent applications (e.g., multiple devices sharing communication resources in cognitive radio). This thesis is thus motivated to extend previous single-agent decision-making studies to their multi-agent settings, which raises new challenges in system modeling, communication strategies, and beyond. In particular, this thesis focuses on two core topics in designing sequential decision-making algorithms for intelligent multi-agent systems: how to communicate and how to collaborate.

First, communication is one unique component in multi-agent systems compared with their single-agent counterparts. This thesis investigates this direction in providing efficient and robust information-sharing mechanisms. In particular, focusing on a decentralized multi-player MAB system, novel communication tools are developed, e.g., adaptive quantization, and error-correction coding. Besides communication, the collaboration strategy is also the key to enabling effective multi-agent systems. In this part, this thesis presents a line of works on federated MAB that extends the core principles of federated learning to MAB, and in particular, summarizes a modularized design principle for federated contextual bandits.

With these advances, this thesis deepens the understanding of decision-making designs in multi-agent systems and provides fundamental insights for future developments.

To my parents.

Acknowledgement

While writing this thesis, I feel more and more fortunate to receive invaluable support and help from my amazing research mentors, reliable co-authors, generous friends, and loving family. Without them, I certainly cannot have such a fruitful and enjoyable Ph.D. journey.

My first and deepest gratitude goes to my Ph.D. advisor, Professor Cong Shen. I met Prof. Shen during my undergraduate study at the University of Science and Technology of China, and he offered me the chance to continue to work with him at the University of Virginia. During this journey, his exceptional guidance, unwavering support, and invaluable mentorship have been instrumental in shaping my research and academic growth. I am truly grateful for his expertise, patience, and dedication, which have been pivotal in my success.

I am also indebted to my esteemed committee members, Professor Nikolaos D. Sidiropoulos, Professor Scott T. Acton, Professor Tariq Iqbal, Professor Hongning Wang, and Professor Haifeng Xu, for their valuable insights, constructive feedback, and contributions to the refinement of this dissertation. Their expertise and commitment to academic excellence have significantly enriched my research.

I am immensely grateful to my collaborators both within and outside of UVA, including Professor Jing Yang, Professor Jie Xu, Professor Tong Zhang, Professor Lixin Chen, Wei Xiong, Ruida Zhou, and Han Zhong. Their collaboration, intellectual exchange, and shared expertise have enriched my research, expanded its scope, and led to significant advancements. I am fortunate to have had the opportunity to work alongside such talented individuals. I would also like to acknowledge the collective support and inspiration provided by other members in Prof. Shen's group, namely, Li Fan, Yujia Mu, Kun Yang, Xizixiang Wei, Wei Shen, and Zihan Chen. Their intellectual discussions, encouragement, and friendship have enhanced my academic experience and made it truly meaningful. Moreover, I want to thank many other friends of mine, including but not limited to Shuman Sun, Zijiao Yang, Beichen Wang, and Wenjie Gao, who have provided me with generous help and support in both life and study.

I have also been very fortunate to be support by the Bloomberg Data Science Ph.D. Fellowship from 2021 to 2024. During two internships with the Bloomberg AI group, I have met many mentors and colleagues, including Anju Kambadur, Umut Topkara, Edgar Meij, Shubham Chopra, Diego Ceccarelli, Mozhgan

Azimpourkivi, and Hang Ren. Their support and guidance were the keys to my two successful and enjoyable internships.

Last but most importantly, my heartfelt appreciation goes to my family, especially my parents, Jianbing Shi and Shaomin Cheng, for their understanding, support, and love. Their belief in me and their sacrifices have been the foundation of my journey. I am grateful for the countless ways in which they have inspired and motivated me.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Sequential decision-making models, especially multi-armed bandits (MAB) and reinforcement learning (RL) have found wide applications in cognitive radios, recommender systems, healthcare, and beyond. Tremendous successes have also been witnessed in recent years. Comprehensive overviews of related basics and recent advances can be found in Lattimore and Szepesvári (2020); Sutton and Barto (2018).

However, the majority of previous MAB and RL studies are focused on single-agent scenarios. While being the fundamental setting to study, it may fail to capture many real-world applications, especially in the modern era where multi-agent systems commonly exist. In particular, in cognitive radio systems, there often exist multiple devices sharing communication resources, and in recommender systems, the online shopping platform typically serves hundreds and thousands of clients at the same time. Such applications motivate us to extend previous single-agent MAB/RL studies to their multi-agent settings. While single-agent studies provide many important insights, corresponding multi-agent designs remain challenging. In particular, two core questions are: how to communicate and how to collaborate.

This thesis is centered around these two core questions. It first discusses how to efficiently and robustly share information among agents in a decentralized multi-player MAB (MPMAB) system, where novel communication tools are developed. In particular, to achieve higher communication efficiency, an adaptive differential communication protocol is proposed, which contributes to closing a long-standing performance gap in the heterogeneous MPMAB problem. At the same time, to guarantee the robustness of communication, error-correction coding techniques are leveraged and nicely adapted to the MPMAB system and largely boost the performance to approach centralized ones.

Then, for collaboration designs, this thesis presents a line of works on federated MAB which extends the core principles of federated learning (FL) to MAB. Especially, this thesis covers collaboration designs in

different global-local relationships and varying generalization-personalization balance setups. Also, a general modulized approach is provided to flexibly involve FL protocols.

These advances, presented by this thesis, deepen the understanding of decision-making designs in multi-agent systems and provide fundamental insights for future developments. In particular, the proposed communication and collaboration mechanisms are broadly applicable beyond the specific problems.

Chapter 2

Information-sharing Designs for Multi-agent Decision Making

2.1 Decentralized Multi-player Multi-armed Bandits

Motivated by the application of cognitive radio (Anandkumar et al., 2010, 2011; Gai et al., 2010), the multi-player version of the multi-armed bandits problem (MP-MAB) has sparked significant interest in recent years. MP-MAB takes player interactions into account by having multiple decentralized players simultaneously play the bandit game and interact with each other through arm collisions.

2.1.1 Problem Formulation

A decentralized MP-MAB model consists of $K \in \mathbb{N}$ arms and $M \in \mathbb{N}$ players. As commonly assumed in Bistritz and Leshem (2020); Boursier et al. (2020), there are more arms than players, i.e., $M \leq K$, and initially the players have knowledge of K but not M. Furthermore, no *explicit* communications are allowed among players, which results in a decentralized system. Also, time is assumed to be slotted, and at time step t, each player $m \in [M]$ chooses and pulls an arm $s_m(t) \in [K]$. The action vector of all players at time t is denoted as $S(t) := [s_1(t), ..., s_M(t)]$, which is referred to as a "matching" for convenience.

Individual Outcomes

For each player m, an outcome $O_{k,m}(t)$ is associated with her action of pulling arm $s_m(t) = k$ at time t, which is defined as

$$O_{k,m}(t) := X_{k,m}(t) \cdot \eta_k(S(t)). \tag{2.1}$$

In Eqn. (2.1), $X_{k,m}(t)$ is a random variable of arm utility and $\eta_k(S(t))$ is the no-collision indicator defined by $\eta_k(S) := \mathbb{1}\{|C_k(S)| \leq 1\}$ with $C_k(S) := \{n \in [M] | s_n = k\}$. In other words, if player m is the only player choosing arm k, the outcome is $X_{k,m}(t)$; if multiple players choose arm k simultaneously, a collision happens on this arm and the outcome is zero regardless of $X_{k,m}(t)$.

For a certain arm-player pair, i.e., (k, m), the set of random arm utilities $\{X_{k,m}(t)\}_{t\geq 1}$ is assumed to be independently sampled from an unknown distribution $\phi_{k,m}$, which has a bounded support on $[0, 1]$ and an unknown expectation $\mathbb{E}[X_{k,m}(t)] = \mu_{k,m}$. The homogeneous and heterogeneous settings are specified in the following:

- **Homogeneous setting**: the expected utility of each arm is the same for all players, i.e., $\mu_{k,m} = \mu_k, \forall m \in [M], \forall k \in [M]$;
- **Heterogeneous setting**: the players may have different expected utilities of each arm, i.e., potentially, $\mu_{k,m} \neq \mu_{k,n}$ when $m \neq n$.

To ease the exposition, we define $\mathcal{S} = \{S = [s_1, ..., s_M] | s_m \in [K], \forall m \in [M]\}$ as the set of all possible matchings S and abbreviate the arm k of player m as arm (k, m). We further denote $\boldsymbol{\mu} = [\mu_{k,m}]_{(k,m)\in[K]\times[M]}$ and $\boldsymbol{\mu}_S = [\mu_{s_m,m}]_{m\in[M]}$ for $S = [s_1, ..., s_M]$.

System Rewards

Besides players' individual outcomes, with matching $S(t)$ chosen at time t, a random system reward, denoted as $V(S(t), t)$, is collected for the entire system. The most commonly-studied reward function (Bistritz and Leshem, 2020; Boursier et al., 2020) is the sum of outcomes from different players (referred to as the *linear reward function*), i.e.,

$$V(S(t), t) := \sum\nolimits_{m\in[M]} O_{s_m(t),m}(t).$$

With this linear reward function, for matching S, the expected system reward is denoted as $V_{\boldsymbol{\mu},S} := \mathbb{E}[V(S, t)] = \sum_{m\in[M]} \mu_{s_m,m}\eta_{s_m}(S)$ under matrix $\boldsymbol{\mu}$. As almost all the existing MP-MAB literature focus on the linear reward function, we also focus on this case first, but note that the problem formulation presented in this section can be extended to general (nonlinear) reward functions.

Feedback Model

Different feedback models exist in the MP-MAB literature. Especially, we consider the collision-sensing and no-sensing models specified in the following:

- **Collision-sensing model**: player m can access her own outcome $O_{s_m(t),m}(t)$ and the corresponding no-collision indicator $\eta_{s_m(t)}(S(t))$;
- **No-sensing model**: player m can access her own outcome $O_{s_m(t),m}(t)$ but not the corresponding no-collision indicator $\eta_{s_m(t)}(S(t))$.

Note that in both cases, neither the overall reward $V(S(t),t)$ nor the outcomes of other players can be observed by each player. In other words, at time t, player m chooses arm $s_m(t)$ based on her own history

$$H_m(t) = \left\{s_m(\tau), O_{s_m(\tau),m}(\tau), \eta_{s_m(\tau)}(S(\tau))\right\}_{1\le\tau\le t-1} \qquad \text{(Collision-sensing)}$$

or

$$H_m(t) = \left\{s_m(\tau), O_{s_m(\tau),m}(\tau)\right\}_{1\le\tau\le t-1} \qquad \text{(No-sensing)}.$$

Regret Definition

If $\boldsymbol{\mu}$ is known *a priori*, the optimal choice is the matching that gives the highest expected reward $V_{\boldsymbol{\mu},*} := \max_{S\in\mathcal{S}} V_{\boldsymbol{\mu},S}$. We formally define the regret after T rounds of playing as

$$R(T) = TV_{\boldsymbol{\mu},*} - \mathbb{E}\left[\sum_{t=1}^{T} V(S(t),t)\right], \tag{2.2}$$

where the expectation is w.r.t. the randomness of the policy and the environment.

2.1.2 Related Works

The MP-MAB setting were originally motivated from the application of cognitive radio and can be dated back to Anandkumar et al. (2010, 2011); Gai et al. (2010). Since being proposed, most studies considered the homogeneous collision-sensing setting (Liu and Zhao, 2010; Avner and Mannor, 2014; Rosenski et al., 2016; Besson and Kaufmann, 2018). With implicit communications, Boursier and Perchet (2019); Wang et al. (2020a) proved regrets that approach the centralized ones; thus, the homogeneous collision-sensing variant was fairly well understood.

The Heterogeneous Variant

Compared with the homogeneous variant, the heterogeneous MP-MAB problems (Kalathil et al., 2014; Nayyar et al., 2016) with player-dependent arm utilities, on the other hand, was less investigated. Some attempts includes Bistritz and Leshem (2020); Magesh and Veeravalli (2019); Tibrewal et al. (2019); Boursier et al.

(2020), whose regrets are far from the (natural) centralized lower bound as discussed later. Our work Shi et al. (2021) filled this understanding gap with the BEACON algorithm proposed, whose regret, for the first time, is capable of approaching the lower bound.

The No-sensing Variant

On the no-sensing model, there were also limited progress before our work Shi et al. (2020). In particular, Lugosi and Mehrabian (2018); Boursier and Perchet (2019) touches upon the this setting. However, unlike our work Shi et al. (2020), their proposed designs are incapable of approaching the centralized performance.

Other variants

Many other variants beyond the basic MP-MAB settings have also been investigated. First, while all the aforementioned works are confined to the linear reward function, some attempts have been made to consider other reward functions. For example, a fairness measurement was considered in Bistritz et al. (2020), while "stable" allocations were investigated in Avner and Mannor (2016); Darak and Hanawal (2019). Our work (Shi et al., 2021), instead, provided a general consideration towards this direction.

Secondly, the adversarial, instead of stochastic, rewards were studied in Alatur et al. (2020); Bubeck et al. (2020). Our work (Shi and Shen, 2021b) makes additional contributions on this direction in understanding how to perform robust communications in adversarial environments.

2.2 Heterogeneous Collision-sensing Model: Adaptive Differential Communication

We first consider the collision-sensing setting in a heterogeneous MP-MAB model, which is defined in Section 2.1.1 and summarized in the following:

- **Heterogeneous setting**: potentially $\mu_{k,m} \neq \mu_{k,n}$ when $m \neq n$;
- **Collision-sensing model**: player m can access her own outcome $O_{s_m(t),m}(t)$ and the corresponding no-collision indicator $\eta_{s_m(t)}(S(t))$;

For this setting, we propose the BEACON – *Batched Exploration with Adaptive COmmunicatioN* algorithm in Shi et al. (2021), whose design, analysis and evaluation are provided in the following subsections.

2.2.1 The BEACON Algorithm

Algorithm Structure and Key Ideas

The BEACON algorithm starts with the orthogonalization procedure proposed in Wang et al. (2020a) at the beginning of the game, during which each player individually estimates the number of players M and assigns herself of a unique index $m \in [M]$. Then, BEACON proceeds in epochs and each epoch consists of two phases: (implicit) communication and exploration. While similar two-phase structures have been adopted by other heterogeneous MP-MAB algorithms (Tibrewal et al., 2019; Boursier et al., 2020), those designs fail to have regrets approaching the centralized lower bound.

The challenge in approaching the centralized lower bound is not only designing more efficient implicit communications and explorations, but also connecting them in a way that neither phase dominates the overall regret and both approach the centralized lower bound simultaneously. BEACON precisely achieves these goals, with several key ideas that not only are crucial to closing the regret gap but also hold individual values in MP-MAB research. First, a novel adaptive differential communication (ADC) method is proposed, which is fundamental in improving the effectiveness and efficiency of implicit communications. Specifically, ADC drastically reduces the communication cost from up to $O(\log(T))$ per epoch in state-of-the-art designs (Boursier et al., 2020) to $O(1)$ per epoch, which ensures a low communication cost. Second, CUCB principles (Chen et al., 2013) are incorporated with a batched exploration structure to ensure a low exploration loss. CUCB principles address a critical challenge of *large amount of matchings* in heterogeneous MP-MAB (i.e., $|\mathcal{S}| = K^M$), which hampered prior designs. The batched structure, on the other hand, is carefully embedded and optimized such that the need of communication and exploration is balanced, leading to neither dominating the overall regret.

Batched Exploration

To facilitate the illustration, we first present the batched exploration scheme and also a sketch of BEACON under an imaginary communication-enabled setting. Specifically, players are assumed to be able to communicate with each other freely in this subsection.

The batched exploration proceeds as follows. At the beginning of epoch r, each player m maintains an arm counters $p^r_{k,m}$ for each arm k of hers. The counters are updated as $p^r_{k,m} = \lfloor \log_2(T^r_{k,m}) \rfloor$, where $T^r_{k,m}$ is the number of exploration pulls on arm (k, m) up to epoch r. Then, the leader (referring to the player with index 1) collects arm statistics from followers (referring to the players other than the leader). Specifically, if $p^r_{k,m} > p^{r-1}_{k,m}$, statistics $\tilde{\mu}^r_{k,m}$ is collected from follower m; otherwise, $\tilde{\mu}^r_{k,m}$ is not updated and kept the same as $\tilde{\mu}^{r-1}_{k,m}$, where $\tilde{\mu}^r_{k,m}$ is a to-be-specified characterization of arm (k, m)'s sample mean $\hat{\mu}^r_{k,m}$. With the updated

information, an upper confidence bound (UCB) matrix $\bar{\boldsymbol{\mu}}_r = [\bar{\mu}^r_{k,m}]_{(k,m)\in[K]\times[M]}$ is calculated by the leader, where

$$\bar{\mu}^r_{k,m} = \tilde{\mu}^r_{k,m} + \sqrt{3\ln t_r / 2^{p^r_{k,m}+1}},$$

and t_r is the time step at the beginning of epoch r.

The UCB matrix $\bar{\boldsymbol{\mu}}_r$ is then fed into a combinatorial optimization solver, denoted as $\texttt{Oracle}(\cdot)$, which outputs the optimal matching w.r.t. the input. Specifically,

$$S_r = [s^r_1, ..., s^r_M] \leftarrow \texttt{Oracle}(\bar{\boldsymbol{\mu}}_r) = \arg\max_{S\in\mathcal{S}} \left\{ \sum_{s_m\in S} \bar{\mu}^r_{s_m,m} \right\},$$

which can be computed with a polynomial time complexity using the Hungarian algorithm (Munkres, 1957). We note that similar optimization solvers are also required by Boursier et al. (2020); Tibrewal et al. (2019). Inspired by the exploration choice of CUCB, this matching S_r is chosen to be explored. The leader thus assigns the matching S_r to followers, i.e., arm s^r_m for player m.

After the assignment, the exploration begins. One important ingredient of BEACON is that the duration of exploring the chosen matching, i.e., the adopted batch size, is determined by the smallest arm counter in it. Specifically, for S_r, we denote $p_r = \min_{m\in[M]} p^r_{s^r_m,m}$ and the batch size is chosen to be 2^{p_r}. In other words, during the following 2^{p_r} time steps, players are fixated to exploring the matching S_r. Then, epoch $r+1$ starts, and the same procedures are iterated.

Remark 2.2.1. BEACON directly selects the matching with the largest UCB to explore. It turns out that this natural method significantly outperforms the "matching-elimination" scheme in Boursier et al. (2020), and is critical to achieving a near-optimal exploration loss. In addition, the chosen batch size of 2^{p_r} ensures *sufficient but not excessive* pulls w.r.t. the least pulled arm(s) in the chosen matching, which dominate the uncertainties. Furthermore, while similar batched structures have been utilized in the bandit literature (Auer et al., 2002; Hillel et al., 2013), the updating of arm counters in BEACON is carefully tailored. Last, the leader collects followers' statistics only when arm counters increase, i.e., $p^r_{k,m} > p^{r-1}_{k,m}$, which means $\tilde{\mu}^r_{k,m}$ is sufficiently more precise than $\tilde{\mu}^{r-1}_{k,m}$. This design contributes to a low communication frequency while not affecting the exploration efficiency.

Efficient Implicit Communication

Since explicit communication is prohibited in decentralized MP-MAB problems, we now discuss how to use *implicit* communication (Boursier and Perchet, 2019) to share information in BEACON. Specifically, players can take predetermined turns to "communicate" by having the "receive" player sample one arm and the

"send" player either pull (create collision; bit 1) or not pull (create no collision; bit 0) the same arm to transmit one-bit information. Although information sharing is enabled, such a forced-collision communication approach is inevitably costly, as collisions reduce the rewards. The challenge now is how to keep the communication loss small, ideally $O(\log(T))$.

Algorithm 1 BEACON: Leader

Initialization: $r \leftarrow 0$; $\forall (k,m), p^r_{k,m} \leftarrow -1, T^r_{k,m} \leftarrow 0, \tilde{\mu}^r_{k,m} \leftarrow 0$
Play each arm $k \in [K]$ and $T^{r+1}_{k,1} \leftarrow T^r_{k,1} + 1$
while not reaching the time horizon **do**
  $r \leftarrow r + 1$
  $\forall (k,m), p^r_{k,m} \leftarrow \left\lfloor \log_2(T^r_{k,m}) \right\rfloor$
  $\forall k \in [K]$, update sample mean $\hat{\mu}^r_{k,1}$ with the first $2^{p^r_{k,1}}$ exploratory samples from arm k
▷ *Communication Phase*
  for $(k,m) \in [K] \times [M]$ **do**
    if $p^r_{k,m} > p^{r-1}_{k,m}$ **then**
      $\tilde{\delta}^r_{k,m} \leftarrow \texttt{Receive}(\tilde{\delta}^r_{k,m}, m)$
      $\tilde{\mu}^r_{k,m} \leftarrow \tilde{\mu}^{r-1}_{k,m} + \tilde{\delta}^r_{k,m}$
    else
      $\tilde{\mu}^r_{k,m} \leftarrow \tilde{\mu}^{r-1}_{k,m}$
    end if
  end for
  $\forall (k,m), \bar{\mu}^r_{k,m} \leftarrow \tilde{\mu}^r_{k,m} + \sqrt{3 \ln t_r / 2^{p^r_{k,m}+1}}$
  $S_r = [s^r_1, ..., s^r_M] \leftarrow \texttt{Oracle}(\bar{\boldsymbol{\mu}}_{\boldsymbol{r}})$
  $\forall m \in [M], \texttt{Send}(s^r_m, m)$
▷ *Exploration Phase*
  $p_r \leftarrow \min_{m \in [M]} p^r_{s^r_m, m}$
  Play arm s^r_1 for 2^{p_r} times
  Signal followers to stop exploration
  Update $\forall m \in [M], T^{r+1}_{s_m,m} \leftarrow T^r_{s_m,m} + 2^{p_r}$
end while

The *batched* exploration scheme plays a key role in reducing the communication loss via infrequent information updating. In other words, players only communicate statistics and decisions before each batch instead of each time step. With the aforementioned batch size, there are at most $O(\log(T))$ epochs in horizon T. Thus, intuitively, if the communication loss per epoch can be controlled of order $O(1)$ irrelevant of T, the overall communication loss would not be dominating. However, this requirement is challenging and none of the existing implicit communication schemes (Boursier and Perchet, 2019; Boursier et al., 2020) can meet it, which calls for a novel communication design.

From the discussion of the exploration phases, we can see that sharing arm statistics $\tilde{\mu}^r_{k,m}$ is the most challenging part. Specifically, as opposed to sharing integers of arm indices in S_r and the batch size parameter p_r, statistics $\tilde{\mu}^r_{k,m}$ is often a decimal while forced-collision is fundamentally a digital communication protocol. We thus focus on the communication design for sharing statistics $\tilde{\mu}^r_{k,m}$, and propose the adaptive differential

communication (ADC) method as detailed below. Details of sharing S_r and p_r can be found in supplementary material.

The first important idea is to let followers **adaptively** quantize sample means for communication. Specifically, upon communication, the arm statistics $\tilde{\mu}^r_{k,m}$ is not directly set as the collected sample mean $\hat{\mu}^r_{k,m}$. Instead, $\tilde{\mu}^r_{k,m}$ is a quantized version of $\hat{\mu}^r_{k,m}$ using $\lceil 1 + p^r_{k,m}/2 \rceil$ bits. Since $\tilde{\mu}^r_{k,m}$ is communicated only upon an increase of the arm counter $p^r_{k,m}$, this quantization length is adaptive to the arm counter (or equivalently the arm pulls), and further to the adopted confidence bound, i.e., $\sqrt{3 \ln t_r / 2^{p^r_{k,m}+1}}$. However, this idea alone is not sufficient because $p^r_{k,m}$ is of order up to $O(\log(T))$, instead of $O(1)$.

To overcome this obstacle, the second key idea is **differential** communication, which significantly reduces the redundancies in statistics sharing. Specifically, follower m first computes the difference

$$\tilde{\delta}^r_{k,m} = \tilde{\mu}^r_{k,m} - \tilde{\mu}^{r-1}_{k,m},$$

and then truncates the bit string of $\tilde{\delta}^r_{k,m}$ upon the most significant non-zero bit, e.g., 110 for 000110. She only communicates this truncated version of $\tilde{\delta}^r_{k,m}$ in the transmission of $\tilde{\mu}^r_{k,m}$ to the leader. The intuition is that $\tilde{\mu}^r_{k,m}$ and $\tilde{\mu}^{r-1}_{k,m}$ are both concentrated at $\mu_{k,m}$ with high probabilities, which results in a small $\tilde{\delta}^r_{k,m}$. From an information-theoretic perspective, the conditional entropy of $\tilde{\mu}^r_{k,m}$ on $\tilde{\mu}^{r-1}_{k,m}$, i.e., $H(\tilde{\mu}^r_{k,m} | \tilde{\mu}^{r-1}_{k,m})$, is often small because they are highly correlated.[1]

As will be clear in the regret analysis, putting these two ideas together results in an effective communication design, i.e, the ADC scheme, whose expected regret is of order $O(1)$ per epoch and $O(\log(T))$ overall. This method itself represents an important improvement over prior implicit communication protocols in MP-MAB, whose loss is typically of order $O(\log(T))$ per epoch and $O(\log^2(T))$ in total with multiple optimal matchings (Boursier and Perchet, 2019; Boursier et al., 2020). Techniques similar to ADC have been utilized in areas outside of MAB, e.g., wireless communications (Goldsmith and Chua, 1998), with proven success in practice (Goldsmith, 2005).

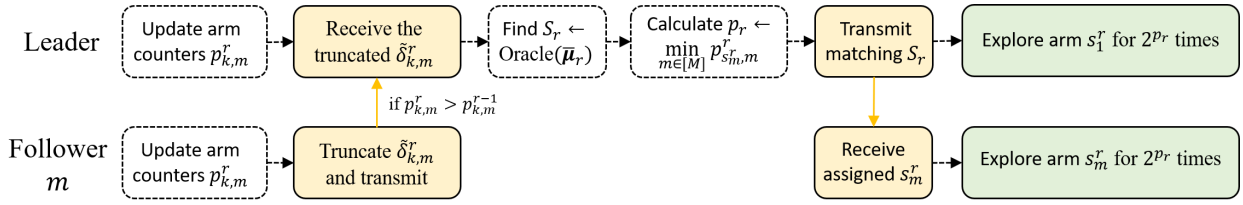


Figure 2.1: A sketch of epoch r in BEACON. Yellow boxes and yellow lines indicate communications, green boxes for explorations, and boxes with dotted frame for computations.

[1] Note that sharing the truncated version of $\tilde{\delta}^r_{k,m}$ results in another difficulty that its length varies for different player-arm pairs and is unknown to the leader. A specially crafted "signal-then-communicate" scheme is designed to tackle this challenge.

The complete BEACON algorithm can now be obtained by plugging ADC into the batched exploration structure. A sketch of one BEACON epoch is illustrated in Fig. 2.1, and the leader's algorithm is presented in Algorithm 1. The follower's algorithm can be found in the supplementary material, along with the definitions of the implicit communication protocols denoted by functions `Send()` and `Receive()`. Note that the for-loops with (k, m) and $\forall(k, m)$ in the pseudo-codes indicate the iteration over all possible arm-player pairs of $[K] \times [M]$. In addition, the communications of the leader to herself indicated by the pseudo-codes denote her own calculations instead of real forced-collision communications (among the leader and followers), which is a simplification for better exposition.

2.2.2 Regret Analysis

With notations

$$\mathcal{S}_c := \{S \in \mathcal{S} | \exists m \neq n, s_m = s_n\}$$

as the set of collided matchings;

$$\mathcal{S}_* := \{S \in \mathcal{S} | V_{\boldsymbol{\mu},S} = V_{\boldsymbol{\mu},*}\}$$

as the set of optimal matchings;

$$\mathcal{S}_b = \mathcal{S} \backslash (\mathcal{S}_* \cup \mathcal{S}_c)$$

as the set of collision-free suboptimal matchings;

$$\Delta_{\min}^{k,m} := V_{\boldsymbol{\mu},*} - \max\{V_{\boldsymbol{\mu},S} | S \in \mathcal{S}_b, s_m = k\}$$

as the minimum sub-optimality gap for collision-free matchings containing arm-player pair (k, m);

$$\Delta_{\min} := \min_{(k,m)}\{\Delta_{\min}^{k,m}\}$$

as the minimum sub-optimality gap for all collision-free matchings, the regret of BEACON with the linear reward function is analyzed in the following theorem.

Theorem 2.2.2. *With the linear reward function, the regret of BEACON is upper bounded as*

$$R_{\text{linear}}(T) = \tilde{O}\left(\sum_{(k,m)\in[K]\times[M]} \frac{M \log(T)}{\Delta_{\min}^{k,m}} + M^2 K \log(T)\right) = \tilde{O}\left(\frac{M^2 K \log(T)}{\Delta_{\min}}\right).$$

Note that in Eqn. (2.3), the first term represents the exploration regret of BEACON, and the second term the communication regret. Compared with the state-of-the-art regret result $\tilde{O}(M^3K\log(T)/\Delta_{\min})$ for METC (Boursier et al., 2020), the regret bound in Theorem 2.2.2 improves the dependence of M from M^3 to M^2. It turns out that this quadratic dependence is optimal because the same dependence exists in the centralized lower bound (hence a natural lower bound for decentralized MP-MAB) for the linear reward function, as from Kveton et al. (2015):[2]

$$R_{\text{linear}}(T) = \Omega\left(\frac{M^2K}{\Delta_{\min}}\log(T)\right). \tag{2.3}$$

By comparing Theorem 2.2.2 and Eqn. (2.3), it can be observed that with the linear reward function, BEACON achieves a regret that approaches the centralized lower bound. The efficiency and effectiveness of both exploration and communication phases are critical in this achievement, as we can see that both terms in Theorem 2.2.2 are non-dominating at $\tilde{O}(M^2K\log(T))$.

In addition to the problem-dependent bound given in Theorem 2.2.2, the following theorem establishes a problem-independent bound, which can be thought of as a worst-case characterization.

Theorem 2.2.3. *With the linear reward function, it holds that*

$$R_{\text{linear}}(T) = O\left(M\sqrt{KT\log(T)}\right).$$

Theorem 2.2.3 not only improves the best known problem-independent bound $O(M^{\frac{3}{2}}\sqrt{KT\log(T)})$ (Boursier et al., 2020) in the decentralized MP-MAB literature, but also approaches the centralized lower bound $\Omega(M\sqrt{KT})$ (Kveton et al., 2015; Merlis and Mannor, 2020) up to logarithmic factors.

Theorems 2.2.2 and 2.2.3 demonstrate that for the linear reward function, BEACON closes the performance gap (both problem-dependent and problem-independent) between decentralized heterogeneous MP-MAB algorithms and their centralized counterparts. The regret bounds of various MP-MAB algorithms, including BEACON, are summarized in Table 2.1.

Remark 2.2.4. We note that it is also feasible to combine the ADC protocol and METC (Boursier et al., 2020), which can address its communication inefficiency, especially with multiple optimal matchings. However, with ideas from CUCB, BEACON is much more efficient in exploration than "Explore-then-Commit"-type of algorithms (e.g., METC), which is the main reason we did not fully elaborate the combination of METC and ADC in this work. Theoretically, this superiority can be reflected in the extra multiplicative factor in the exploration loss of METC shown in Table 2.1.

[2]This lower bound holds for the cases with arbitrarily correlated arms, as considered in this work. Under additional arm independence assumptions (Combes et al., 2015), lower regrets can be achieved.

Table 2.1: Regret Bounds of Decentralized MP-MAB Algorithms

Algorithm/Reference	Reward function	Assumptions			Regret
		Known horizon T	Known gap $\Delta_{\min}$	Unique optimal matching	
GoT † (Bistritz and Leshem, 2020)	Linear	No	Yes	Yes	$O\left(M\log^{1+\kappa}(T)\right)$
Decentralized MUMAB (Magesh and Veeravalli, 2019)	Linear	No	Yes	No	$O\left(K^3\log(T)\right)$
ESE1 (Tibrewal et al., 2019)	Linear	No	No	Yes	$O\left(\frac{M^2K}{\Delta_{\min}^2}\log(T)\right)$
METC (Boursier et al., 2020)	Linear	Yes	No	Yes	$O\left(\frac{M^3K}{\Delta_{\min}}\log(T)\right)$
METC (Boursier et al., 2020)	Linear	Yes	No	No	$O\left(MK\left(\frac{M^2\log(T)}{\Delta_{\min}}\right)^{1+\iota}\right)$
BEACON (this work, Theorem 2.2.8)	General	No	No	No	$\tilde{O}\left(\frac{MK\Delta_{\max}}{(f^{-1}(\Delta_{\min}))^2}\log(T)\right)$
BEACON (this work, Theorem 2.2.2)	Linear	No	No	No	$\tilde{O}\left(\frac{M^2K}{\Delta_{\min}}\log(T)\right)$
Lower bound (Kveton et al., 2015)	Linear	N/A	N/A	N/A	$\Omega\left(\frac{M^2K}{\Delta_{\min}}\log(T)\right)$

†: tuning parameters in GoT requires knowledge of arm utilities;
κ, ι: arbitrarily small non-zero constants.

2.2.3 Beyond Linear Reward Functions

General Reward Functions

In this section, we move away from the linear reward functions in almost all prior MP-MAB research, and extend the study to general (nonlinear) reward functions. Two exemplary nonlinear reward functions are given below, with more examples provided in the supplementary material.

- Proportional fairness: $V(S,t) = \sum_{m\in[M]} \omega_m \ln(\epsilon + O_{s_m,m}(t))$, where $\epsilon > 0$ and $\omega_m > 0$ are constants. It promotes fairness among players (Mo and Walrand, 2000);
- Minimal: $V(S,t) = \min_{m\in[M]}\{O_{s_m,m}(t)\}$, which indicates the system reward is determined by the least-rewarded player, i.e., the short board of the system;

These reward functions all hold their value in real-world applications, but are largely ignored and cannot be effectively solved by previous approaches. The difficulty introduced by this extension not only lies in the complex mapping from the (unreliable) individual outcomes to system rewards, but also comes from the potential "coupling" effect among players (e.g., the minimal reward function).

To better characterize the problem, the following mild assumptions are considered.

Assumption 2.2.5. *There exists an expected reward function $v(\cdot)$ such that $V_{\boldsymbol{\mu},S} := \mathbb{E}[V(S,t)] = v(\boldsymbol{\mu}_S \odot \boldsymbol{\eta}_S)$, where $\boldsymbol{\eta}_S := [\eta_{s_m}(S)]_{m\in[M]}$ and $\boldsymbol{\mu}_S \odot \boldsymbol{\eta}_S := [\mu_{s_m,m}\eta_{s_m}(S)]_{m\in[M]}$.*

Assumption 2.2.6 (Monotonicity)**.** *The expected reward function is monotonically non-decreasing with respect to the vector $\boldsymbol{\Lambda} = \boldsymbol{\mu}_S \odot \boldsymbol{\eta}_S$, i.e., if $\boldsymbol{\Lambda} \preceq \boldsymbol{\Lambda}'$, we have $v(\boldsymbol{\Lambda}) \le v(\boldsymbol{\Lambda}')$.*

Assumption 2.2.7 (Bounded smoothness)**.** *There exists a strictly increasing (and thus invertible) function $f(\cdot)$ such that $\forall \boldsymbol{\Lambda}, \boldsymbol{\Lambda}', |v(\boldsymbol{\Lambda}) - v(\boldsymbol{\Lambda}')| \le f(\|\boldsymbol{\Lambda} - \boldsymbol{\Lambda}'\|_\infty)$.*

Assumption 2.2.5 indicates that the expected reward $V_{\boldsymbol{\mu},S}$ of matching S is determined only by its expected individual outcomes. It is true for the linear reward function, and also generally holds if distributions $\{\phi_{k,m}\}$ are mutually independent and determined by their expectations $\{\mu_{k,m}\}$, e.g., Bernoulli distribution. Assumptions 2.2.6 and 2.2.7 concern the monotonicity and smoothness of the expected reward function, which are natural for most practical reward functions, including the above examples. Similar assumptions have been adopted by Chen et al. (2013, 2016b); Wang and Chen (2018).

BEACON Adaption and Performance Analysis

In Section 2.2.2, a combinatorial optimization solver $\texttt{Oracle}(\cdot)$ is implemented for the linear reward function. With ideas from CUCB (Chen et al., 2013), BEACON can be extended to handle a general reward function with a corresponding solver $\texttt{Oracle}(\cdot)$ that outputs the optimal (non-collision) matching w.r.t. the input matrix $\boldsymbol{\mu}'$, i.e., $S' \leftarrow \texttt{Oracle}(\boldsymbol{\mu}') = \arg\max_{S\in\mathcal{S}\setminus\mathcal{S}_c} V_{\boldsymbol{\mu}',S}$.

With such an oracle, the following theorem provides performance guarantees of BEACON.

Theorem 2.2.8 (**General reward function**)**.** *Under Assumptions 2.2.5, 2.2.6, and 2.2.7, denoting $\Delta_{\max}^{k,m} := V_{\boldsymbol{\mu},*} - \min\{V_{\boldsymbol{\mu},S} | S \in \mathcal{S}_b, s_m = k\}$ and $\Delta_c := f(1)$, the regret of BEACON is upper bounded as*

$$\begin{aligned} R(T) &= \tilde{O}\left(\sum_{(k,m)\in[K]\times[M]} \left[\frac{\Delta_{\min}^{k,m}}{(f^{-1}(\Delta_{\min}^{k,m}))^2} + \int_{\Delta_{\min}^{k,m}}^{\Delta_{\max}^{k,m}} \frac{1}{(f^{-1}(x))^2}\mathrm{d}x\right]\log(T) + M^2K\Delta_c\log(T)\right) \\ &= \tilde{O}\left(\sum_{(k,m)\in[K]\times[M]} \frac{\Delta_{\max}^{k,m}\log(T)}{(f^{-1}(\Delta_{\min}^{k,m}))^2} + M^2K\Delta_c\log(T)\right). \end{aligned}$$

With a stronger smoothness assumption, we can obtain a clearer exposition of the regret.

Corollary 2.2.9. *Under Assumptions 2.2.5 and 2.2.6, if there exists $B > 0$ such that $\forall \boldsymbol{\Lambda}, \boldsymbol{\Lambda}', |v(\boldsymbol{\Lambda}) - v(\boldsymbol{\Lambda}')| \le B\|\boldsymbol{\Lambda} - \boldsymbol{\Lambda}'\|_\infty$, it holds that*

$$R(T) = \tilde{O}\left(\sum_{(k,m)\in[K]\times[M]} \frac{B^2}{\Delta_{\min}^{k,m}}\log(T) + M^2KB\log(T)\right).$$

In addition, since the combinatorial optimization problems with general reward functions can be NP-hard, it is more practical to adopt approximate solvers rather than the exact ones (Vazirani, 2013). To accommodate such needs, we introduce the following definition of (α, β)-approximation oracle for $\alpha, \beta \in [0, 1]$ as in Chen et al. (2013, 2016a,b); Wang and Chen (2017):

Definition 2.2.10. *With a matrix $\boldsymbol{\mu}' = [\mu'_{k,m}]_{(k,m)\in[K]\times[M]}$ as input, an (α, β)-approximation oracle outputs a matching S', such that $\mathbb{P}[V_{\boldsymbol{\mu}',S'} \geq \alpha \cdot V_{\boldsymbol{\mu}',*}] \geq \beta$, where $V_{\boldsymbol{\mu}',*} = \max_{S\in\mathcal{S}} V_{\boldsymbol{\mu}',S}$.*

With only an approximate solver, it is no longer fair to compare the performance against the optimal reward. Instead, as in the CMAB literature (Chen et al., 2013, 2016a,b; Wang and Chen, 2017), an (α, β)-approximation regret is considered: $R_{\alpha,\beta}(T) = T\alpha\beta V_{\boldsymbol{\mu},*} - \mathbb{E}[\sum_{t=1}^{T} V(S(t), t)]$, where the performance is compared to the $\alpha\beta$ fraction of the optimal reward. As shown in the supplementary material, for this (α, β)-approximation regret, an upper bound similar to Theorem 2.2.8 can be obtained.

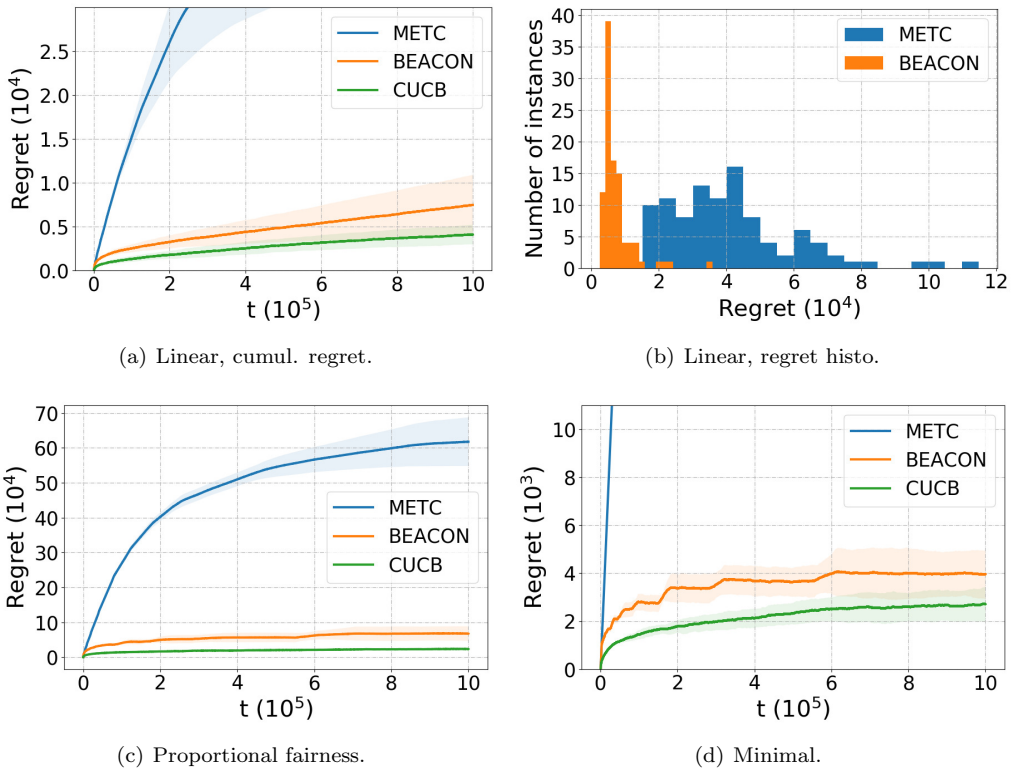
2.2.4 Experimental Results

In this section, BEACON is empirically evaluated with both linear and general (nonlinear) reward functions. All results are averaged over 100 experiments and the utilities follow mutually independent Bernoulli distributions. Additional experimental details, empirical algorithm enhancements and more experimental results (e.g., with a large game), can be found in the supplementary material.

Linear Reward Function.

BEACON is evaluated along with the centralized CUCB (Chen et al., 2013) and the state-of-the-art decentralized algorithm METC (Boursier et al., 2020). The decentralized GoT algorithm (Bistritz and Leshem, 2020) is also evaluated but its regrets are over 100× larger than those of BEACON, and thus is omitted in the plots. Fig. 2.2 reports results under the same instance in Boursier et al. (2020) with $K = 5, M = 5$. Although this is a relatively hard instance with multiple optimal matchings and small sub-optimality gaps, BEACON still achieves a comparable performance as CUCB, and significantly outperforms METC: an approximate 7× regret reduction at the horizon.

To validate whether this significant gain of BEACON over METC is representative, we plot in Fig. 2.2(b) the histogram of regrets with 100 randomly generated instances still with $M = 5, K = 5, T = 10^6$. Expected arm utilities are uniformly sampled from $[0, 1]$ in each instance. It can be observed that the gain of BEACON is very robust – its average regret is approximately 6× lower than METC.



(a) Linear, cumul. regret.

(b) Linear, regret histo.

(c) Proportional fairness.

(d) Minimal.

Figure 2.2: Regret comparisons between BEACON and other MP-MAB algorithms. The continuous curves represent the empirical average values, and the shadowed areas represent the standard deviations. (a), (c) and (d) are evaluated with specific game instances, and (b) is the regret histogram of 100 randomly generated instances.

General Reward Function.

Two representative nonlinear reward functions are used to evaluate BEACON: (1) the proportional fairness function with $\forall m \in [M], \omega_m = 1, \epsilon = 10^{-2}$; (2) the minimal function. BEACON is compared with CUCB and METC.[3] Under a game instance with $M = 6, K = 8$, Fig. 2.2(c) reports the regrets under the proportional fairness function, and Fig. 2.2(d) with the minimal function. From both results, it can be observed that BEACON has slightly larger (but comparable) regrets than the centralized CUCB, while significantly outperforming METC.

To summarize, BEACON not only significantly outperforms state-of-the-art decentralized MP-MAB algorithms, but is also capable of *empirically* approaching the centralized performance, which is the first time for a decentralized heterogeneous MP-MAB algorithm to the best of our knowledge.

[3]To make meaningful comparisons, non-trivial adjustments and enhancements have been applied to METC, which originally applies only to the linear reward function. Details are given in the supplementary material.

2.2.5 Omitted Algorithmic Details

Some omitted algorithmic details of BEACON are presented in this section.

Orthogonalization Procedure

In the orthogonalization (sometimes also referred to as the initialization) procedure, players estimate the number of players in the MP-MAB game and obtain distinct indices in a fully distributed manner. The initialization technique from Wang et al. (2020a) is adopted in BEACON. It consists of two sub-phases: orthogonalization and rank assignment. The orthogonalization sub-phase aims at assigning each player with a unique external rank $k \in [K]$. It contains a sequence of blocks with length $K+1$, where each player attempts to fixate on arms without collision at first time step and states of fixation (successful or not) are broadcast (enabled by implicit communication). Note that in the original scheme (Wang et al., 2020a), the broadcast is performed on the reserved arm K, which results in the need of $K > M$. To accommodate the scenarios with $K = M$, the broadcast can take place sequentially on arm 1 to arm K. In the rank assignment sub-phase, a modified Round-Robin sequential hopping scheme helps the players convert their external ranks to internal ranks $m \in [M]$ and estimate the overall number of players M. Detailed algorithms can be found in Wang et al. (2020a). Using the same proofs in Lemma 1 and Lemma 2 in Wang et al. (2020a), we have the following performance characterization.

Lemma 2.2.11. *The expected duration of the orthogonalization procedure in BEACON is less than* $\frac{K^2 M}{K-M} + 2K$ *time steps. Once the procedure completes, all players correctly learn the number of players* M *and each of them is assigned with a unique index between* 1 *and* M*.*

Detailed Communication Protocols

In this section, more details of the communication design are presented. First, as illustrated in Section 2.2.1, the implicit communications are performed by having the "receive" player sample one arm and the "send" player either pull (create collision; bit 1) or not pull (create no collision; bit 0) the same arm to transmit one-bit information. Other players that are not communicating would fixate on other arms to avoid interruptions. The arm(s) that the players pull for receiving or avoiding are referred to as "communication arm(s)", which is an arm-player matching and is assigned before the communication happens. In BEACON, the matching of communication arms for epoch $r > 1$ is chosen as the exploration matching in the previous epoch, i.e., S_{r-1}. The benefit of this choice is that with the increasing explorations, S_{r-1} would gradually become near-optimal with a high probability, which also leads to smaller communication losses. Specifically, in epoch r, follower $m > 1$ (resp. the leader) communicates to the leader (resp. follower $m > 1$) by either pulling or not pulling

arm s_1^{r-1} (resp. arm s_m^{r-1}), while the leader (resp. the follower m) stays on arm s_1^{r-1} (resp. arm s_m^{r-1}) during receiving. To make this happen, in addition to the knowledge of index s_m^{r-1} which is assigned to follower m for explorations, index s_1^{r-1} should also be communicated to the followers in the communication phase of epoch $r-1$.

Then, as illustrated in Section 2.2.1, there are three kinds of information to be communicated, which are separately discussed in the following.

Arm statistics. The main idea of the adaptive differential communication (ADC) design is illustrated in Section 2.2.1. However, two important ingredients are missing. The first is when follower m quantizes the arm statistics $\tilde{\mu}_{k,m}^r$ from the collected sample mean $\hat{\mu}_{k,m}^r$ using $\lceil 1 + p_{k,m}^r/2 \rceil$ bits. The least significant bit (LSB) is always ceiled to 1 if $\lceil 1 + p_{k,m}^r/2 \rceil$ bits cannot fully represent $\hat{\mu}_{k,m}^r$. We refer such process of quantizing $\tilde{\mu}_{k,m}^r$ as $\texttt{ceil}(\hat{\mu}_{k,m}^r)$ with $\lceil 1 + p_{k,m}^r/2 \rceil$ bits. This process is needed for the later theoretical analysis to have $\tilde{\mu}_{k,m}^r \geq \hat{\mu}_{k,m}^r$.

The second missing component in ADC is referred to as the **signal-then-communicate** approach. The purpose of this approach is to synchronize the communication order and communication duration among players. It consists of two parts: the leader would first create a collision on the follower's communication arm to indicate the beginning of her statistics sharing; then, since the length of non-zero LSB at the end of $\delta_{k,m}^r$ is not fixed, after receiving the start signal, the follower m would take the following approach to transmit L bits (L is however unknown to the leader), in which creating no collision indicates there are more bits to transmit while creating collision means the end of transmission:

$$\text{collision: start signal} \to \text{no collision} \to \text{one information bit} \to \cdots$$
$$\to \text{no collision} \to \text{one information bit} \to \text{collision: end signal.}$$

Using no collision as an indicator also reduces the practical communication loss, as it avoids creating collisions during communications. In summary, with this signal-to-communicate approach, the original L-bits information of arm statistics would require no more than $(2L+2)$-bits.

The chosen matching and leader's communication arm. In epoch r, the leader needs to notify follower m of both s_m^r (for exploration) and s_1^r (for communication in the next epoch). Similar to sharing arm statistics, the leader has to initiate the communication with a specific follower by creating a collision. Since both arm indices can be communicated via a fixed length of $\lceil \log_2(K) \rceil$ bits, they can be directly transmitted without using no-collisions to synchronize. Thus, with K arms for each player, this part of communication can be done in $2\lceil \log_2(K) \rceil + 1$ bits for each follower.

Batch size. A naive idea to transmit the batch size p_r is to directly notify the followers of this number.

However, the value of p_r is at most $O(\log(T))$, which requires $O(\log\log(T))$ bits. With at most $O(\log(T))$ epochs of communication, directly sharing p_r may lead to a dominating regret. Luckily, sharing p_r only serves to let players explore the same length, which can be achieved by a much simpler and more efficient **stop-upon-signal** approach. Specifically, while p_r is calculated by the leader, rather than broadcasting it to the followers via implicit collisions, she counts the exploration length herself and creates a collision on the exploration arm of each follower upon the end of exploration in this epoch. Upon perceiving collisions, followers become aware that the current exploration phase has ended.

2.2.6 Full Proofs

Proof for Theorem 2.2.8

We begin with the analysis of BEACON with general reward functions, i.e., Theorem 2.2.8, since it is more intuitive than the one for the linear reward function, i.e., Theorem 2.2.2. The latter follows the same spirit of the former but is carefully tailored to the linear reward function.

The complete version of Theorem 2.2.8 is first presented in the following.

Theorem 2.2.12 (**Complete version of Theorem 2.2.8**). *Under Assumptions 2.2.5, 2.2.6, and 2.2.7, the regret of BEACON is upper bounded as*

$$
\begin{aligned}
R(T) \leq & \sum_{(k,m)\in[K]\times[M]} \left[\frac{28\Delta_{\min}^{k,m}\ln(T)}{(f^{-1}(\Delta_{\min}^{k,m}))^2} + \int_{\Delta_{\min}^{k,m}}^{\Delta_{\max}^{k,m}} \frac{28\ln(T)}{(f^{-1}(x))^2}\mathrm{d}x + 4KM\Delta_{\max}^{k,m} \right] \\
& + \frac{6}{\ln 2}M^2K\log_2(K)\Delta_c\ln(T) + \frac{18}{\ln 2}MK\Delta_c\ln(T) + MK\Delta_c + \left(\frac{K^2M}{K-M} + 2K\right)\Delta_c + K\Delta_{\max} \\
= & \tilde{O}\left(\sum_{(k,m)\in[K]\times[M]} \left[\frac{\Delta_{\min}^{k,m}}{(f^{-1}(\Delta_{\min}^{k,m}))^2} + \int_{\Delta_{\min}^{k,m}}^{\Delta_{\max}^{k,m}} \frac{1}{(f^{-1}(x))^2}\mathrm{d}x \right] \log(T) + M^2K\Delta_c\log(T) \right) \\
= & \tilde{O}\left(\sum_{(k,m)\in[K]\times[M]} \frac{\Delta_{\max}^{k,m}\log(T)}{(f^{-1}(\Delta_{\min}^{k,m}))^2} + M^2K\Delta_c\log(T) \right).
\end{aligned}
$$

To facilitate the proof, we introduce (or recall) the following notations:

$V_{\boldsymbol{\mu},*} = \max\{V_{\boldsymbol{\mu},S}|S\in\mathcal{S}\} = \max\{v(\boldsymbol{\mu}_S\odot\boldsymbol{\eta}_S)|S\in\mathcal{S}\}$: the optimal reward value;

$\mathcal{S}_* = \{S|S\in\mathcal{S}, V_{\boldsymbol{\mu},S} = V_{\boldsymbol{\mu},*}\}$: the set of the optimal matchings;

$\mathcal{S}_c = \{S|\exists m\neq n, s_m = s_n\}$: the set of matchings with collisions;

$\mathcal{S}_b = \mathcal{S}\backslash(\mathcal{S}_*\cup\mathcal{S}_c)$: the set of collision-free suboptimal matchings;

$\Delta_{\min}^{k,m} = V_{\boldsymbol{\mu},*} - \max\{V_{\boldsymbol{\mu},S}|S\in\mathcal{S}_b, s_m = k\}$;

$\Delta_{\max}^{k,m} = V_{\boldsymbol{\mu},*} - \min\{V_{\boldsymbol{\mu},S} | S \in \mathcal{S}_b, s_m = k\}$;

$\Delta_{\min} = \min\{\Delta_{\min}^{k,m}\}$: the smallest reward gap among collision-free matchings;

$\Delta_{\max} = \max\{\Delta_{\max}^{k,m}\}$: the largest reward gap among collision-free matchings;

$\Delta_c = V_{\boldsymbol{\mu},*} - \min\{V_{\boldsymbol{\mu},S} | S \in \mathcal{S}_c\} \leq f(1)$: the largest possible per-step loss upon collisions.

Proof for Theorems 2.2.8 and 2.2.12. The overall regret $R(T)$ can be decomposed into three parts: the exploration regret $R_e(T)$, the communication regret $R_c(T)$, and the other regret $R_o(T)$, i.e.,

$$R(T) = R_e(T) + R_c(T) + R_o(T).$$

The exploration regret $R_e(T)$ and the communication regret $R_c(T)$ are caused by exploration and communication phases, respectively, and are analyzed in the following subsections. The other regret $R_o(T)$ contains the regret caused by orthogonalization and activation, i.e., the explorations before epoch 1, and can be easily bounded as

$$R_o(T) \leq \left(\frac{K^2 M}{K - M} + 2K\right) \Delta_c + K\Delta_{\max}, \tag{2.4}$$

where the first term is the regret from orthogonalization (Lemma 2.2.11) and the second term is the regret from activation.

With Lemmas 2.2.13 and 2.2.14, which bound $R_c(T)$ and $R_e(T)$ respectively, established in the following subsections, and the bound on $R_o(T)$ in Eqn. (2.4), Theorems 2.2.8 and 2.2.12 can be directly proved. □

Lemma 2.2.13. *For BEACON, under time horizon T, the cumulative length of all communication phases D_c is bounded as*

$$\mathbb{E}[D_c] \leq \frac{6}{\ln 2} M^2 K \log_2(K) \ln(T) + \frac{18}{\ln 2} MK \ln(T) + MK,$$

and the communication loss $R_c(T)$ is bounded as

$$R_c(T) \leq \mathbb{E}[D_c]\Delta_c \leq \frac{6}{\ln 2} M^2 K \log_2(K) \Delta_c \ln(T) + \frac{18}{\ln 2} MK\Delta_c \ln(T) + MK\Delta_c.$$

Proof for Lemma 2.2.13. As illustrated in Section 2.2.5, communication phases consist of three parts of information sharing: arm statistics $\tilde{\mu}_{k,m}^r$, the chosen matching S_r, and the batch size parameter p_r. With

the detailed communication protocol described in Section 2.2.5, we bound the communication lengths of the aforementioned three parts, respectively.

Part I: Arm statistics. We take arm $(k,m), m \neq 1$ as an example. In epoch 1, $\tilde{\mu}^0_{k,m}$ is initialized as 0 while $\bar{\mu}^1_{k,m}$ is the value of one random utility sample from arm (k,m). With $p^1_{k,m} = \lfloor \log_2(T^1_{k,m}) \rfloor = \lfloor \log_2(1) \rfloor = 0$, $\tilde{\mu}^1_{k,m}$ is quantized from $\hat{\mu}^1_{k,m}$ with $1 + p^1_{k,m} = 1$ bit. The difference $\tilde{\delta}^1_{k,m} = \tilde{\mu}^1_{k,m} - \tilde{\mu}^0_{k,m} = \tilde{\mu}^1_{k,m}$ is transmitted and it contains only 1 bit.

In epoch $r > 1$, if $p^r_{k,m} > p^{r-1}_{k,m}$, i.e., $p^r_{k,m} = p^{r-1}_{k,m} + 1$, arm statistics of arm (k,m) should be communicated via the truncated version of the difference $\tilde{\delta}^r_{k,m} = \tilde{\mu}^r_{k,m} - \tilde{\mu}^{r-1}_{k,m}$. Then, we can bound the duration of communication through bounding $\tilde{\delta}^r_{k,m}$. Specifically, it holds that

$$\begin{aligned}
|\tilde{\delta}^r_{k,m}| &= |\tilde{\mu}^r_{k,m} - \tilde{\mu}^{r-1}_{k,m}| \\
&= |\tilde{\mu}^r_{k,m} - \hat{\mu}^r_{k,m} - (\tilde{\mu}^{r-1}_{k,m} - \hat{\mu}^{r-1}_{k,m}) + (\hat{\mu}^r_{k,m} - \hat{\mu}^{r-1}_{k,m})| \\
&\leq |\tilde{\mu}^r_{k,m} - \hat{\mu}^r_{k,m}| + |\tilde{\mu}^{r-1}_{k,m} - \hat{\mu}^{r-1}_{k,m}| + |\hat{\mu}^r_{k,m} - \hat{\mu}^{r-1}_{k,m}| \\
&\overset{(a)}{\leq} \sqrt{\frac{1}{2^{p^r_{k,m}}}} + \sqrt{\frac{1}{2^{p^r_{k,m}-1}}} + |\hat{\mu}^r_{k,m} - \hat{\mu}^{r-1}_{k,m}|,
\end{aligned}$$

where inequality (a) is due to the quantization process specified Section 2.2.1, i.e., $\tilde{\mu}^r_{k,m} = \texttt{ceil}(\hat{\mu}^r_{k,m})$ with $\lceil 1 + p^r_{k,m}/2 \rceil$ bits. This quantization leads to a quantization error of at most $2^{-p^r_{k,m}/2}$. Further, denoting $\gamma^{k,m}_\tau$ as the τ-th random utility sample from arm (k,m) during exploration phases, we can rewrite the difference $\hat{\mu}^r_{k,m} - \hat{\mu}^{r-1}_{k,m}$ as

$$\begin{aligned}
\hat{\mu}^r_{k,m} - \hat{\mu}^{r-1}_{k,m} &= \frac{\sum_{\tau=1}^{2^{p^r_{k,m}}} \gamma^{k,m}_\tau}{2^{p^r_{k,m}}} - \frac{\sum_{\tau=1}^{2^{p^r_{k,m}-1}} \gamma^{k,m}_\tau}{2^{p^r_{k,m}-1}} \\
&= \frac{\sum_{\tau=1}^{2^{p^r_{k,m}-1}} \gamma^{k,m}_\tau + \sum_{\tau=1+2^{p^r_{k,m}-1}}^{2^{p^r_{k,m}}} \gamma^{k,m}_\tau}{2^{p^r_{k,m}}} - \frac{\sum_{\tau=1}^{2^{p^r_{k,m}-1}} \gamma^{k,m}_\tau}{2^{p^r_{k,m}-1}} \\
&= \frac{\sum_{\tau=1+2^{p^r_{k,m}-1}}^{2^{p^r_{k,m}}} \gamma^{k,m}_\tau - \sum_{\tau=1}^{2^{p^r_{k,m}-1}} \gamma^{k,m}_\tau}{2^{p^r_{k,m}}} \\
&= \frac{1}{2^{p^r_{k,m}}} \sum_{\tau=1}^{2^{p^r_{k,m}-1}} \left(\gamma^{k,m}_{\tau+2^{p^r_{k,m}-1}} - \gamma^{k,m}_\tau \right)
\end{aligned}$$

which is a $\frac{1}{\sqrt{2^{p^r_{k,m}+1}}}$-sub-Gaussian random variable since the utility samples are independent across time. Thus, we can further derive that, with a dummy variable $x \geq \sqrt{\ln 2}$,

$$\mathbb{P}\left(\left| \hat{\mu}^r_{k,m} - \hat{\mu}^{r-1}_{k,m} \right| \geq \sqrt{\frac{x^2}{2^{p^r_{k,m}}}} \right) \leq 2\exp\left[-2^{p^r_{k,m}} \frac{x^2}{2^{p^r_{k,m}}} \right] \leq 2\exp[-x^2]$$

$$
\begin{aligned}
&\Rightarrow \mathbb{P}\left(|\tilde{\delta}^r_{k,m}| \geq \sqrt{\frac{1}{2^{p^r_{k,m}}}} + \sqrt{\frac{1}{2^{p^r_{k,m}-1}}} + \sqrt{\frac{x^2}{2^{p^r_{k,m}}}}\right) \leq 2\exp[-x^2] \\
&\overset{(a)}{\Rightarrow} \mathbb{P}\left(L^r_{k,m} \geq 3 + \frac{p^r_{k,m}}{2} + \log_2\left(\frac{1+\sqrt{2}+x}{\sqrt{2^{p^r_{k,m}}}}\right)\right) \leq 2\exp[-x^2] \\
&\Rightarrow \mathbb{P}\left(L^r_{k,m} \geq 3 + \log_2\left(3+x\right)\right) \leq 2\exp[-x^2] \\
&\Rightarrow \mathbb{P}\left(L^r_{k,m} \leq 3 + \log_2\left(3+x\right)\right) \geq 1 - 2\exp[-x^2] \\
&\overset{(b)}{\Rightarrow} \mathbb{P}\left(L^r_{k,m} \leq l\right) \geq 1 - 2\exp\left[-(2^{l-3}-3)^2\right]
\end{aligned}
$$

where $L^r_{k,m}$ in implication (a) is the length of the truncated version $|\tilde{\delta}^r_{k,m}|$ and is upper bounded by

$$
\begin{aligned}
L^r_{k,m} &\leq \lceil 1 + p^r_{k,m}/2 \rceil - \lfloor \log_2(1/|\tilde{\delta}^r_{k,m}|) \rfloor \\
&\leq 3 + p^r_{k,m}/2 + \log_2(|\tilde{\delta}^r_{k,m}|).
\end{aligned}
$$

In deriving (b), we substitute the variable $3 + \log_2(3+x)$ with l, which satisfies that $l \geq 3 + \log_2(3+\sqrt{\ln 2})$, and thus equivalently $x = 2^{l-3} - 3$. With the above results and viewing $L^r_{k,m}$ as a random variable, we have that its cumulative distribution function (CDF) $F_{L^r_{k,m}}(l)$ satisfies the following property:

$$
\forall l \geq 5 > 3 + \log_2(3+\sqrt{\ln 2}), F_{L^r_{k,m}}(l) = \mathbb{P}\left(L^r_{k,m} \leq l\right) \geq 1 - 2\exp\left[-(2^{l-3}-3)^2\right].
$$

Using the property of CDF, we can bound the expectation of $L^r_{k,m}$ as

$$
\begin{aligned}
\mathbb{E}\left[L^r_{k,m}\right] &= \sum_{l=0}^{\infty}(1 - F_{L^r_{k,m}}(l)) \\
&\leq 6 + \sum_{l=6}^{\infty} 2\exp\left[-(2^{l-3}-3)^2\right] \\
&\leq 6 + \int_{l=5}^{\infty} 2\exp\left[-(2^{l-3}-3)^2\right] \mathrm{d}l \\
&\leq 7.
\end{aligned}
$$

Thus, we have that in expectation, the truncated version of $|\tilde{\delta}^r_{k,m}|$ has a length that is less than 7 bits. In addition, 1-bit information should also be transmitted to indicate the sign of $\tilde{\delta}^r_{k,m}$. As a summary, in expectation, 8 bits is sufficient to represent the truncated version of $\tilde{\delta}^r_{k,m}$,

With overall time horizon of T, there are at most $\log_2(T)$ statistics updates of arm (k,m) in addition to the first epoch. The expected communication duration for arm statistics D_s is bounded as

$$\begin{aligned}\mathbb{E}\left[D_s\right] \overset{(a)}{=} & \underbrace{MK}_{\text{epoch } r=1} + \underbrace{\mathbb{E}\left[\sum_r \sum_{(k,m):p^r_{k,m}>p^{r-1}_{k,m}} (2+2(L^r_{k,m}+1))\right]}_{\text{epoches } r>1} \\ \leq & MK + (2+2\times 8)MK\log_2(T) \\ \leq & \, 18MK\log_2(T) + MK \\ = & \frac{18}{\ln 2}MK\ln(T) + MK, \end{aligned} \tag{2.5}$$

where equation (a) takes the signal-then-communicate protocol described in Section 2.2.5 into consideration, where transmitting $\tilde{\delta}^r_{k,m}$ consists of 1 step of the leader notifying the follower to start, $(L^r_{k,m}+1)$ steps of the truncated version of $\tilde{\delta}^r_{k,m}$ and correspondingly $(L^r_{k,m}+2)$ steps of synchronization between the leader and follower.

Part II & III: Matching choice and batch size. These two parts of communications are relatively easy to bound. In each epoch r, the leader initiates and then transmits two arm indices (s^r_1 and s^r_m) to each follower m, thus, the communication duration D_m for matching assignments is bounded as

$$\begin{aligned} D_m = & \sum_r (M-1)(1+2\lceil\log_2(K)\rceil) \\ \leq & \, (M-1)(2\log_2(K)+3)MK\log_2(T) \\ < & \frac{1}{\ln 2}M^2K(2\log_2(K)+3)\ln(T). \end{aligned} \tag{2.6}$$

For the communication duration D_b for the batch size, as illustrated in Section 2.2.5, the leader notifies followers to stop exploring by sending stopping signals. Thus, it holds that

$$D_b = \sum_r (M-1) \leq (M-1)MK\log_2(T) < \frac{1}{\ln 2}M^2K\ln(T). \tag{2.7}$$

By combining Eqns. (2.5), (2.6) and (2.7), Lemma 2.2.13 can be obtained as

$$\begin{aligned} \mathbb{E}[D_c] = & \, \mathbb{E}[D_s] + \mathbb{E}[D_m] + \mathbb{E}[D_b] \\ \leq & \frac{18}{\ln 2}MK\ln(T) + MK + \frac{1}{\ln 2}M^2(2\log_2(K)+3)K\ln(T) + \frac{1}{\ln 2}M^2K\ln(T) \\ \leq & \frac{6}{\ln 2}M^2K\log_2(K)\ln(T) + \frac{18}{\ln 2}MK\ln(T) + MK. \end{aligned}$$

□

Lemma 2.2.14. *For BEACON, under time horizon T, the exploration regret is upper bounded as*

$$R_e(T) \leq \sum_{(k,m)\in[K]\times[M]} \left[\frac{28\Delta_{\min}^{k,m}\ln(T)}{(f^{-1}(\Delta_{\min}^{k,m}))^2} + \int_{\Delta_{\min}^{k,m}}^{\Delta_{\max}^{k,m}} \frac{28\ln(T)}{(f^{-1}(x))^2}\mathrm{d}x + 4KM\Delta_{\max}^{k,m} \right].$$

Proof for Lemma 2.2.14. The following proof is inspired by the proof for CUCB in Chen et al. (2013). However, Chen et al. (2013) does not consider the batched structure, which introduces additional challenges for the proof here. To better characterize the exploration regret, we introduce the following notations:

$$\mathcal{S}_b^{k,m} = \{S | S \in \mathcal{S}_b, s_m = k\} = \{S_1^{k,m}, ..., S_{N(k,m)}^{k,m}\};$$
$$\Delta_n^{k,m} = V_{\boldsymbol{\mu},*} - V_{\boldsymbol{\mu},S_n^{k,m}}, \forall n \in \{1, ..., N(k,m)\},$$

where $\mathcal{S}_b^{k,m}$ is the set of collision-free sub-optimal matchings that contain arm (k,m) and we denote its size as $N(k,m)$. $\Delta_n^{k,m}$ denotes the sub-optimality gap of the matching $S_n^{k,m}$. In the following proof, we re-arrange the set $\mathcal{S}_b^{k,m} = \{S_1^{k,m}, ..., S_{N(k,m)}^{k,m}\}$ in a decreasing order w.r.t. the gap $\Delta_n^{k,m}$, i.e., if $n_1 \geq n_2$, $\Delta_{n_1}^{k,m} \leq \Delta_{n_2}^{k,m}$. Also, for convenience, we denote $\Delta_{N(k,m)+1}^{k,m} = 0$. Furthermore, it naturally holds that $\Delta_{\min}^{k,m} = \Delta_{N(k,m)}^{k,m}$ and $\Delta_{\max}^{k,m} = \Delta_1^{k,m}$.

We denote $q_n^{k,m}, \forall n \in \{1, ..., N(k,m)\}$ as the integer such that

$$2^{q_n^{k,m}-1} \leq \frac{14\ln(T)}{(f^{-1}(\Delta_n^{k,m}))^2} < 2^{q_n^{k,m}} < \frac{28\ln(T)}{(f^{-1}(\Delta_n^{k,m}))^2}.$$

In addition, we define $q_0^{k,m} = 0$ and $q_{N(k,m)+1}^{k,m} = \lceil \log_2(T) \rceil$. Note that with the above definition of $q_n^{k,m}$, it holds that

$$\forall p \geq q_n^{k,m}, f\left(2\sqrt{\frac{3\ln t_r}{2^{p+1}}} + \sqrt{\frac{1}{2^p}}\right) \leq f\left(3\sqrt{\frac{3\ln t_r}{2^{p+1}}}\right) \leq f\left(3\sqrt{\frac{3\ln T}{2^{p+1}}}\right) < \Delta_n^{k,m}, \tag{2.8}$$

which is a key property that is utilized in the subsequent proofs.

For epoch r, we define the "representative arm" $\rho_r = (s_m^r, m)$ as one of the arms in S_r such that $p_{s_m^r,m}^r = p_r$. If there are more than one arm in S_r with arm counter p_r, ρ_r is randomly chosen from them. Thus, it is guaranteed that there is one and only one representative arm for each exploration phase. With the arm counter updating rule specified in Section 2.2.1, the counter of arm ρ_r will certainly increase by 1 after epoch r.

Step I: Regret decomposition. With respect to the representative arm, we decompose the exploration regret as

$$\begin{aligned}
R_e(T) &= \mathbb{E}\left[\sum_r 2^{p_r}(V_{\boldsymbol{\mu},*} - V_{\boldsymbol{\mu},S_r})\right] \\
&= \mathbb{E}\left[\sum_r \sum_{(k,m)\in[K]\times[M]} 2^{p_r}(V_{\boldsymbol{\mu},*} - V_{\boldsymbol{\mu},S_r})\mathbb{1}\left\{\rho_r = (k,m)\right\}\right] \\
&\overset{(a)}{=} \mathbb{E}\left[\sum_r \sum_{(k,m)\in[K]\times[M]} 2^{p^r_{k,m}}(V_{\boldsymbol{\mu},*} - V_{\boldsymbol{\mu},S_r})\mathbb{1}\left\{\rho_r = (k,m)\right\}\right] \\
&\overset{(b)}{=} \mathbb{E}\left[\sum_r \sum_{(k,m)\in[K]\times[M]} \sum_{n=1}^{N(k,m)} 2^{p^r_{k,m}}\Delta_n^{k,m}\mathbb{1}\left\{\rho_r = (k,m), S_r = S_n^{k,m}\right\}\right] \\
&\overset{(c)}{=} \mathbb{E}\left[\sum_{(k,m)\in[K]\times[M]} \sum_{p_{k,m}\geq 0} \sum_{n=1}^{N(k,m)} 2^{p_{k,m}}\Delta_n^{k,m}\mathbb{1}\left\{S_{k,m,p_{k,m}} = S_n^{k,m}\right\}\right] \\
&\overset{(d)}{=} \sum_{(k,m)\in[K]\times[M]} R_e^{k,m}(T),
\end{aligned} \tag{2.9}$$

where equality (a) is from the definition of the representative arm that if $\rho_r = (k,m)$, it holds that $p_r = p^r_{k,m}$. Equality (b) further associates the regret of each exploration phase with specific sub-optimal matchings. $S_{k,m,p_{k,m}}$ denotes the exploration matching with representative arm (k,m) and the corresponding arm counter $p_{k,m}$. Equality (c) holds because once $\rho_r = (k,m)$, its arm counter will increase. Equality (d) denotes $R_e^{k,m}(T) := \mathbb{E}\left[\sum_{p_{k,m}>0}\sum_{n=1}^{N(k,m)} 2^{p_{k,m}}\Delta_n^{k,m}\mathbb{1}\left\{S_{k,m,p_{k,m}} = S_n^{k,m}\right\}\right]$, which represents the regret associated with arm (k,m).

For term $R_e^{k,m}(T)$, we further have

$$\begin{aligned}
R_e^{k,m}(T) =& \mathbb{E}\left[\sum_{p_{k,m}\geq 0}\sum_{n=1}^{N(k,m)} 2^{p_{k,m}}\Delta_n^{k,m}\mathbb{1}\left\{S_{k,m,p_{k,m}} = S_n^{k,m}\right\}\right] \\
=& \sum_{p_{k,m}\geq 0}\sum_{n=1}^{N(k,m)} 2^{p_{k,m}}\Delta_n^{k,m}\mathbb{P}\left(S_{k,m,p_{k,m}} = S_n^{k,m}\right) \\
\overset{(a)}{\leq}& \sum_{p_{k,m}>0}\sum_{n=1}^{N(k,m)} 2^{p_{k,m}}\Delta_n^{k,m}\mathbb{P}\left(S_{k,m,p_{k,m}} = S_n^{k,m}|\mathcal{E}_{k,m,p_{k,m}}\right)\mathbb{P}\left(\mathcal{E}_{k,m,p_{k,m}}\right) \\
&+ \sum_{p_{k,m}\geq 0}\sum_{n=1}^{N(k,m)} 2^{p_{k,m}}\Delta_n^{k,m}\mathbb{P}\left(S_{k,m,p_{k,m}} = S_n^{k,m}|\bar{\mathcal{E}}_{k,m,p_{k,m}}\right)\mathbb{P}\left(\bar{\mathcal{E}}_{k,m,p_{k,m}}\right) \\
\leq& \sum_{p_{k,m}\geq 0}\sum_{n=1}^{N(k,m)} 2^{p_{k,m}}\Delta_n^{k,m}\mathbb{P}\left(S_{k,m,p_{k,m}} = S_n^{k,m}|\mathcal{E}_{k,m,p_{k,m}}\right)
\end{aligned}$$

$$
\begin{aligned}
&+ \sum_{p_{k,m}\geq 0} 2^{p_{k,m}} \Delta_{\max}^{k,m} \mathbb{P}\left(\bar{\mathcal{E}}_{k,m,p_{k,m}}\right) \\
&\leq \underbrace{\sum_{h=0}^{N(k,m)} \sum_{q_h^{k,m} \leq p_{k,m} < q_{h+1}^{k,m}} \sum_{n=1}^{N(k,m)} 2^{p_{k,m}} \Delta_n^{k,m} \mathbb{P}\left(S_{k,m,p_{k,m}} = S_n^{k,m} | \mathcal{E}_{k,m,p_{k,m}}\right)}_{\text{term (A)}} \\
&\quad + \underbrace{\sum_{p_{k,m}\geq 0} 2^{p_{k,m}} \Delta_{\max}^{k,m} \mathbb{P}\left(\bar{\mathcal{E}}_{k,m,p_{k,m}}\right)}_{\text{term (B)}},
\end{aligned}
$$

where equality (a) introduces the notion of the "nice event" $\mathcal{E}_{k,m,p_{k,m}}$, which is described in the following.

At epoch r, the nice event $\mathcal{E}_r$ is defined as

$$
\mathcal{E}_r = \left\{ \forall (k,m) \in [K] \times [M], -\sqrt{\frac{3\ln t_r}{2^{p_{k,m}^r+1}}} < \tilde{\mu}_{k,m}^r - \mu_{k,m} < \sqrt{\frac{3\ln t_r}{2^{p_{k,m}^r+1}}} + \sqrt{\frac{1}{2^{p_{k,m}^r}}} \right\}.
$$

Furthermore, when the representative arm in epoch r is arm (k,m) with counter $p_{k,m}$, $\mathcal{E}_r$ is denoted as $\mathcal{E}_{k,m,p_{k,m}}$.

Step II: Bounding term (B). We start with term (B) by bounding the probability that event $\bar{\mathcal{E}}_r$ happens. Specifically, it holds that

$$
\begin{aligned}
\mathbb{P}\left(\bar{\mathcal{E}}_r\right) &\leq \sum_{(k,m)\in[K]\times[M]} \mathbb{P}\left(\tilde{\mu}_{k,m}^r - \mu_{k,m} \leq -\sqrt{\frac{3\ln t_r}{2^{p_{k,m}^r+1}}}\right) \\
&\quad + \sum_{(k,m)\in[K]\times[M]} \mathbb{P}\left(\tilde{\mu}_{k,m}^r - \mu_{k,m} \geq \sqrt{\frac{3\ln t_r}{2^{p_{k,m}^r+1}}} + \sqrt{\frac{1}{2^{p_{k,m}^r}}}\right) \\
&= \sum_{(k,m)\in[K]\times[M]} \mathbb{P}\left(\tilde{\mu}_{k,m}^r - \hat{\mu}_{k,m}^r + \hat{\mu}_{k,m}^r - \mu_{k,m} \leq -\sqrt{\frac{3\ln t_r}{2^{p_{k,m}^r+1}}}\right) \\
&\quad + \sum_{(k,m)\in[K]\times[M]} \mathbb{P}\left(\tilde{\mu}_{k,m}^r - \hat{\mu}_{k,m}^r + \hat{\mu}_{k,m}^r - \mu_{k,m} \geq \sqrt{\frac{3\ln t_r}{2^{p_{k,m}^r+1}}} + \sqrt{\frac{1}{2^{p_{k,m}^r}}}\right) \\
&\overset{(a)}{\leq} \sum_{(k,m)\in[K]\times[M]} \mathbb{P}\left(\hat{\mu}_{k,m}^r - \mu_{k,m} \leq -\sqrt{\frac{3\ln t_r}{2^{p_{k,m}^r+1}}}\right) \\
&\quad + \sum_{(k,m)\in[K]\times[M]} \mathbb{P}\left(\hat{\mu}_{k,m}^r - \mu_{k,m} \geq \sqrt{\frac{3\ln t_r}{2^{p_{k,m}^r+1}}}\right) \\
&\leq \sum_{(k,m)\in[K]\times[M]} \sum_{p_{k,m}=0}^{\lfloor \log_2(t_r)\rfloor} 2\mathbb{P}\left(\hat{\mu}_{k,m}^r - \mu_{k,m} \geq \sqrt{\frac{3\ln t_r}{2^{p_{k,m}^r+1}}}, p_{k,m}^r = p_{k,m}\right) \\
&\leq \sum_{(k,m)\in[K]\times[M]} \sum_{p_{k,m}=0}^{\lfloor \log_2(t_r)\rfloor} 2\mathbb{P}\left(\frac{\sum_{\tau=1}^{2^{p_{k,m}}} \gamma_\tau^{k,m}}{2^{p_{k,m}}} - \mu_{k,m} \geq \sqrt{\frac{3\ln t_r}{2^{p_{k,m}+1}}}\right)
\end{aligned}
$$

$$
\begin{aligned}
&\overset{(b)}{\leq} \sum_{(k,m)\in[K]\times[M]} \sum_{p_{k,m}=0}^{\lfloor \log_2(t_r)\rfloor} 2\exp\left[-2\cdot 2^{p_{k,m}}\frac{3\ln t_r}{2^{p_{k,m}+1}}\right] \\
&\leq 2KM\frac{\lfloor \log_2(t_r)\rfloor+1}{(t_r)^3} \\
&\leq 2KM\frac{1}{(t_r)^2} \\
&\overset{(c)}{\leq} 2KM\frac{1}{(2^{p_r})^2},
\end{aligned} \tag{2.10}
$$

where inequality (a) holds because $\tilde{\mu}^r_{k,m} = \texttt{ceil}(\hat{\mu}^r_{k,m})$ with $\lceil 1+p^r_{k,m}/2\rceil$ bits and $\tilde{\mu}^r_{k,m} - \hat{\mu}^r_{k,m} > 0$ Inequality (b) is from the Hoeffding's inequality. Inequality (c) utilizes the observation that $t_r \geq 2^{p_r}$.

With Eqn. (2.10), we can further bound term (B) as

$$
\begin{aligned}
\text{term (B)} =& \sum_{p_{k,m}\geq 0} 2^{p_{k,m}}\Delta^{k,m}_{\max}\mathbb{P}\left(\bar{\mathcal{E}}_{k,m,p_{k,m}}\right) \\
\overset{(a)}{\leq}& 2\sum_{p_{k,m}\geq 0} 2^{p_{k,m}}\Delta^{k,m}_{\max}\cdot KM\frac{1}{(2^{p_{k,m}})^2} \\
=& 2\sum_{p_{k,m}\geq 0} \Delta^{k,m}_{\max}\cdot KM\frac{1}{2^{p_{k,m}}} \\
\leq& 4KM\Delta^{k,m}_{\max},
\end{aligned}
$$

where inequality (a) is with Eqn. (2.10) and $p_r = p_{k,m}$.

Step III: Bounding term (A). Before bounding term (A), we first establish the following implications. For epoch r, if $\rho_r = (k,m)$ and $p_r = p^r_{k,m} = p_{k,m}$, denoting $\bar{\boldsymbol{\mu}}_r$ and S_r as $\bar{\boldsymbol{\mu}}^{k,m,p_{k,m}}$ and $S_{k,m,p_{k,m}}$ respectively, if event $\mathcal{E}_{k,m,p_{k,m}}$ happens, we have

$$
\begin{aligned}
&p_{k,m} \geq q^{k,m}_h, \text{ the oracle outputs } S_{k,m,p_{k,m}} = S^{k,m}_n \\
\Rightarrow& p_{k,m} \geq q^{k,m}_h, \forall S\in\mathcal{S}_*\backslash\mathcal{S}_c, v(\bar{\boldsymbol{\mu}}^{k,m,p_{k,m}}_{S^{k,m}_n}\odot\boldsymbol{\eta}_{S^{k,m}_n}) \geq v(\bar{\boldsymbol{\mu}}^{k,m,p_{k,m}}_{S}\odot\boldsymbol{\eta}_{S}) \\
\Rightarrow& p_{k,m} \geq q^{k,m}_h, \forall S\in\mathcal{S}_*\backslash\mathcal{S}_c, v(\bar{\boldsymbol{\mu}}^{k,m,p_{k,m}}_{S^{k,m}_n}) \geq v(\bar{\boldsymbol{\mu}}^{k,m,p_{k,m}}_{S}) \\
\overset{(a)}{\Rightarrow}& p_{k,m} \geq q^{k,m}_h, \forall S\in\mathcal{S}_*\backslash\mathcal{S}_c, v(\boldsymbol{\mu}_{S^{k,m}_n}) + f\left(\left\|\bar{\boldsymbol{\mu}}^{k,m,p_{k,m}}_{S^{k,m}_n} - \boldsymbol{\mu}_{S^{k,m}_n}\right\|_\infty\right) \geq v(\bar{\boldsymbol{\mu}}^{k,m,p_{k,m}}_{S}) \\
\overset{(b)}{\Rightarrow}& p_{k,m} \geq q^{k,m}_h, \forall S\in\mathcal{S}_*\backslash\mathcal{S}_c, V_{\boldsymbol{\mu},S^{k,m}_n} + f\left(2\sqrt{\frac{3\ln t_r}{2^{p_{k,m}+1}}} + \sqrt{\frac{1}{2^{p_{k,m}}}}\right) \geq V_{\boldsymbol{\mu},*} \\
\overset{(c)}{\Rightarrow}& p_{k,m} \geq q^{k,m}_h, V_{S^{k,m}_n} + \Delta^{k,m}_h > V_*,
\end{aligned} \tag{2.11}
$$

where implication (a) is from Assumption 2.2.7 and implication (b) utilizes the definition of $\mathcal{E}_{k,m,p_{k,m}}$, Assumption 2.2.6 and that arms in $S_{k,m,p_{k,m}}$ have counters at least $p_{k,m}$. Implication (c) is from the definition

of $q_h^{k,m}$ and Eqn. (2.8).

With Eqn. (2.11), we can get that if $p_{k,m} \geq q_h^{k,m}$, the matchings $S_n^{k,m}$ with $n \leq h$ cannot be S_r; otherwise it contradicts with the definition of $\Delta_h^{k,m}$. Thus, we can further bound term (A) as

$$\begin{aligned}
\text{term (A)} &= \sum_{h=0}^{N(k,m)} \sum_{q_h^{k,m} \leq p_{k,m} < q_{h+1}^{k,m}} \sum_{n=1}^{N(k,m)} 2^{p_{k,m}} \Delta_n^{k,m} \mathbb{P}\left(S_{k,m,p_{k,m}} = S_n^{k,m} | \mathcal{E}_{k,m,p_{k,m}}\right) \\
&= \sum_{h=0}^{N(k,m)} \sum_{q_h^{k,m} \leq p_{k,m} < q_{h+1}^{k,m}} \sum_{n=h+1}^{N(k,m)} 2^{p_{k,m}} \Delta_n^{k,m} \mathbb{P}\left(S_{k,m,p_{k,m}} = S_n^{k,m} | \mathcal{E}_{k,m,p_{k,m}}\right) \\
&\overset{(a)}{\leq} \sum_{h=0}^{N(k,m)} \sum_{q_h^{k,m} \leq p_{k,m} < q_{h+1}^{k,m}} \sum_{n=h+1}^{N(k,m)} 2^{p_{k,m}} \Delta_{h+1}^{k,m} \mathbb{P}\left(S_{k,m,p_{k,m}} = S_n^{k,m} | \mathcal{E}_{k,m,p_{k,m}}\right) \\
&\overset{(b)}{\leq} \sum_{h=0}^{N(k,m)} \sum_{q_h^{k,m} \leq p_{k,m} < q_{h+1}^{k,m}} 2^{p_{k,m}} \Delta_{h+1}^{k,m} \\
&= \sum_{h=0}^{N(k,m)} (2^{q_{h+1}^{k,m}} - 2^{q_h^{k,m}}) \Delta_{h+1}^{k,m} \\
&= \sum_{h=0}^{N(k,m)-1} (2^{q_{h+1}^{k,m}} - 2^{q_h^{k,m}}) \Delta_{h+1}^{k,m} \\
&\leq 2^{q_{N(k,m)}^{k,m}} \Delta_{N(k,m)}^{k,m} + \sum_{h=1}^{N(k,m)-1} 2^{q_h^{k,m}} \left(\Delta_h^{k,m} - \Delta_{h+1}^{k,m}\right) \\
&\overset{(c)}{\leq} \frac{28 \Delta_{N(k,m)}^{k,m} \ln(T)}{(f^{-1}(\Delta_{N(k,m)}^{k,m}))^2} + \sum_{h=1}^{N(k,m)-1} \frac{28 \ln(T)}{(f^{-1}(\Delta_h^{k,m}))^2} \left(\Delta_h^{k,m} - \Delta_{h+1}^{k,m}\right) \\
&\overset{(d)}{\leq} \frac{28 \Delta_{N(k,m)}^{k,m} \ln(T)}{(f^{-1}(\Delta_{N(k,m)}^{k,m}))^2} + \int_{\Delta_{N(k,m)}^{k,m}}^{\Delta_1^{k,m}} \frac{28 \ln(T)}{(f^{-1}(x))^2} dx \\
&= \frac{28 \Delta_{\min}^{k,m} \ln(T)}{(f^{-1}(\Delta_{\min}^{k,m}))^2} + \int_{\Delta_{\min}^{k,m}}^{\Delta_{\max}^{k,m}} \frac{28 \ln(T)}{(f^{-1}(x))^2} dx,
\end{aligned}$$

where inequality (a) holds because $\forall n \geq h+1$, $\Delta_n^{k,m} \leq \Delta_{h+1}^{k,m}$, and inequality (b) is from

$$\sum_{n=h+1}^{N(k,m)} \mathbb{P}\left(S_{k,m,p_{k,m}} = S_n^{k,m} | \mathcal{E}_{k,m,p_{k,m}}\right) \leq 1.$$

Inequality (c) is from the definition of $q_n^{k,m}$ and inequality (d) is because $\frac{28 \ln(T)}{(f^{-1}(x))^2}$ is strictly decreasing in $[\Delta_{N(k,m)}^{k,m}, \Delta_1^{k,m}]$.

By combining terms (A) and (B), we have

$$R_e^{k,m}(T) \leq \frac{28 \Delta_{\min}^{k,m} \ln(T)}{(f^{-1}(\Delta_{\min}^{k,m}))^2} + \int_{\Delta_{\min}^{k,m}}^{\Delta_{\max}^{k,m}} \frac{28 \ln(T)}{(f^{-1}(x))^2} dx + 4KM\Delta_{\max}^{k,m}$$

$$\leq \frac{28\Delta_{\max}^{k,m}\ln(T)}{(f^{-1}(\Delta_{\min}^{k,m}))^2} + 4KM\Delta_{\max}^{k,m}.$$

Overall, we conclude that

$$\begin{aligned}
R_e(T) &= \sum_{(k,m)\in[K]\times[M]} R_e^{k,m}(T) \\
&\leq \sum_{(k,m)\in[K]\times[M]} \left[\frac{28\Delta_{\min}^{k,m}\ln(T)}{(f^{-1}(\Delta_{\min}^{k,m}))^2} + \int_{\Delta_{\min}^{k,m}}^{\Delta_{\max}^{k,m}} \frac{28\ln(T)}{(f^{-1}(x))^2}dx + 4KM\Delta_{\max}^{k,m}\right] \\
&\leq \sum_{(k,m)\in[K]\times[M]} \frac{28\Delta_{\max}^{k,m}\ln(T)}{(f^{-1}(\Delta_{\min}^{k,m}))^2} + 4K^2M^2\Delta_{\max}.
\end{aligned}$$

□

Theorems 2.2.8 and 2.2.12 can be proved by combining Lemmas 2.2.13, 2.2.14, and Eqn. (2.4).

Proof for Theorem 2.2.2

A complete version of Theorem 2.2.2 is first presented in the following.

Theorem 2.2.15 (**Complete version of Theorem 2.2.2**). *With a linear reward function, the regret of BEACON is upper bounded as*

$$\begin{aligned}
R_{\text{linear}}(T) \leq & \sum_{(k,m)\in[K]\times[M]} \frac{3727M}{\Delta_{\min}^{k,m}}\ln(T) + 8K^2M^3 + M^2K \\
& + (22M + 2M\log_2(K))\left[\frac{2MK}{\ln 2}\ln(T) + MK\left(\frac{3M\sqrt{3\ln(T)}}{\sqrt{2}-1} + \frac{8KM^2}{3}\right)\right] \\
&= \tilde{O}\left(\sum_{(k,m)\in[K]\times[M]} \frac{M\log(T)}{\Delta_{\min}^{k,m}} + M^2K\log(T)\right) \\
&= \tilde{O}\left(\frac{M^2K\log(T)}{\Delta_{\min}} + M^2K\log(T)\right).
\end{aligned}$$

Proof for Theorems 2.2.2 and 2.2.15. Similar to the previous proof, the overall regret $R_{\text{linear}}(T)$ can be decomposed into three parts: the exploration regret $R_{e,\text{linear}}(T)$, the communication regret $R_{c,\text{linear}}(T)$, and the other regret $R_{o,\text{linear}}(T)$, i.e.,

$$R_{\text{linear}}(T) = R_{e,\text{linear}}(T) + R_{c,\text{linear}}(T) + R_{o,\text{linear}}(T).$$

The last component can be similarly bounded as

$$R_{o,\text{linear}}(T) \leq \left(\frac{K^2 M}{K-M} + 2K\right)\Delta_c + K\Delta_{\max},$$

The communication regret and exploration regret are bounded Lemmas 2.2.16 and 2.2.17 that are presented in the subsequent subsections. Putting them all together completes the proof. □

Lemma 2.2.16. *For BEACON, under time horizon T, the communication loss $R_{c,linear}(T)$ is upper bounded as*

$$R_{c,linear}(T) \leq M^2 K + (22M + 2M\log_2(K))\left[\frac{2MK}{\ln 2}\ln(T) + MK\left(\frac{3M\sqrt{3\ln(T)}}{\sqrt{2}-1} + \frac{8KM^2}{3}\right)\right].$$

Proof for Lemma 2.2.16. From the proof for Lemma 2.2.13, we can draw the following facts:

(i) For epoch 1, communicating $\tilde{\delta}^1_{k,m}$ takes 1 time step;

(ii) For epoch $r > 1$, if $p^r_{k,m} > p^{r-1}_{k,m}$, $\tilde{\delta}^r_{k,m}$ is communicated and the communication in expectation takes $2 + 2 \times (1 + \mathbb{E}[L^r_{k,m}]) \leq 18$ time steps;

(iii) For epoch $r > 1$, the communication of the chosen matching and the batch size parameter takes less than $M(3 + 2\log_2(K)) + M$ time steps.

These facts hold for the general reward functions, thus naturally hold for the linear reward function.

However, with the linear reward function, the loss caused by communication can be characterized more carefully as

$$\begin{aligned}
R_{c,\text{linear}}(T) &\overset{(a)}{\leq} MK \times M \\
&+ \mathbb{E}\left[\sum_r (2 + V_{\boldsymbol{\mu},*} - V_{\boldsymbol{\mu},S_r})\mathbb{1}\{\mathcal{E}_r\}\left[\sum_{(k,m)} 18\mathbb{1}\left\{p^r_{k,m} \geq p^{r-1}_{k,m}\right\} + M(3 + 2\log_2(K)) + M\right]\right] \\
&+ \mathbb{E}\left[\sum_r M\mathbb{1}\{\bar{\mathcal{E}}_r\}\left[\sum_{(k,m)} 18\mathbb{1}\left\{p^r_{k,m} \geq p^{r-1}_{k,m}\right\} + M(3 + 2\log_2(K)) + M\right]\right] \\
&\overset{(b)}{\leq} M^2 K + \sum_r \mathbb{E}\left[(2 + V_{\boldsymbol{\mu},*} - V_{\boldsymbol{\mu},S_r})\mathbb{1}\{\mathcal{E}_r\} + M\mathbb{1}\{\bar{\mathcal{E}}_r\}\right](22M + 2M\log_2(K)) \\
&\overset{(c)}{\leq} M^2 K + \sum_r \left(2 + 3M\sqrt{\frac{3\ln(T)}{2^{p_r+1}}} + 2M\frac{KM}{(2^{p_r})^2}\right)(22M + 2M\log_2(K)) \\
&\leq M^2 K + (22M + 2M\log_2(K))\left[2MK\log_2(T) + MK\sum_{p_r=0}^{\lceil\log_2 T\rceil}\left(3M\sqrt{\frac{3\ln(T)}{2^{p_r+1}}} + 2\frac{KM^2}{(2^{p_r})^2}\right)\right]
\end{aligned}$$

$$\leq M^2K + (22M + 2M\log_2(K))\left[2MK\log_2(T) + MK\left(3M\sqrt{3\ln(T)}\frac{1}{\sqrt{2}-1} + \frac{8KM^2}{3}\right)\right]$$

where inequality (a) is from that there are at most 2 players colliding with each other (leader and one follower) under the nice event $\mathcal{E}_r$. Specifically, with arms in S_r used for communications in epoch r, one communication step leads to a loss at most $2 + V_{\boldsymbol{\mu},*} - V_{\boldsymbol{\mu},S_r}$. Inequality (b) is from that in each epoch $r > 1$, at most M arms statistics need to be communicated. Inequality (c) holds because if the nice event $\mathcal{E}_r$ happens

$$\begin{aligned}
&\forall S \in \mathcal{S}_*\backslash\mathcal{S}_c, v(\bar{\boldsymbol{\mu}}^r_{S_r}) \geq v(\bar{\boldsymbol{\mu}}^r_S)\\
\Rightarrow&\forall S \in \mathcal{S}_*\backslash\mathcal{S}_c, V_{\boldsymbol{\mu},S_r} + M\left(2\sqrt{\frac{3\ln t_r}{2^{p_r+1}}} + \sqrt{\frac{1}{2^{p_r}}}\right) \geq v(\bar{\boldsymbol{\mu}}^r_{S_r}) \geq v(\bar{\boldsymbol{\mu}}^r_S) > v(\boldsymbol{\mu}_S) = V_{\boldsymbol{\mu},*}\\
\Rightarrow&V_{\boldsymbol{\mu},*} - V_{\boldsymbol{\mu},S_r} \leq M\left(2\sqrt{\frac{3\ln t_r}{2^{p_r+1}}} + \sqrt{\frac{1}{2^{p_r}}}\right) \leq 3M\sqrt{\frac{3\ln(T)}{2^{p_r+1}}};
\end{aligned}$$

otherwise, the nice event does not happen with $\mathbb{P}(\bar{\mathcal{E}}_r) \leq \frac{2KM}{(2^{p_r})^2}$ proved in the Eqn. (2.10), $\mathbb{E}[M\mathbb{1}\left\{\bar{\mathcal{E}}_r\right\}] \leq 2M\frac{KM}{(2^{p_r})^2}$. □

Lemma 2.2.17. *For BEACON, under time horizon T, the exploration loss $R_{e,linear}(T)$ is upper bounded as*

$$R_{e,\text{linear}}(T) \leq \sum_{(k,m)} \frac{3727M}{\Delta^{k,m}_{\min}}\ln(T) + 4K^2M^2\Delta_{\max}.$$

Proof for Lemma 2.2.17. The following proof is based on the proof for CUCB with a linear reward function in Kveton et al. (2015), but is carefully designed for the complicated batched exploration. In the following proof, we introduce the following notations:

$$\begin{aligned}
&S^* = [s^*_1, ..., s^*_M] \in \mathcal{S}_*\backslash\mathcal{S}_c\text{: one particular collision-free optimal matching;}\\
&\Delta_{S_r} := V_{\boldsymbol{\mu},*} - V_{\boldsymbol{\mu},S_r};\\
&[\tilde{M}_r] := \{m | m \in [M], s^r_m \neq s^*_m\}.
\end{aligned}$$

Step I: Regret decomposition. First, we can decompose the exploration regret $R_{e,\text{linear}}(T)$ as

$$\begin{aligned}
R_{e,\text{linear}}(T) &= \mathbb{E}\left[\sum_r 2^{p_r}(V_{\boldsymbol{\mu},*} - V_{\boldsymbol{\mu},S_r})\right]\\
&= \mathbb{E}\left[\sum_r 2^{p_r}\Delta_{S_r}\mathbb{1}\left\{\mathcal{E}_r, \Delta_{S_r} > 0\right\}\right] + \mathbb{E}\left[\sum_r 2^{p_r}\Delta_{S_r}\mathbb{1}\left\{\bar{\mathcal{E}}_r, \Delta_{S_r} > 0\right\}\right] \qquad (2.12)
\end{aligned}$$

$$\overset{(a)}{\leq} \underbrace{\mathbb{E}\left[\sum_r 2^{p_r}\Delta_{S_r}\mathbb{1}\left\{\sum_{m\in[\tilde{M}_r]}\left(2\sqrt{\frac{3\ln t_r}{2^{p^r_{s^r_m,m}+1}}}+\sqrt{\frac{1}{2^{p^r_{s^r_m,m}}}}\right)\geq\Delta_{S_r},\Delta_{S_r}>0\right\}\right]}_{\text{term (C)}}$$
$$+\underbrace{\mathbb{E}\left[\sum_r 2^{p_r}\Delta_{S_r}\mathbb{1}\left\{\bar{\mathcal{E}}_r\right\}\right]}_{\text{term (D)}},$$

where inequality (a) is because when the nice event $\mathcal{E}_r$ happens, choosing a sub-optimal matching S_r, i.e., $\Delta_{S_r}>0$, implies

$$\begin{aligned}
&\forall S\in\mathcal{S}_*, v(\bar{\boldsymbol{\mu}}^r_{S_r})\geq v(\bar{\boldsymbol{\mu}}^r_S)\\
\Rightarrow& v(\bar{\boldsymbol{\mu}}^r_{S_r})\geq v(\bar{\boldsymbol{\mu}}^r_{S^*})\\
\Rightarrow&\sum_{m\in[\tilde{M}_r]}\bar{\mu}^r_{s^r_m,m}\geq\sum_{m\in[\tilde{M}_r]}\bar{\mu}^r_{s^*_m,m}\\
\Rightarrow&\sum_{m\in[\tilde{M}_r]}\mu_{s^r_m,m}+\sum_{m\in[\tilde{M}_r]}\left(2\sqrt{\frac{3\ln t_r}{2^{p^r_{s^r_m,m}+1}}}+\sqrt{\frac{1}{2^{p^r_{s^r_m,m}}}}\right)\geq\sum_{m\in[\tilde{M}_r]}\mu_{s^*_m,m}\\
\Rightarrow&\sum_{m\in[\tilde{M}_r]}\left(2\sqrt{\frac{3\ln t_r}{2^{p^r_{s^r_m,m}+1}}}+\sqrt{\frac{1}{2^{p^r_{s^r_m,m}}}}\right)\geq V_{\boldsymbol{\mu},*}-V_{\boldsymbol{\mu},S_r}=\Delta_{S_r}.
\end{aligned}$$

Step II: Bounding term (D). With essentially the same approach of bounding term (B) in the proof of Lemma 2.2.14, especially Eqn. (2.10), we can directly bound term (D) as

$$\text{term (D)}=\mathbb{E}\left[\sum_r 2^{p_r}\Delta_{S_r}\mathbb{1}\left\{\bar{\mathcal{E}}_r\right\}\right]\leq 4K^2M^2\Delta_{\max}.$$

Step III: Bounding term (C). First, we denote event

$$\mathcal{F}_r=\left\{\sum_{m\in[\tilde{M}_r]}\left(2\sqrt{\frac{3\ln t_r}{2^{p^r_{s^r_m,m}+1}}}+\sqrt{\frac{1}{2^{p^r_{s^r_m,m}}}}\right)\geq\Delta_{S_r},\Delta_{S_r}>0\right\},$$

thus

$$\begin{aligned}
\text{term (C)}&=\mathbb{E}\left[\sum_r 2^{p_r}\Delta_{S_r}\mathbb{1}\left\{\sum_{m\in[\tilde{M}_r]}\left(2\sqrt{\frac{3\ln t_r}{2^{p^r_{s^r_m,m}+1}}}+\sqrt{\frac{1}{2^{p^r_{s^r_m,m}}}}\right)\geq\Delta_{S_r},\Delta_{S^r}>0\right\}\right]\\
&=\mathbb{E}\left[\sum_r 2^{p_r}\Delta_{S_r}\mathbb{1}\left\{\mathcal{F}_r\right\}\right].
\end{aligned}$$

Following the ideas in Kveton et al. (2015), we introduce two decreasing sequences of constants:

$$1 = b_0 > b_1 > b_2 > \cdots > b_i > \cdots$$

$$a_1 > a_2 > \cdots > a_i > \cdots$$

such that $\lim_{i\to\infty} a_i = \lim_{i\to\infty} b_i = 0$. Furthermore, we specify q_{i,S_r} as the integer satisfying

$$2^{q_{i,S_r}-1} \leq a_i \frac{M^2}{(\Delta_{S_r})^2} \ln(T) < 2^{q_{i,S_r}} \leq 2a_i \frac{M^2}{(\Delta_{S_r})^2} \ln(T).$$

For convenience, we denote $q_{0,S_r} = 0$ and $q_{\infty,S_r} = \infty$. Also, set H_i^r is defined as

$$\forall i \geq 1, H_i^r = \left\{ m | m \in [\tilde{M}_r], p^r_{s^r_m,m} < q_{i,S_r} \right\},$$

which represents the arms that are not sufficiently sampled compared with q_{i,S_r}, and $H_0^r := [\tilde{M}_r]$.

With the above introduce notations, we define the following infinitely-many events at epoch r as

$$\begin{aligned}
&G_1^r = \{|H_1^r| \geq b_1 M\};\\
&G_2^r = \{|H_1^r| < b_1 M\} \cap \{|H_2^r| \geq b_2 M\};\\
&\quad \cdots\\
&G_i^r = \{|H_1^r| < b_1 M\} \cap \{|H_2^r| < b_2 M\} \cap \cdots \cap \left\{\left|H_{i-1}^r\right| < b_{i-1} M\right\} \cap \{|H_i^r| \geq b_i M\};\\
&\quad \cdots
\end{aligned}$$

Clearly, these events are mutually exclusive. We have the following proposition.

Proposition 2.2.18. *Let*

$$\sqrt{14} \sum_{i=1}^{\infty} \frac{b_{i-1} - b_i}{\sqrt{a_i}} \leq 1. \tag{2.13}$$

If event $\mathcal{F}_r$ happens at epoch r, then there exists i such that G_i^r happens.

This proposition can be proved by assuming that $\mathcal{F}_r$ happens while none of G_i^r happens. Denoting $\bar{G}_r = \overline{\cup_i G_i^r}$, we can get

$$\begin{aligned}
\bar{G}_r &= \overline{\cup_{i=1}^{\infty} G_i^r}\\
&= \cap_{i=1}^{\infty} \bar{G}_i^r
\end{aligned}$$

$$
\begin{aligned}
&= \cap_{i=1}^{\infty} \left[\overline{\left(\cap_{j=1}^{i-1} \left\{ \left| H_j^r \right| < b_j M \right\}\right)} \cup \overline{\left\{ |H_i^r| \geq b_i M \right\}} \right] \\
&= \cap_{i=1}^{\infty} \left[\left(\cup_{j=1}^{i-1} \overline{\left\{ \left| H_j^r \right| < b_j M \right\}} \right) \cup \overline{\left\{ |H_i^r| \geq b_i M \right\}} \right] \\
&= \cap_{i=1}^{\infty} \left[\left(\cup_{j=1}^{i-1} \left\{ \left| H_j^r \right| \geq b_j M \right\}\right) \cup \left\{ |H_i^r| < b_i M \right\} \right] \\
&= \cap_{i=1}^{\infty} \left\{ |H_i^r| < b_i M \right\}.
\end{aligned}
$$

If $\bar{G}_r$ happens, denoting $\tilde{H}_i^r = [\tilde{M}_r] \backslash H_i^r$, which implies $\tilde{H}_{i-1}^r \subseteq \tilde{H}_i^r$ and $[\tilde{M}_r] = \cup_i (\tilde{H}_i^r \backslash \tilde{H}_{i-1}^r)$, then it holds that

$$
\begin{aligned}
&\sum_{m \in [\tilde{M}_r]} \left(2\sqrt{\frac{3\ln T}{2^{p^r_{s^r_m,m}+1}}} + \sqrt{\frac{1}{2^{p^r_{s^r_m,m}}}} \right) \\
\leq & 3\sqrt{3\ln T} \sum_{m \in [\tilde{M}_r]} \frac{1}{\sqrt{2^{p^r_{s^r_m,m}+1}}} \\
= & 3\sqrt{3\ln T} \sum_{i=1}^{\infty} \sum_{m \in \tilde{H}_i^r \backslash \tilde{H}_{i-1}^r} \frac{1}{\sqrt{2^{p^r_{s^r_m,m}+1}}} \\
= & 3\sqrt{3\ln T} \sum_{i=1}^{\infty} \frac{|\tilde{H}_i^r \backslash \tilde{H}_{i-1}^r|}{\sqrt{2^{q_{i-1,S_r}+1}}} \\
\leq & 3\sqrt{3\ln T} \sum_{i=1}^{\infty} \frac{|\tilde{H}_i^r \backslash \tilde{H}_{i-1}^r|}{\sqrt{2a_i \frac{M^2}{(\Delta_{S_r})^2} \ln(T)}} \\
\leq & 3\sqrt{3/2} \frac{\Delta_{S_r}}{M} \sum_{i=1}^{\infty} \left(|H_{i-1}^r| - |H_i^r| \right) \frac{1}{\sqrt{a_i}} \\
= & 3\sqrt{3/2} \frac{\Delta_{S_r}}{M} |H_0^r| \frac{1}{\sqrt{a_1}} + 3\sqrt{3/2} \frac{\Delta_{S_r}}{M} \sum_{i=1}^{\infty} |H_i^r| \left(\frac{1}{\sqrt{a_{i+1}}} - \frac{1}{\sqrt{a_i}} \right) \\
\overset{(a)}{\leq} & 3\sqrt{3/2} \frac{\Delta_{S_r}}{M} b_0 M \frac{1}{\sqrt{a_1}} + 3\sqrt{3/2} \frac{\Delta_{S_r}}{M} \sum_{i=1}^{\infty} b_i M \left(\frac{1}{\sqrt{a_{i+1}}} - \frac{1}{\sqrt{a_i}} \right) \\
< & \sqrt{14} \sum_{i=1}^{\infty} \frac{b_{i-1} - b_i}{\sqrt{a_i}} \Delta_{S_r} \\
\leq & \Delta_{S_r},
\end{aligned}
$$

where inequality is because $|H_i^r < b_i M$ with $\bar{G}_r$ happening. This result contradicts with the definition of $\mathcal{F}_r$ as

$$
\mathcal{F}_r = \left\{ \sum_{m \in [\tilde{M}^r]} \left(2\sqrt{\frac{3\ln t_r}{2^{p^r_{s^r_m,m}+1}}} + \sqrt{\frac{1}{2^{p^r_{s^r_m,m}}}} \right) \geq \Delta_{S_r}, \Delta_{S_r} > 0 \right\}.
$$

With Proposition 2.2.18, when Eqn. (2.13) holds, we can further decompose term (C) as

$$\text{term (C)} = \mathbb{E}\left[\sum_r 2^{p_r}\Delta_{S_r}\mathbb{1}\left\{\mathcal{F}_r\right\}\right] = \mathbb{E}\left[\sum_r\sum_{i=1}^{\infty}2^{p_r}\Delta_{S_r}\mathbb{1}\left\{G_i^r, \Delta_{S_r}>0\right\}\right].$$

Then, the following events are defined

$$G_{i,k,m}^r = G_i^r \cap \left\{m \in [\tilde{M}_r], s_m^r = k, p_{k,m}^r < q_{i,S_r}\right\},$$

which imply that

$$\mathbb{1}\left\{G_i^r, \Delta_{S_r}>0\right\} \le \frac{1}{b_i M}\sum_{(k,m)}\mathbb{1}\left\{G_{i,s_m^r,m}^r, \Delta_{S_r}>0\right\}$$

since at least $b_i M$ arms with event $G_{i,k,m}^r$ happening are required to make G_i^r happen.

Thus, recall $\mathcal{S}_b^{k,m} = \{S|S\in\mathcal{S}_b, s_m = k\} = \{S_1^{k,m}, ..., S_{N(k,m)}^{k,m}\}$, we can get

$$\begin{aligned}
&\text{term (C)} = \mathbb{E}\left[\sum_r\sum_{i=1}^{\infty}2^{p_r}\Delta_{S_r}\mathbb{1}\left\{G_i^r, \Delta_{S_r}>0\right\}\right]\\
&\le \mathbb{E}\left[\sum_r\sum_{i=1}^{\infty}2^{p_r}\Delta_{S_r}\frac{1}{b_i M}\sum_{(k,m)}\mathbb{1}\left\{G_{i,k,m}^r, \Delta_{S_r}>0\right\}\right]\\
&\le \mathbb{E}\left[\sum_r\sum_{i=1}^{\infty}2^{p_r}\Delta_{S_r}\frac{1}{b_i M}\sum_{(k,m)}\mathbb{1}\left\{m\in[\tilde{M}_r], s_m^r = k, p_{k,m}^r < q_{i,S_r}, \Delta_{S_r}>0\right\}\right]\\
&= \mathbb{E}\left[\sum_{(k,m)}\sum_{n=1}^{N(k,m)}\sum_r\sum_{i=1}^{\infty}2^{p_r}\frac{1}{b_i M}\mathbb{1}\left\{s_m^r = k, p_{k,m}^r < q_{i,S_n^{k,m}}, S_r = S_n^{k,m}\right\}\Delta_n^{k,m}\right]\\
&= \mathbb{E}\left[\sum_{(k,m)}\sum_{i=1}^{\infty}\underbrace{\sum_r\sum_{n=1}^{N(k,m)}2^{p_r}\frac{1}{b_i M}\mathbb{1}\left\{s_m^r = k, p_{k,m}^r < q_{i,S_n^{k,m}}, S_r = S_n^{k,m}\right\}\Delta_n^{k,m}}_{\text{term (E)}}\right]\\
&\overset{(a)}{\le} \mathbb{E}\left[\sum_{(k,m)}\left[\sum_{i=1}^{\infty}\frac{6a_i}{b_i}\right]\frac{M}{\Delta_{N(k,m)}^{k,m}}\ln(T)\right]
\end{aligned}$$

where inequality (a) holds because term (E) can be bounded as

$$\begin{aligned}
\text{term (E)} &= \sum_r\sum_{n=1}^{N(k,m)}2^{p_r}\frac{1}{b_i M}\mathbb{1}\left\{s_m^r = k, p_{k,m}^r < q_{i,S_n^{k,m}}, S_r = S_n^{k,m}\right\}\Delta_n^{k,m}\\
&\le 3\times 2^{q_{i,S_1^{k,m}}-1}\frac{\Delta_1^{k,m}}{b_i M} + \frac{1}{b_i M}\sum_{n=2}^{N(k,m)}\left(3\times 2^{q_{i,S_n^{k,m}}-1} - 3\times 2^{q_{i,S_{n-1}^{k,m}}-1}\right)\Delta_n^{k,m}
\end{aligned}$$

$$
\begin{aligned}
&\leq \frac{3a_iM}{b_i\Delta_1^{k,m}}\ln(T) + \frac{3a_iM}{b_i}\sum_{n=2}^{N(k,m)}\left(\frac{1}{(\Delta_n^{k,m})^2} - \frac{1}{(\Delta_{n-1}^{k,m})^2}\right)\Delta_n^{k,m}\ln(T)\\
&= \frac{3a_iM}{b_i}\ln(T)\left[\sum_{n=1}^{N(k,m)-1}\frac{\Delta_n^{k,m}-\Delta_{n+1}^{k,m}}{(\Delta_n^{k,m})^2} + \frac{1}{\Delta_{N(k,m)}^{k,m}}\right]\\
&\leq \frac{3a_iM}{b_i}\ln(T)\left[\sum_{n=1}^{N(k,m)-1}\frac{\Delta_n^{k,m}-\Delta_{n+1}^{k,m}}{\Delta_n^{k,m}\Delta_{n+1}^{k,m}} + \frac{1}{\Delta_{N(k,m)}^{k,m}}\right]\\
&\leq \frac{3a_iM}{b_i}\ln(T)\frac{2}{\Delta_{N(k,m)}^{k,m}}.
\end{aligned}
$$

At last, we specify the choices of a_i and b_i, which resolve to the following optimization problem:

$$
\begin{aligned}
&\text{minimize } \sum_{i=1}^{\infty}\frac{6a_i}{b_i}\\
&\text{subject to } \lim_{i\to\infty}a_i = \lim_{i\to\infty}b_i = 0\\
&\qquad\text{Monotonicity: } 1 = b_0 > b_1 > b_2 > \cdots > b_i > \cdots; a_1 > a_2 > \cdots > a_i > \cdots\\
&\qquad\text{Eqn. (2.13): } \sqrt{14}\sum_{i=1}^{\infty}\frac{b_{i-1}-b_i}{\sqrt{a_i}} \leq 1.
\end{aligned}
$$

We choose a_i and b_i to be geometric sequences as in Kveton et al. (2015), specifically $a_i = d(a)^i$ and $b_i = (b)^i$ with $0 < a, b < 1$ and $d > 0$. Moreover, if $b \leq \sqrt{a}$, to meed Eqn. (2.13), it needs

$$
\sqrt{14}\sum_{i=1}^{\infty}\frac{b_{i-1}-b_i}{\sqrt{a_i}} = \sqrt{14}\sum_{i=1}^{\infty}\frac{(b)^{i-1}-(b)^i}{\sqrt{d(a)^i}} = \sqrt{\frac{14}{d}}\frac{1-b}{\sqrt{a}-b} \leq 1 \Rightarrow d \geq 14\left(\frac{1-b}{\sqrt{a}-b}\right)^2.
$$

Thus, the best choice for d is $d = 14\left(\frac{1-b}{\sqrt{a}-b}\right)^2$ and the problem is reformulated as

$$
\begin{aligned}
&\text{minimize } \sum_{i=1}^{\infty}\frac{6a_i}{b_i} = 84\left(\frac{1-b}{\sqrt{a}-b}\right)^2\frac{\alpha}{b-a}\\
&\text{conditioned on } 0 < a < b < \sqrt{a} < 1.
\end{aligned}
$$

With numerically calculated $a = 0.1459$ and $b = 0.2360$ in Kveton et al. (2015), we get $\sum_{i=1}^{\infty}\frac{6a_i}{b_i} \leq 3727$. Thus, we conclude that

$$
\begin{aligned}
\text{term (C)} &\leq \mathbb{E}\left[\sum_{(k,m)}\left[\sum_{i=1}^{\infty}\frac{6a_i}{b_i}\right]\frac{M}{\Delta_{N(k,m)}^{k,m}}\ln(T)\right]\\
&\leq \sum_{(k,m)}\frac{3727M}{\Delta_{N(k,m)}^{k,m}}\ln(T)
\end{aligned}
$$

$$\leq \sum_{(k,m)} \frac{3727M}{\Delta_{\min}^{k,m}} \ln(T).$$

Lemma 2.2.17 can be proved by combining term (C) and term (D). □

Proof for Theorem 2.2.3

Proof. This proof follows naturally from Theorem 2.2.15 by categorizing sub-optimal gaps with a threshold ϵ.

Specifically, we can modify Eqn. (2.12) as

$$\begin{aligned}
R_{e,\text{linear}}(T) &= \mathbb{E}\left[\sum_r 2^{p_r}(V_{\boldsymbol{\mu},*} - V_{\boldsymbol{\mu},S_r})\right] \\
&= \mathbb{E}\left[\sum_r 2^{p_r}\Delta_{S_r}\mathbb{1}\left\{\mathcal{E}_r, \Delta_{S_r} > 0\right\}\right] + \mathbb{E}\left[\sum_r 2^{p_r}\Delta_{S_r}\mathbb{1}\left\{\bar{\mathcal{E}}_r, \Delta_{S_r} > 0\right\}\right] \\
&\leq T\epsilon + \mathbb{E}\left[\sum_r 2^{p_r}\Delta_{S_r}\mathbb{1}\left\{\mathcal{E}_r, \Delta_{S_r} > \epsilon\right\}\right] + \mathbb{E}\left[\sum_r 2^{p_r}\Delta_{S_r}\mathbb{1}\left\{\bar{\mathcal{E}}_r, \Delta_{S_r} > \epsilon\right\}\right] \\
&\overset{(a)}{\leq} T\epsilon + \sum_{(k,m)} \frac{3727M}{\epsilon}\ln(T) + 4K^2M^2\Delta_{\max},
\end{aligned}$$

where inequality (a) follows the same proof for Lemma 2.2.17. For the overall regret, we can further get

$$\begin{aligned}
R_{\text{linear}}(T) \leq& T\epsilon + \frac{3727M^2K}{\epsilon}\ln(T) + \text{terms of order } O(\ln(T)) \text{ and independent with } \epsilon \\
\overset{(a)}{\leq}& 124M\sqrt{KT\ln(T)} + \text{terms of order } O(\ln(T)) \text{ and independent with } \epsilon \\
=& O\left(M\sqrt{KT\log(T)}\right),
\end{aligned}$$

where ϵ is taken as $62M\sqrt{\frac{K\ln(T)}{T}}$ in inequality (a). Theorem 2.2.3 is then proved. □

(α, β)-Approximation Oracle and Regret

In this section, we discuss how to extend from exact oracles to (α, β)-approximation oracles, and the corresponding performance guarantees. With the definition given in Section 2.2.3, it is straightforward to use (α, β)-approximation oracles to replace the original exact oracles in BEACON. To facilitate the discussion, we further assume that this approximation oracle always outputs collision-free matchings, which naturally holds for most of approximate optimization solvers (Vazirani, 2013).

With an (α, β)-approximation oracle, as stated in Section 2.2.3, a regret bound similar to Theorem 2.2.8 can be obtained regarding the (α, β)-approximation regret. First, the following notations are redefined and slightly abused to accommodate the (α, β)-approximation regret: $\mathcal{S}_* = \{S | S \in \mathcal{S}, V_{\boldsymbol{\mu},S} \geq \alpha V_{\boldsymbol{\mu},*}\}$: the

set of matchings with rewards larger than $\alpha V_{\boldsymbol{\mu},*}$; $\Delta_{\min}^{k,m} = \alpha V_{\boldsymbol{\mu},*} - \max\{V_{\boldsymbol{\mu},S} | S \in \mathcal{S}_b, s_m = k\}$; $\Delta_{\max}^{k,m} = \alpha V_{\boldsymbol{\mu},*} - \min\{V_{\boldsymbol{\mu},S} | S \in \mathcal{S}_b, s_m = k\}$. With these notations, BEACON's performance with an approximate oracle is established in the following.

Theorem 2.2.19 ((α, β)-approximation regret)**.** *Under Assumptions 2.2.5, 2.2.6, and 2.2.7, with an (α, β)-approximation oracle, the (α, β)-approximation regret of BEACON is upper bounded as*

$$R(T) = \tilde{O}\left(\sum_{(k,m)\in[K]\times[M]} \left[\frac{\Delta_{\min}^{k,m}}{(f^{-1}(\Delta_{\min}^{k,m}))^2} + \int_{\Delta_{\min}^{k,m}}^{\Delta_{\max}^{k,m}} \frac{1}{(f^{-1}(x))^2}\mathrm{d}x\right]\log(T) + M^2 K \Delta_c \log(T)\right).$$

Proof. The proof for Theorem 2.2.19 closely follows the proof for Theorem 2.2.8. To avoid unnecessarily redundant exposition, we here only highlight the key steps and major differences.

The communication regret and the other regret can be obtained with the same approach in the proof for Theorem 2.2.8. The main difference lies in the exploration regret. In the following proof, unless specified explicitly before, the adopted notations share the same definition as in the proof for Theorem 2.2.8. Similar to Eqn. (2.9), we can decompose the exploration regret w.r.t. the definition of the (α, β)-approximation regret as

$$\begin{aligned}
R_e(T) &= \mathbb{E}\left[\sum_r 2^{p_r}(\alpha\beta V_{\boldsymbol{\mu},*} - V_{\boldsymbol{\mu},S_r})\right] \\
&= \mathbb{E}\left[\sum_r 2^{p_r}(\alpha V_{\boldsymbol{\mu},*} - V_{\boldsymbol{\mu},S_r})\right] + \alpha(\beta-1)V_{\boldsymbol{\mu},*}\mathbb{E}[T_e] \\
&= \mathbb{E}\left[\sum_r 2^{p_r}(\alpha V_{\boldsymbol{\mu},*} - V_{\boldsymbol{\mu},S_r})(\mathbb{1}\{\mathcal{G}_r\} + \mathbb{1}\{\bar{\mathcal{G}}_r\})\right] + \alpha(\beta-1)V_{\boldsymbol{\mu},*}T_e \\
&\le \mathbb{E}\left[\sum_r 2^{p_r}(\alpha V_{\boldsymbol{\mu},*} - V_{\boldsymbol{\mu},S_r})(\mathbb{1}\{\mathcal{G}_r\} + 1 - \beta)\right] + \alpha(\beta-1)V_{\boldsymbol{\mu},*}T_e \\
&\le \mathbb{E}\left[\sum_r 2^{p_r}(\alpha V_{\boldsymbol{\mu},*} - V_{\boldsymbol{\mu},S_r})\mathbb{1}\{\mathcal{G}_r\}\right]
\end{aligned}$$

where T_e is the length of overall exploration phases. Notation $\mathcal{G}_r := \{V_{\boldsymbol{\mu},S_r} \ge \alpha V_{\boldsymbol{\mu},*}\}$ denotes the event that the oracle successfully outputs a good matching at epoch r, which happens with a probability at least β. Then, conditioned on event $\mathcal{G}_r$, the remaining analysis follows the same process in the proof for Lemma 2.2.14, and Theorem 2.2.19 can be obtained. □

2.3 Homogeneous No-sensing Model: Error-correction Coding

In this section, we turn to study the no-sensing model, where the players only perceive received rewards but not collision indicators, and for simplicity, we here only consider the homogeneous setting, which are both summarized in the following:

- **Homogeneous setting**: $\mu_{k,m} = \mu_k, \forall m \in [M], \forall k \in [M]$;
- **No-sensing model**: player m can access her own outcome $O_{s_m(t),m}(t)$ but not the corresponding no-collision indicator $\eta_{s_m(t)}(S(t))$.

For this setting, we propose the EC-SIC – *Error Correction Synchronization Involving Communication* algorithm in our paper Shi et al. (2020), whose design, analysis and evaluation are provided in the following subsections.

To ease the notations, we assume that $\mu_{(1)} \geq \mu_{(2)} \geq \cdots \geq \mu_{(K)}$. Two technical assumptions are made to faciliate the design, which are also widely used in the literature. The first is a strictly positive lower bound of μ_K as adopted in Lugosi and Mehrabian (2018); Boursier and Perchet (2019). The second assumption is a finite gap between the optimal and suboptimal (group of) arms; see Avner and Mannor (2014); Kalathil et al. (2014); Rosenski et al. (2016); Nayyar et al. (2016) for this assumption.

Assumption 2.3.1. *A positive lower bound $\mu_{\min}$ is known to all players such that $0 < \mu_{\min} \leq \mu_{(K)}$.*

Assumption 2.3.2. *There exists a positive gap $\Delta \doteq \mu_{(M)} - \mu_{(M+1)} > 0$, and it is known to all players.*

Assumption 2.3.1 implies that $\forall k \in [K]$, $\mathbb{P}(X_k > 0) \geq \mu_{\min}$, which thus bounds $\mathbb{P}(X_k = 0)$. Note that although $\mu_{\min}$ provides a lower bound for $\mu_{(K)}$, Assumption 2.3.1 does not require the exact value of $\mu_{(K)}$. The gap in Assumption 2.3.2 measures the difficulty of the bandit game and ensures the existence of only one optimal choice.

2.3.1 The EC-SIC Algorithm

Algorithm Structure and Key Ideas

As in the BEACON algorithm illustrated in Section 2.2, the EC-SIC algorithm also starts with an initialization phase, during which each player individually estimates the number of players M and assigns herself of a unique index $m \in [M]$. Then, until a player fixates on a specific arm and enters the exploitation phase, the algorithm keeps iterating between the exploration and communication phases. Players that have (not) entered the exploitation phase are called inactive (active). We denote the set of active players during the p-th phase

by $[M_p]$ and its cardinality by M_p. Similarly, arms that have not been decided to be optimal or sub-optimal are called active. The set of active arms during the p-th phase is denoted by $[K_p]$ with cardinality K_p.

Batched Exploration

During the p-th exploration phase, active players sequentially hop among the active arms for $K_p 2^p \lceil \log(T) \rceil$ steps, and any active arm is pulled $2^p \lceil \log(T) \rceil$ times by each active player. Since the hopping is based on each player's internal rank, the exploration phase is collision-free.

We note that the length of an exploration phase is different from Boursier and Perchet (2019), which is a key component of the performance improvement. The difference of a $\lceil \log(T) \rceil$ factor, in fact, results in an overall $O(\log \log(T))$ rounds of exploration and communication phases in the ADAPTED SIC-MMAB algorithm of Boursier and Perchet (2019). This directly leads to a dominating communication loss that breaks the order-optimality. With an expansion of length by $\lceil \log(T) \rceil$ in EC-SIC, the overall rounds become a constant, and the communication regret can be better controlled as shown in later analyses.

Robust Implicit Communication

In the communication phase, as in Section 2.2.1, all players attempt to exchange their sampled reward information via a a lead-follower communication scheme (i.e., player 1 as the leader and other players as followers). Also, similarly, the communication takes place via a careful collision design. All players enter this phase synchronously and, by default, keep pulling different arms based on their internal ranks. Then, when it is player i's turn to communicate with player j, she would purposely pull (not pull) player j's arm as a way to communicate bit 1 (0). If player j can fully access the collision information, i.e., knowing whether collision happens or not at each time step, she will be able to receive the bit sequence successfully, which conveys player i's sample reward statistics. However, for the no-sensing model, such error-free communication becomes impossible.

The main new ideas in the communication phase of EC-SIC compared with Boursier and Perchet (2019) is the introduction of *Z-channel coding*. In the no-sensing scenario, players cannot directly identify collision. If the same communication protocol in Boursier and Perchet (2019) (representing 1 or 0 by collision or no collision) is used, the confusion may mislead the player to believe that collision has occurred (bit 1) while it is actually a null statistic of reward sampling (bit 0). This error has a catastrophic consequence in that it breaks the essential synchronization between players. We are thus facing the challenge of communicating the reward statistics to other users while controlling the error rate for the overall communication loss to *not* dominate the regret.

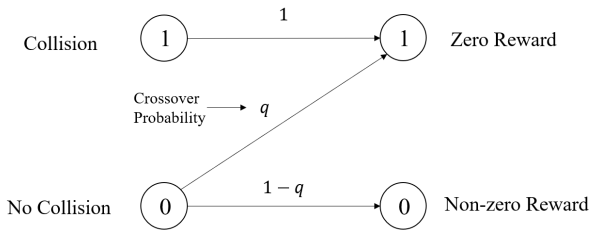


Figure 2.3: The Z-channel model for robust communication in the no-sensing setting.

Luckily, this is the well-known *reliable communication over a noisy channel* problem, one of the foundations in information theory. In particular, our communication channel is asymmetric: 1 (collision) is always received correctly and 0 (no collision) may be received incorrectly with a certain probability $P(X_k = 0)$. This corresponds to the **Z-channel** model (see Fig. 2.3) in information theory (Tallini et al., 2002), which represents a broad class of asymmetric channels. The Z-channel has a crossover probability q of $0 \to 1$ that corresponds to $P(X_k = 0)$. Since the crossover probability $P(X_k = 0)$ is unknown and varies for different arm k, $1 - \mu_{\min}$ is used to capture the worst case.

The Z-channel capacity is derived in Tallini et al. (2002) as follows.

Theorem 2.3.3. *The capacity $C_Z(q)$ of a Z-channel with crossover $0 \to 1$ probability q is:*

$$C_Z(q) = \log_2(1 + (1 - q)q^{q/(1-q)}). \tag{2.14}$$

Shannon theory guarantees that as long as the coding rate R is below the above capacity $C_Z(q)$, there exists at least one code that allows for an arbitrarily low error rate asymptotically. This means that theoretically, for this Z-channel, it is possible to transmit information nearly error-free when the rate is close to $C_Z(q)$ bits per channel use. In reality, however, different finite block-length channel codes may have different performances; we thus evaluate several practical codes both theoretically (in Section 2.3.2) and experimentally (in Section 2.3.3). For simplicity, Functions `Send()`, `Receive()`, `Encoder()` and `Decoder()` are used in the algorithm as the sending and receiving protocol and the encoder and corresponding decoder, respectively.

2.3.2 Regret Analysis

The overall regret of EC-SIC can be decomposed as $R(T) = R^{init} + R^{expl} + R^{comm}$. The first, second and third term refers to the regret caused by the initialization, exploration, and communication phase, respectively, and the overall main result is presented in Theorem 2.3.4, and each component regret is subsequently analyzed.

Algorithm 2 The EC-SIC Algorithm

Require: $T,\ K,\ \Delta,\ \epsilon,\ \mu_{\min}$;

Initialize $p \leftarrow 1$; $F \leftarrow -1$; $T_0, T_0^j \leftarrow 0$; $[K_p] \leftarrow [K]$; $Q \leftarrow \max\{\lceil \log_2 \frac{1}{\frac{\Delta}{4}-\epsilon} \rceil, \lceil \log_2(K+1) \rceil\}$; $T_c \leftarrow \lceil \frac{\log(T)}{\mu_{\min}} \rceil$
Select an error-correction code (N', Q) with code length N' defined in Theorem 2.3.4
Initialization Phase
$k \leftarrow$ Musical_Chair($[K]$, KT_c)
$(M,\ j) \leftarrow$ Estimate_M_NoSensing(k, T_c)
while $F = -1$ **do**
Exploration Phase
    $\pi \leftarrow j$-th active arm
    for $K_p 2^p \lceil \log(T) \rceil$ time steps **do**
        $\pi \leftarrow \pi + 1 \pmod{K_p}$ and play arm π
        $s[\pi] \leftarrow s[\pi] + r^j(t)$
    end for
    $T_p^j = T_{p-1}^j + 2^p \lceil \log(T) \rceil$
    $\hat{\mu}_j = s/T_p^j$
Communication Phase
    if $j = 1$ **then**
        $(F,\ M_{p+1},\ [K_{p+1}]) \leftarrow$ Communication Leader$(\hat{\mu}_1,\ p,\ [K_p],\ M_p,\ Q,\ N')$
    else
        $(F,\ M_{p+1},\ [K_{p+1}]) \leftarrow$ Communication Follower$(\hat{\mu}_j,\ j,\ p,\ [K_p],\ M_p,\ Q,\ N')$
    end if
    $p \leftarrow p + 1$
end while
`Exploitation phase`
Pull F until T

Theorem 2.3.4. *With an optimal coding technique that achieves Gallager's error exponent $E(\mu_{min})$ for the corresponding Z-channel with crossover probability $1 - \mu_{min}$, for any $\epsilon \in (0, \frac{\Delta}{4})$, we have*

$$R(T) \leq MK\frac{\log(T)}{\mu_{\min}} + \frac{\Delta}{\epsilon}\sum_{k>M}\frac{\log(T)}{\mu_{(M+1)} - \mu_{(k)} + 4\epsilon} + N'\left(M^2(K+2)\log\left(\frac{1}{4\epsilon}\right) + M^2K\right) \tag{2.15}$$

where

$$N' = \max\left\{\frac{Q}{C_Z(1-\mu_{\min})}, \frac{1}{E(\mu_{min})}\log(T)\right\}.$$

Theorem 2.3.4 involves an information-theoretic concept called *error exponent*, which is explained in Theorem 2.3.7 but more details can be found in (Gallager, 1968).

An asymptotic upper bound can be obtained from (2.15) with $\epsilon = \frac{\Delta}{8}$:

$$R(T) = O\left(\sum_{k>M}\frac{\log(T)}{\mu_{(M)} - \mu_{(k)}} + (\frac{M^2K\log(\frac{1}{\Delta})}{E(\mu_{\min})} + \frac{MK}{\mu_{\min}})\log(T)\right). \tag{2.16}$$

Compared to SIC-MMAB2, we have successfully removed the multiplicative factor of M in the first $\log(T)$

Algorithm 3 Communication Leader

Require: $\hat{\mu}_1,\ p,\ [K_p],\ M_p,\ Q,\ N'$
Ensure: $F,\ M_{p+1},\ [K_{p+1}]$
Initialize $T_p = T_{p-1} + M_p 2^p \lceil \log(T) \rceil$; $T_p^i = T_p^1,\ i = 1, ..., M_p$; $\bar{\mu}_i^p = \bar{\mu}_i^{p-1},\ T_p^i = T_{p-1}^i,\ i = M_p + 1, ..., M$
Gather information from followers
for $i = 2, ..., M_p$ **do** ▷ `Receive arm statistics`
  for $k \in [K_p]$ **do**
    $\bar{\mu}_i^p[k] \leftarrow$ Decoder(Receive(1, i, N'))
  end for
end for
$\bar{\mu}^p = \sum_{i=1}^{M} \bar{\mu}_i^p \cdot T_p^i / T_p$; $B_{T_p} = \sqrt{\frac{2\log(T)}{T_p}} + (\frac{\Delta}{4} - \epsilon)$
Update statistics
Rej $\leftarrow$ set of active arms k satisfying $|\{i \in [K_p] | \bar{\mu}^p[i] - B_{T_p} \geq \bar{\mu}^p[k] + B_{T_p}\}| \geq M_p$
Acc $\leftarrow$ set of active arms k satisfying $|\{i \in [K_p] | \bar{\mu}^p[k] - B_{T_p} \geq \bar{\mu}^p[i] + B_{T_p}\}| \geq K_p - M_p$, ordered according to their indices
Transmit acc\rej arms to followers
for $i = 2, ..., M_p$ **do** ▷ `Send acc\rej set size`
  Send(1, i, N', Encoder(|Rej|, Q))
  Send(1, i, N', Encoder(|Acc|, Q))
end for
for $i = 2, ..., M_p$ **do** ▷ `Send acc\rej set content`
  Send(1, i, N', Encoder(k, Q)) for $k \in$ Rej
  Send(1, i, N', Encoder(k, Q)) for $k \in$ Acc
end for
if $M_p \leq$ |Acc| **then**
  $F \leftarrow$ Acc$[M_p]$
else $M_p \leftarrow M_p -$ |Acc|
  $[K_{p+1}] \leftarrow [K_p] \backslash$ (Acc $\cup$ Rej)
end if

term. This is due to the efficient communication phase that transmits the reward statistics. In addition, we have a M^2K factor in the second $\log(T)$ term, as opposed to MK^2 in SIC-MMAB2. This is also an improvement since $M < K$.

To prove Theorem 2.3.4, we first define the "typical event" as the success of initialization, communication and exploration throughout the entire horizon T. More specifically, we define three events: $A_1 = \{$each player has a correct estimation of M and an orthogonal internal rank after initialization$\}$; $A_2 = \{$messages are decoded correctly in all communication phases$\}$; $A_3 = \{|\bar{\mu}^p[k] - \mu[k]| \leq B_{T_p}$ holds for phase $p, \forall k \in [K_p], \forall p\}$. We use P_s to denote the probability that the typical event happens, which is $A_1 \cap A_2 \cap A_3$. The regret caused by the "atypical event" can be simply bounded by a linear regret $O(MT)$. Then the result of (2.15) can be proved by controlling P_s to balance both events.

Initialization phase

Similar to Lemma 11 in Boursier and Perchet (2019), we can bound the regret of initialization as follows.

Algorithm 4 Communication Follower

Require: $\hat{\mu}_j,\ j,\ p,\ [K_p],\ M_p,\ Q,\ N'$
Ensure: $F,\ M_{p+1},\ [K_{p+1}]$
Transmit information to the leader
for $i = 2, ..., M_p$ **do** ▷ `Send arm statisitcs`
  if $j = i$ **then**
    Send(j, 1, N', Encoder($\hat{\mu}_j[k]$, Q)) for $k \in [K_p]$
  else pull the j-th active arm for $K_p N'$ steps
  end if
end for
Receive acc\rej arms from the leader:
for $i = 2, ..., M_p$ **do** ▷ `Receive acc\rej set size`
  if $j = i$ **then**
    $N_{\text{rej}} \leftarrow$ Decoder(Receive(j, 1, N'))
    $N_{\text{acc}} \leftarrow$ Decoder(Receive(j, 1, N'))
  else pull j-th active arm for $2N'$ steps
  end if
end for
for $i = 2, ..., M_p$ **do** ▷ `Receive acc\rej set content`
  if $j = i$ **then**
    $\boldsymbol{w}[k] \leftarrow$Receive$(j, 1, N'))$ and
    Rej$[k] \leftarrow$ Decoder$(\boldsymbol{w}[k])$ for $k = 1, ..., N_{\text{rej}}$
    $\boldsymbol{w}[k] \leftarrow$Receive$(j, 1, N'))$ and
    Acc$[k] \leftarrow$ Decoder$(\boldsymbol{w}[k])$ for $k = 1, ..., N_{\text{acc}}$
  else pull j-th active arm for $(N_{\text{rej}} + N_{\text{acc}})N'$ steps
  end if
end for
if $M_p - j + 1 \leq |\text{Acc}|$ **then**
  $F \leftarrow \text{Acc}[M_p - j + 1]$
else $M_p \leftarrow M_p - |\text{Acc}|$
  $[K_{p+1}] \leftarrow [K_p] \backslash (\text{Acc} \cup \text{Rej})$
end if

Lemma 2.3.5. *With probability $P_i = 1 - O(\frac{MK}{T})$, event A_1 happens. Furthermore, the regret of the initialization phase satisfies:*

$$R^{init} < 3MK \left\lceil \frac{\log(T)}{\mu_{\min}} \right\rceil.$$

Exploration phase

The regret due to exploration is bounded in the following lemma.

Lemma 2.3.6. *With probability $P_s = 1 - O(\frac{MK\log(T)}{T})$, the typical event happens and the exploration regret conditioned on the typical event satisfies:*

$$R^{expl} = O\left(\frac{\Delta}{4\epsilon} \sum_{k>M} \min\left\{ \frac{\log(T)}{\mu_{(M+1)} - \mu_{(k)} + 4\epsilon}, \sqrt{T\log(T)} \right\} \right).$$

We first present a fundamental result of channel coding for communication in a noisy channel, known as the *error exponent* (Gallager, 1968).

Theorem 2.3.7. *For a discrete memory-less channel, if $R < C$, there exists a code of block length N without feedback such that the error probability is bounded by*

$$P_e \leq \exp[-NE_r(R)],$$

where $E_r(R)$ is the random coding error exponent with rate R.

We note that the error exponent used in Theorem 2.3.4 corresponds to $E(\mu_{\min}) = E_r(C_Z(1-\mu_{\min}))$.

Theorem 2.3.7 suggests that, to transmit a Q-bit message over a Z-channel, there exists an optimal coding scheme with length $N' = \max\{\frac{Q}{C_Z(1-\mu_{\min})}, N\}$ to achieve an error rate less than $\frac{1}{T}$, where $N = \frac{1}{E(\mu_{\min})}\log(T)$. Several of the existing coding techniques, although not optimal, can achieve this error rate with $N = \Theta(\log(T))$, which only leads to a multiplicative factor larger than $\frac{1}{E(\mu_{\min})}$ but does not change the regret order. For example, with repetition code, flip code and modified Hamming code, we have $N_{rep} = Q\lceil\frac{\log(QT)}{\mu_{\min}}\rceil$, $N_{flip} = Q\lceil\frac{\log(QT/2)}{\mu_{\min}}\rceil$, $N_{ham} = \frac{7Q}{8}\lceil\frac{\log(7QT/8)}{\mu_{\min}}\rceil$ respectively. The remaining analysis will be based on the optimal channel coding with the caveat that a "good" Z-channel code should be applied in practice.

With at most $\log(T)$ exploration and communication phases and K arms to be accepted or rejected, there are at most $MK\log(T)$ communication instances on arm statistics, $2M\log(T)$ communication instances on the number of acc/rej arms, and KM communication instances on the index of acc/rej arms. A simple union bound analysis leads to the following result.

Lemma 2.3.8. *Denoting the probability that event A_2 holds by P_r, with an optimal Z-channel code of $N' = \max\{\frac{Q}{C_Z(1-\mu_{\min})}, \frac{\log(T)}{E(\mu_{\min})}\}$, we have*

$$P_r = 1 - O\left(\frac{MK\log(T)}{T}\right).$$

Lemma 2.3.8 guarantees all communications are correct. To bound the probability that all arms are correctly estimated, we have the following result.

Lemma 2.3.9. *In phase p, for any active arm $k \in [K_p]$,*

$$P\left\{|\bar{\mu}^p[k] - \mu[k]| \geq B_{T_p}\right\} \leq \frac{2}{T}.$$

With at most $\log(T)$ exploration-communication phases, event A_3 happens with probability:

$$P_c = 1 - O\left(\frac{K\log(T)}{T}\right). \tag{2.17}$$

A union bound argument leveraging P_i, P_r and P_c leads to probability P_s for the typical event to happen, as defined in Lemma 2.3.6. Finally, for the exploration phases, the number of times that an arm is pulled before being accepted or rejected are well controlled.

Lemma 2.3.10. *In the typical event, every optimal arm is accepted after at most* $O\left(\frac{\log(T)}{(\mu_{(k)}-\mu_{(M)}+4\epsilon)^2}\right)$ *pulls, and every sub-optimal arm is rejected after at most* $O\left(\frac{\log(T)}{(\mu_{(M+1)}-\mu_{(k)}+4\epsilon)^2}\right)$ *pulls.*

Denote T^{expl} as the overall time of exploration and exploitation phase and $T_{(k)}^{expl}(T)$ as the number of time steps where the k-th best arm is pulled during these two phases. With no collision in exploration and exploitation, the exploration regret can be decomposed as (Anantharam et al., 1987)

$$\begin{aligned} R^{expl} = &\sum_{k>M}(\mu_{(M)}-\mu_{(k)})T_{(k)}^{expl}(T) \\ &+\sum_{k\leq M}(\mu_{(k)}-\mu_{(M)})(T^{expl}-T_{(k)}^{expl}(T)), \end{aligned} \tag{2.18}$$

Both components in Eqn. (2.18) can be upper bounded further bounded, which proves Lemma 2.3.6.

Communication phase

Thanks to the expanded length of each exploration phase and the fixed-length quantization of arm statistics, the regret R^{comm} does not dominate the overall regret, as stated in the following lemma.

Lemma 2.3.11. *In the typical event,*

$$R^{comm} = O\left(N'\left(M^2(K+2)\log\left(\min\left\{\frac{1}{4\epsilon},T\right\}\right)+M^2K\right)\right).$$

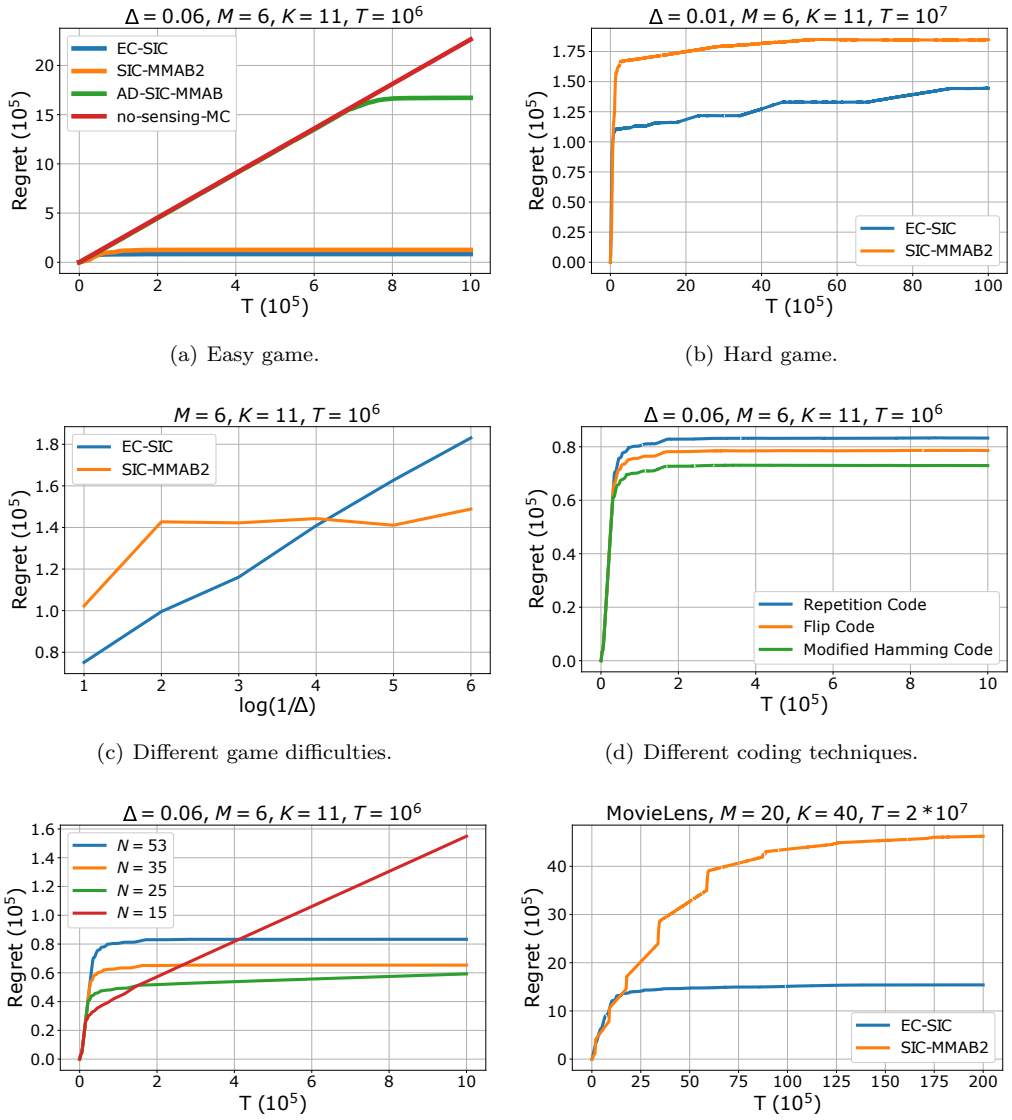
We note that $\log(\min\{\frac{1}{4\epsilon},T\})$ becomes a constant when T is sufficiently large. Noting that $N' = \max\{\frac{Q}{C_Z(1-\mu_{\min})},\frac{\log(T)}{E(\mu_{\min})}\}$, the communication loss has the same order as other phases.

Overall regret

When the typical event happens, the overall regret is bounded by the sum of R^{init}, R^{comm} and R^{expl}; otherwise, for the atypical event, the regret can be upper bounded as MT. Thus, the overall regret satisfies

$$R(T) \leq R^{init} + R^{expl} + R^{comm} + O(M^2K\log(T)).$$

With Lemmas 2.3.5, 2.3.6 and 2.3.11, Theorem 2.3.4 can be proven.



(a) Easy game.

(b) Hard game.

(c) Different game difficulties.

(d) Different coding techniques.

(e) Different codeword lengths.

(f) The MovieLens dataset.

Figure 2.4: Regret comparisons between EC-SIC and other MP-MAB algorithms. (a) is evaluated with one easy game instance, (b) is evaluated with one hard game instance, (c), (d), and (e) reflects the regret changes with different game difficulties, coding techniques, and codeword length, and (f) is evaluated with the MovieLens dataset.

2.3.3 Experimental Results

Numerical experiments have been carried out to verify the analysis of EC-SIC and compare its empirical performance to other methods. All rewards follow the Bernoulli distributions with $\mu_{\min} = 0.3$, and we set $\epsilon = \Delta/8$. Results are obtained by averaging over 500 experiments.

We compare state-of-the-art algorithms under both easy and difficult bandit game settings. EC-SIC (with repetition code), ADAPTED SIC-MMAB, SIC-MMAB2, and the algorithm proposed by Lugosi and Mehrabian (2018) (labeled as "no-sensing-MC") are first compared in a relatively easy game ($\Delta = 0.06$). Fig. 2.4 shows that even in an easy game, no-sensing-MC could not finish exploration within 10^6 time steps, and ADAPTED SIC-MMAB has poor performance compared to the other two. Both EC-SIC and SIC-MMAB2 converge to the optimal arm set quickly, but the overall regret of EC-SIC is smaller. For a hard game with $\Delta = 0.01$, Fig. 2.4 shows that EC-SIC is superior to SIC-MMAB2.

A detailed comparison of EC-SIC with SIC-MMAB2 is done by comparing their regrets as a function of the gap Δ in Fig. 2.4. We see that when the game is not extremely difficult ($\Delta > 10^{-4}$), EC-SIC has better performance since players benefit from sharing statistics. When Δ becomes extremely small, the required communication length increases significantly, leading to a dominating communication regret in EC-SIC that cannot be offset by the benefits of sharing statistics.

Fig. 2.4 reports the performance while using different Z-channel codes in communication. We observe that modified Hamming Code has the best performance, which is due to its superior error correction capability. This observation also implies that with a near-optimal code that is specifically designed for Z-channel, performance of EC-SIC can be further improved.

We also evaluate the impact of codeword length on the regret. For our simulation setting, the theoretical analysis requires a repetition code length $N = 53$ to transmit one bit, in order to achieve an error rate of $\frac{1}{T}$. We are interested in evaluating whether the theoretically required code length can be shortened in practice. Under the easy game setting of Fig. 2.4 with 2000 rounds averaging, Fig. 2.4 shows that with N decreasing from 53 to 35, the regret decreases 20%. More importantly, it shows that the convergence of EC-SIC does not change. When further reducing N to 25, we see the regret curve trends upward at large t, which represents a non-negligible loss due to unsuccessful communications. With $N = 15$, the regret increases rapidly, indicating that players suffer from an increased error rate. It is thus essential to strike a balance between error rate and communication loss.

Lastly, we evaluate EC-SIC on a real world dataset: the movie watching dataset (ml-20m) from MovieLens (Harper and Konstan, 2015). It consists of watching data of more than 2×10^4 movies from over 10^5 users between January 09, 1995 and March 31, 2015. In the pre-processing, we group these movies into $K = 40$ categories by their total number of views from high to low. The binary reward at time t (hour) is defined as whether there are users watching films in this group, and we replicate it to a final reward sequence of length $T = 2 \times 10^7$. $M = 20$ players are assumed to engage in the game. This final sequence has $\Delta \approx 0.007$ and $\mu_{\min} \approx 0.6$. Compared to synthetic datasets, this setting poses a larger and more difficult game. For each experiment, the reward sequence is randomly shuffled. We report the cumulative regret of EC-SIC and

SIC-MMAB2, averaged over 100 experiments, in Figure 2.4. One can see that the advantage of EC-SIC over SIC-MMAB2 is significant for this real-world dataset. Intuitively, this is because the game is hard ($\Delta \approx 0.007$), and M and K are also large.

2.3.4 Full Proofs

Initialization phase

The initialization phase starts with a "Muscial Chair" phase, which assigns a unique external rank in $1, ..., K$ for each of the player. Then the following sequential hopping protocol converts the external rank into a unique internal rank in $1, ..., M$ for each player and estimates the number of players M. The proof for Lemma 2.3.5 is the same as Lemma 11 in Boursier and Perchet (2019).

Exploration phase

This section aims at proving Lemma 2.3.6, which bounds the exploration regret. We start with the required lemmas and then go back to proving Lemma 2.3.6.

First, Lemma 2.3.8 ensures that event A_2 happens with a high probability. As mentioned before, there are at most $\log_2(T)$ communication phases, which lead to at most $(MK + 2M)\log_2(T)$ instances of transmissions to send arm statistics to the leader and send the acc/rej arm sets to the followers. Since there are at most K arms to be accepted or rejected, no more than MK instances of transmissions are required for sending the acc/rej arm sets.

Proof of Lemma 2.3.8. Denote $P(\xi_p)$ as the probability that the decoding of a Q-bit message produces a wrong result at round p. With the choice of $N' = \max\{\frac{Q}{C_Z(1-\mu_{\min})}, \frac{\log(T)}{E(R)}\}$, and X_p, Y_p denoting the message before encoding and after decoding at round p, we have

$$P(\xi_p) = P(X_p \neq Y_p) \leq \frac{1}{T}.$$

A simple union bound leads to

$$P_r = 1 - P(\cup p, \xi_p) \geq 1 - \sum_p P(\xi_p) \geq 1 - \frac{(MK + 2M)\log(T) + MK}{T} = 1 - O\left(\frac{MK\log(T)}{T}\right).$$

□

Then, Lemma 2.3.9 ensures the acceptance and rejection of arms are successful with a high probability, which requires a good estimation of the statistics of arms. The estimation error consists of two parts: the quantization error and the sampling error. We analyze them separately.

Proof of Lemma 2.3.9. With the choice of $Q \geq \log_2(\frac{1}{\frac{\Delta}{4}-\epsilon})$, the quantization error in phase p can be bounded as:

$$\begin{aligned}
&\forall i \in [M], |\bar{\mu}_i^p[k] - \hat{\mu}_i^p[k]| \leq \frac{\Delta}{4} - \epsilon, \\
&|\bar{\mu}^p[k] - \hat{\mu}^p[k]| = \frac{\left|\sum_{i=1}^M (\bar{\mu}_i^p[k] - \hat{\mu}_i^p[k]) T_p^i\right|}{T_p} \leq \frac{\Delta}{4} - \epsilon.
\end{aligned}$$

For any active arm $k \in [K_p]$, the gap between the sample mean $\hat{\mu}^p[k]$ (using all players' samples) and the true mean can be bounded with Hoeffding's inequality:

$$P\left(|\hat{\mu}^p[k] - \mu[k]| \geq \sqrt{\frac{2\log(T)}{T_p}}\right) \leq \frac{2}{T}.$$

Then, the overall gap between the quantized mean and the true mean for any active arm $k \in [K_p]$ can be bounded as:

$$\begin{aligned}
&P\left(|\bar{\mu}^p[k] - \mu[k]| > B_{T_p}\right) \\
=&P\left(|\bar{\mu}^p[k] - \hat{\mu}^p[k] + \hat{\mu}^p[k] - \mu[k]| \geq \sqrt{\frac{2\log(T)}{T_p}} + \frac{\Delta}{4} - \epsilon\right) \\
\leq&P\left(|\bar{\mu}^p[k] - \hat{\mu}^p[k]| + |\hat{\mu}^p[k] - \mu[k]| \geq \sqrt{\frac{2\log(T)}{T_p}} + \frac{\Delta}{4} - \epsilon\right) \\
\leq&P\left(|\hat{\mu}^p[k] - \mu[k]| \geq \sqrt{\frac{2\log(T)}{T_p}}) \cup P(|\bar{\mu}^p[k] - \hat{\mu}^p[k]| \geq \frac{\Delta}{4} - \epsilon\right) \\
=&P\left(|\hat{\mu}^p[k] - \mu[k]| \geq \sqrt{\frac{2\log(T)}{T_p}}\right) \\
\leq&\frac{2}{T}.
\end{aligned}$$

There are at most $\log_2(T)$ iterations of exploration and communication. By using a union bound of all these iterations and K arms, Eqn. (2.17) is obtained. □

Lemma 2.3.10 bounds the number of time steps an arm is pulled before being accepted or rejected and is essential to control the rounds of exploration and communication. The proof is similar to the proof of Proposition 1 in Boursier and Perchet (2019).

Proof of Lemma 2.3.10. The proof is conditioned on the typical event. We first consider an optimal arm k. Let $\Delta_k = \mu[k] - \mu_{(M+1)}$ be the gap between the arm k and the first sub-optimal arm. Let s_k be the first integer such that $4B_{s_k} \leq \Delta_k$. It satisfies:

$$s_k \geq \frac{32\log(T)}{(\Delta_k - \Delta + 4\epsilon)^2} = \frac{32\log(T)}{\big(\mu[k] - \mu_{(M)} + 4\epsilon\big)^2}.$$

Recall that the number of time steps an active arm is pulled before the p-th exploration is $T_p = \sum_{l=1}^{p} M_l 2^l \lceil \log(T) \rceil$. With a non-increasing M_p, it holds that

$$T_{p+1} \leq 3T_p. \tag{2.19}$$

For some p such that $T_{p-1} \leq s_k < T_p$, the following inequalities are in order: $\Delta_k \geq 4B_{T_p}$; $|\bar{\mu}^p[k] - \mu[k]| \leq B_{T_p}$; and $|\bar{\mu}^p[i] - \mu[i]| \leq B_{T_p}$ for all sub-optimal arm i.

We then have

$$\bar{\mu}^p[k] - B_{T_p} \geq \bar{\mu}^p[i] + B_{T_p} + \mu[k] - \mu[i] - 4B_{T_p} \geq \bar{\mu}^p[i] + B_{T_p}.$$

This suggests arm k will be accepted at time T_p. Eqn. (2.19) also leads to $T_p = O(s_k) = O\left(\frac{\log(T)}{(\mu[k]-\mu_{(M)}+4\epsilon)^2}\right)$. Thus, arm k will be accepted after at most $O\left(\frac{\log(T)}{(\mu[k]-\mu_{(M)}+4\epsilon)^2}\right)$ pulls. The part of rejecting sub-optimal arms can be similarly proved with $\Delta_k = \mu_{(M)} - \mu[k]$. □

Lemma 2.3.12. *In the typical event, the following results hold.*

1) for any sub-optimal arm k, $(\mu_{(M)} - \mu[k])T_k^{expl}(T) = O\left(\frac{\Delta}{4\epsilon}\min\left\{\frac{\log(T)}{\mu_{(M+1)} - \mu[k] + 4\epsilon}, \sqrt{T\log(T)}\right\}\right)$;

2) $\sum_{k\leq M}(\mu_{(k)} - \mu_{(M)})(T^{expl} - T_{(k)}^{expl}) = O\left(\sum_{k>M}\min\left\{\frac{\log(T)}{\mu_{(M+1)} - \mu_{(k)} + 4\epsilon}, \sqrt{T\log(T)}\right\}\right)$.

The proof of the first part in Lemma 2.3.12 is as follows.

Proof. For a sub-optimal arm k, Lemma 2.3.12 leads to $T_k^{expl}(T) \leq O\left(\min\left\{\frac{\log(T)}{(\mu_{(M+1)}-\mu[k]+4\epsilon)^2}, T\right\}\right)$, and thus

$$\begin{aligned}
(\mu_{(M)} - \mu[k])T_k^{expl}(T) &= \frac{\mu_{(M)} - \mu[k]}{\mu_{(M+1)} - \mu[k] + 4\epsilon} O\left(\min\left\{\frac{\log(T)}{(\mu_{(M+1)} - \mu[k] + 4\epsilon)}, (\mu_{(M+1)} - \mu[k] + 4\epsilon)T\right\}\right) \\
&\overset{(i)}{\leq} O\left(\frac{\Delta}{4\epsilon}\min\left\{\frac{\log(T)}{\delta}, \delta T\right\}\right) \\
&\overset{(ii)}{\leq} O\left(\frac{\Delta}{4\epsilon}\min\left\{\frac{\log(T)}{(\mu_{(M+1)} - \mu[k] + 4\epsilon)}, \sqrt{T\log(T)}\right\}\right),
\end{aligned}$$

in which inequality (i) comes from

$$\frac{\mu_{(M)}-\mu[k]}{\mu_{(M+1)}-\mu[k]+4\epsilon}=\frac{\mu_{(M)}-\mu[k]}{\mu_{(M)}-\mu[k]+4\epsilon-\Delta}\leq\frac{\Delta}{4\epsilon}$$

and $\delta=\mu_{(M+1)}-\mu[k]+4\epsilon$. Inequality (ii) can be obtained with the observation that the term $\frac{\Delta}{4\epsilon}O(\min\{\frac{\log(T)}{\delta},\delta T\})$ is maximized by $\delta=\sqrt{\frac{\log(T)}{T}}$. □

The second part of Lemma 2.3.12 is based on Lemmas 2.3.13 and 2.3.14.

Lemma 2.3.13. *Define $\hat{t}_k$ as the number of exploratory pulls before accepting/rejecting the arm k and H is the total number of exploration phases. Conditioned on the typical event, we have:*

$$\sum_{k\leq M}\left(\mu_{(k)}-\mu_{(M)}\right)\left(T^{expl}-T^{expl}_{(k)}\right)\leq\sum_{j>M}\sum_{k\leq M}\sum_{p=1}^{H}2^p\lceil\log(T)\rceil\left(\mu_{(k)}-\mu_{(M)}\right)\mathbb{1}_{\min\{\hat{t}_{(j)},\hat{t}_{(k)}\}\geq T_{p-1}}.$$

Proof. For an optimal arm k, during phase p, if k has already been accepted, it will be pulled $K_p2^p\lceil\log(T)\rceil$ times. If it is still active (i.e., $\hat{t}_k>T_{p-1}$), it will be pulled $M_p2^p\lceil\log(T)\rceil$ times, meaning that this arm is not pulled for $(K_p-M_p)2^p\lceil\log(T)\rceil$ times. Thus, it holds that $T^{expl}_k\geq T^{expl}-\sum_{p=1}^{H}2^p(K_p-M_p)\lceil\log(T)\rceil\mathbb{1}_{\hat{t}_k>T_{p-1}}$. Notice that $K_p-M_p=\sum_{j>M}\mathbb{1}_{\hat{t}_{(j)}>T_{p-1}}$. We have $T^{expl}_k\geq T^{expl}-\sum_{p=1}^{H}\sum_{j>M}2^p\lceil\log(T)\rceil\mathbb{1}_{\min\{\hat{t}_{(j)},\hat{t}_{(k)}\}>T_{p-1}}$, which proves the lemma. □

Lemma 2.3.14. *Conditioned on the typical event, we have:*

$$\sum_{k\leq M}\sum_{p=1}^{H}2^p\lceil\log(T)\rceil\left(\mu_{(k)}-\mu_{(M)}\right)\mathbb{1}_{\min\{\hat{t}_{(j)},\hat{t}_{(k)}\}\geq T_{p-1}}\leq O\left(\min\left\{\frac{\log(T)}{\mu_{(M)}-\mu_{(j)}+4\epsilon},\sqrt{T\log(T)}\right\}\right).$$

Proof. Define $A_j=\sum_{k\leq M}\sum_{p=1}^{H}2^p\lceil\log(T)\rceil(\mu_{(k)}-\mu_{(M)})\mathbb{1}_{\min\{\hat{t}_{(j)},\hat{t}_{(k)}\}\geq T_{p-1}}$. Notice that

$$\hat{t}_{(k)}\leq\min\left\{\frac{c\log(T)}{(\mu_{(k)}-\mu_{(M)}+4\epsilon)^2},T\right\},$$

and denote $\Delta(p)=\sqrt{\frac{c\log(T)}{T_{p-1}}}$. The inequity $\hat{t}_{(k)}>T_{p-1}$ implies $\mu_{(k)}-\mu_{(M)}<\Delta(p)-4\epsilon$. We also denote N^j as the smallest integer satisfying $\hat{t}_{(j)}\leq T_{N^j}$. Then we have the following:

$$\begin{aligned}A_j&\leq\sum_{k\leq M}\sum_{p=1}^{N^j}2^p\lceil\log(T)\rceil(\Delta(p)-4\epsilon)\mathbb{1}_{\hat{t}_{(k)}\geq T_{p-1}}\\&\leq\sum_{p=1}^{N^j}\Delta(p)2^p\lceil\log(T)\rceil\sum_{k\leq M}\mathbb{1}_{\hat{t}_{(k)}\geq T_{p-1}}\end{aligned}$$

$$
\begin{aligned}
&= \sum_{p=1}^{N^j} \Delta(p) 2^p \lceil \log(T) \rceil M_p \\
&\leq \sum_{p=1}^{N^j} \Delta(p)(T_p - T_{p-1}) \\
&= c \log(T) \sum_{p=1}^{N^j} \Delta(p) \left(\frac{1}{\Delta(p+1)} + \frac{1}{\Delta(p)} \right) \left(\frac{1}{\Delta(p+1)} - \frac{1}{\Delta(p)} \right).
\end{aligned}
$$

Since $T_{p+1} \leq 3T_p$, $\Delta(p) \left(\frac{1}{\Delta(p+1)} + \frac{1}{\Delta(p)} \right) = 1 + \sqrt{\frac{T_p}{T_{p-1}}} \leq 1 + \sqrt{3}$. Thus,

$$
A_j \leq c \log(T) \sum_{p=1}^{N^j} \left(\frac{1}{\Delta(p+1)} - \frac{1}{\Delta(p)} \right) \leq (1 + \sqrt{3}) c \log(T) \frac{1}{\Delta(N^j + 1)}.
$$

With the definition of N^j, we have $\hat{t}_{(j)} \geq T_{N^j - 1}$. With inequality $T_{N^j+1} \leq 3T_{N^j}$ we have $\Delta(N^j) \geq \sqrt{\frac{c \log(T)}{3\hat{t}_{(j)}}}$. $A_j \leq (3 + \sqrt{3}) \sqrt{c \hat{t}_{(j)} \log(T)}$ then holds. With $\hat{t}_{(j)} \leq O\left(\min\{ \frac{c \log(T)}{(\mu_{(M+1)} - \mu_{(j)} + 4\epsilon)^2}, T \} \right)$, we have

$$
A_j \leq (3 + \sqrt{3}) \min \left\{ \frac{c \log(T)}{\mu_{(M+1)} - \mu_{(j)} + 4\epsilon}, \sqrt{cT \log(T)} \right\}.
$$

□

Communication phase

This section presents the proof related to the bound of the communication regret.

Proof for Lemma 2.3.11. Conditioned on the typical event, we denote H as the number of exploration phases. The communication length for sending arm statistics and acc/rej arm sets for $p \in [H]$ is at most $N'(KM+2M)$. Lemma 2.3.10 states that H satisfies $T_H = \sum_{l=1}^{H} M_l 2^l \lceil \log(T) \rceil = O(\max_{k \in [K]} \{s_k\}) = O\left(\min\{ \frac{\log(T)}{4\epsilon}, T \} \right)$. Thus,

$$
H = O\left(\log(\min\{ \frac{1}{4\epsilon}, T \}) \right),
$$

which leads to a regret of $O(N'(KM^2 + 2M^2) \log(\min\{ \frac{1}{4\epsilon}, T \})$. Next, transmitting acc/rej arm sets at most incurs a regret of $M^2 K N'$. Putting them together, the total communication regret is:

$$
O\left(N'(KM^2 + 2M^2) \log\left(\min\left\{ \frac{1}{4\epsilon}, T \right\} \right) + N' M^2 K \right).
$$

□

Chapter 3

Collaboration Designs for Multi-agent Decision Making

Federated learning (FL) (McMahan et al., 2017) is a new distributed machine learning (ML) paradigm that addresses new challenges in modern machine learning (ML). In particular, FL handles distributed ML with the following characteristics: non-IID local datasets, communication efficiency, and privacy. While the state-of-the-art FL largely focuses on the supervised learning setting, we propose to extend the core principles of FL to the multi-armed bandits (MAB) problem.

3.1 Federated Multi-armed Bandits: Different Relationships between Global and Local Models

3.1.1 The Approximate Model

In this section, we present a framework of federated multi-armed bandits (FMAB) as illustrated in Fig. 3.1, which is based on our work of Shi and Shen (2021a).

Clients and The Server

Multiple clients interact with the same set of K arms (referred to as "local arms") in the FMAB framework. We denote M_t as the number of participating clients at time t, who are labeled from 1 to M_t to facilitate discussions (they are not used in the algorithms). A client can only interact with her own local MAB model, and there is no direct communication between clients. Arm k generates independent *observations* $X_{k,m}$ for

client m following a σ-subgaussian distribution with mean $\mu_{k,m}$. Note that $X_{k,m}$ is only an observation but not a reward. For different clients $n \neq m$, their models are non-IID; hence $\mu_{k,n} \neq \mu_{k,m}$ in general.

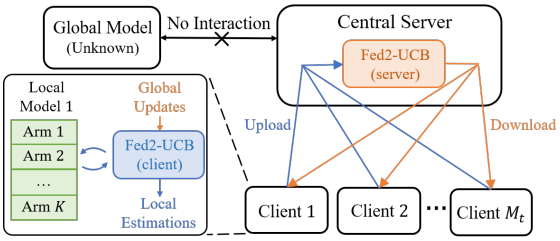


Figure 3.1: The FMAB framework.

There exists a central server with a global stochastic MAB model, which has the same set of K arms (referred to as "global arms") of σ-subgaussian reward distributions with mean reward μ_k for arm k. The true rewards for this system are generated on this global model, thus the learning objective is on the global arms. However, the server cannot directly observe rewards on the global model; she can only interact with clients who feed back information of their local observations. We consider the general non-IID situation where the local models are not necessarily the same as the global model and also make the common assumption that clients and the server are fully synchronized.

Although clients cannot communicate with each other, after a certain time, they can transmit local "model updates" based on their local observations to the server, which aggregates these updates to have a more accurate estimation of the global model. The new estimation is then sent back to the clients to replace the previous estimation for future actions. However, just like in FL, the communication resource is a major bottleneck and the algorithm has to be conscious about its usage. We incorporate this constraint in FMAB by imposing a loss C every time a client communicates to the server, which will be accounted for in the performance measure defined below.

The Approximated Model

Although the non-IID property of local models is an important feature of FMAB, there must exist some relationship between local and global models so that observations on local bandit models help the server learn the global model. Here, we propose the approximate FMAB model, where the global model is a fixed (but hidden) ground truth (i.e., exogenously generated regardless of the participating clients), and the local models are IID random realizations of it.

Specifically, the global arm k has a fixed mean reward of μ_k. For client m, the mean reward $\mu_{k,m}$ of her local arm k is a sample from an unknown distribution ϕ_k, which is a σ_c-subgaussian distribution with mean μ_k. For a different client $n \neq m$, $\mu_{k,n}$ is sampled IID from ϕ_k. Since local models are stochastic realizations of the

global model, a *finite* collection of the former may not necessarily represent the latter. In other words, if there are M involving clients, although $\forall m \in [M], \mathbb{E}[\mu_{k,m}] = \mu_k$, the averaged local model $\hat{\mu}_k^M \doteq \frac{1}{M}\sum_{m=1}^{M}\mu_{k,m}$ may not be consistent with the global model. Specifically, $\hat{\mu}_k^M$ is not necessarily equal (or even close) to μ_k, which introduces significant difficulties. Intuitively, the server needs to sample sufficiently many clients to have a statistically accurate estimation of the global model, but as we show later, the required number of clients cannot be obtained *a priori* without the suboptimality gap knowledge. The need of client sampling also coincides with the property of massively distributed clients in FL.

Motivation Example

The approximate model captures the key characteristics of a practical cognitive radio system, as illustrated in Fig. 3.2. Assume a total of K candidate channels, indexed by $\{1, ..., K\}$. Each channel's availability is location-dependent, with $p_k(x)$ denoting the probability that channel k is available at location x. The goal of the base station is to choose one channel out of K candidates to serve all potential cellular users (e.g., control channel) in the given coverage area $\mathcal{D}$ with area D. Assuming users are uniformly randomly distributed over $\mathcal{D}$, the global channel availability is measured throughout the entire coverage area as

$$p_k = \mathbb{E}_{x\sim u(\mathcal{D})}\left[p_k(x)\right] = \iint_{\mathcal{D}} \frac{1}{D} p_k(x)dx. \tag{3.1}$$

It is well known in wireless research that a base station cannot directly sample p_k by itself, because it is fixed at one location. In addition, Eqn. (3.1) requires a *continuous* sampling throughout the coverage area, which is not possible in practice. Realistically, the base station can only direct cellular user m at *discrete* location x_m to estimate $p_k(x_m)$, and then aggregate observations from finite number of users as $\hat{p}_k = \frac{1}{M}\sum_{m=1}^{M} p_k(x_m)$ to approximate p_k. Clearly, even if $p_k(x_m)$ are perfect, $\hat{p}_k$ may not necessarily represent p_k well.

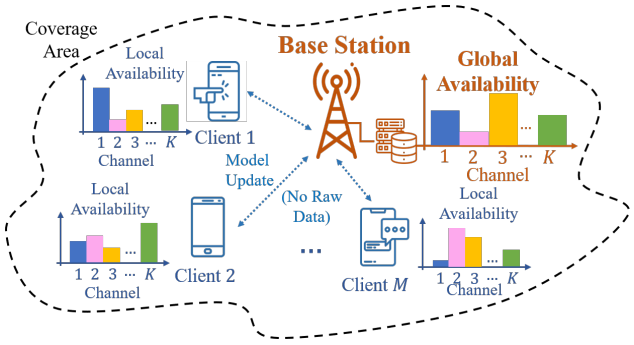


Figure 3.2: The Motivation Example of Cognitive Radio for FMAB.

Regret Definition

Without loss of generality, we assume there is only one optimal global arm k_* with $\mu_* \doteq \mu_{k_*} = \max_{k\in[K]} \mu_k$, and $\Delta = \mu_* - \max_{k\neq k_*}\{\mu_k\}$ denotes the suboptimality gap of the global model (both unknown to the algorithm). We further denote $\gamma_1, \cdots, \gamma_{T_c}$ as the time slots when the clients communicate with the central server for both upload and download. The notion of regret in for the single-player model can be generalized to all the clients with additional communication loss, as follows:

$$R(T) = \mathbb{E}\Bigg[\underbrace{\sum_{t\in[T]}^{T} M_t X_{k_*}(t) - \sum_{t\in[T]}\sum_{m\in[M_t]} X_{\pi_m(t)}(t)}_{\text{exploration and exploitation}} + \underbrace{\sum_{\tau\in[T_c]} CM_{\gamma_\tau}}_{\text{communication}}\Bigg], \tag{3.2}$$

where $\pi_m(t)$ is the arm chosen by client m at time t. In this work, we aim to design algorithms with $O(\log(T))$ regret as in the single-player setting.

Several comments are in place for Eqn. (3.2). First, the cumulative reward of the system is defined on the global model, because clients only receive *observations* from playing the local bandit game, and the *reward* is generated at the system-level global model. Taking the cognitive radio system as an example, the choice by each client only produces her observation of the channel availability, but the reward is generated by the base station when this channel is used for the entire coverage area. Also, the regret definition discourages the algorithm from involving too many clients. Ideally, only sufficiently many clients should be admitted to accurately reconstruct the global model, and any more clients would result in more communication loss without improving the model learning.

3.1.2 The Fed2-UCB Algorithm

Challenges and Main Ideas

The first and foremost challenge in the approximate model comes from that the local models are only stochastic realizations of the global model. Even with the perfect information on all local arms, the optimal global arm may not be produced faithfully. We refer to this new problem as the *uncertainty from client sampling*. How to simultaneously handle the two types of uncertainty (client sampling and arm sampling) is at the center of solving the approximate model.

A second issue comes from the conflict between non-IID local models and the global model. In particular, the globally optimal arm may be sub-optimal for a client's local model, and hence it cannot be correctly inferred by the client individually. Communication between clients and the server is key to addressing this conflict, but the challenge is how to control the communication loss and balance the overall regret.

In this section, we first characterize the uncertainty from client sampling by analyzing the probability that the averaged local model does not faithfully represent the global model, and illustrating that without knowledge of the suboptimality gap Δ, the algorithm cannot determine *a priori* the number of required clients. Then, Federated Double UCB (Fed2-UCB) is proposed, in which a novel "double UCB" principle carefully balances and trades off the two sources of uncertainty while controlling the communication cost.

Client Sampling

In the approximate model, the key to determining whether the local knowledge is sufficient lies in whether the optimal global arm can be inferred correctly. When there are M involving clients, the best approximate of the global model is the averaged local model, i.e., $\hat{\mu}_k^M$. Although the utilities of local arms may be different from the global model, if the true optimal global arm is still optimal in this averaged local model, i.e., $\hat{\mu}_{k_*}^M > \max_{k \neq k_*} \hat{\mu}_k^M$, a sub-linear regret can be achieved with local knowledge. Otherwise, arm k_* is not optimal with respect to $\hat{\mu}_k^M$, and no matter how many explorations are performed locally (even with perfect local knowledge), the global optimal arm cannot be found using the sampled M local models and thus a linear regret occurs.

The following theorem characterizes the accuracy of representing the global model by the averaged local model from a fixed number of clients.

Theorem 3.1.1. *With M involved clients, denote $P_z = \mathbb{P}\left(\hat{\mu}_{k_*}^M \leq \max_{k \in [K]} \hat{\mu}_k^M\right)$, the following result holds:*

$$P_z = O\left(\sum_{k \neq k_*} \exp\left\{-\sigma_c^{-2} M(\mu_* - \mu_k)^2\right\}\right) = O\left(K \exp\left\{-\sigma_c^{-2} M \Delta^2\right\}\right).$$

Theorem 3.1.1 indicates that the probability that the averaged local model does not represent the global model, i.e., $\hat{\mu}_{k_*}^M \leq \max_{k \in [K]} \hat{\mu}_k^M$, decreases exponentially with respect to the number of involved clients M. Thus, it is fundamental to involve a sufficiently large number of clients in order to reconstruct the global model correctly. More specifically, to guarantee that $P_z = O(1/T)$, by which the overall regret can scale sub-linearly, it is sufficient to sample M clients with

$$M = \Omega\left(\sigma_c^2 \Delta^{-2} \log(KT)\right). \tag{3.3}$$

If Eqn. (3.3) is satisfied throughout the bandit game, the optimal arm can be successfully found. However, clients do not have access to the knowledge of Δ. Thus, the requirement in Eqn. (3.3) cannot be guaranteed in advance.

On the other hand, involving too many clients may be detrimental to the regret. Specifically, in order to have an $O(\log(T))$ regret, M should satisfy:

$$M = O\left(\log(T)\right). \tag{3.4}$$

Comparing Eqns. (3.3) and (3.4) suggests that M has to be $\Theta(\log(T))$ to achieve a correct representation of the global model while maintaining an $O(\log(T))$ regret.

Algorithm Design

With the unknown requirement in Eqn. (3.3), it is unwise to only admit a small number of clients in the whole game. On the other hand, Eqn. (3.4) prohibits involving too many clients to achieve an $O(\log(T))$ regret. There are also practical system considerations that prevent having too many clients, which has been discussed in the context of FL (Bonawitz et al., 2019). We propose the Fed2-UCB algorithm where the central server gradually admits new clients into the game after each communication round while keeping local clients gathering observations. The method of gradually increasing the clients ensures that the server samples a set of small but sufficiently representative clients based on the underlying statistical structure of the bandit game. The proposed "double UCB" principle simultaneously addresses the uncertainty from both client sampling and arm sampling.

Algorithm 5 Fed2-UCB: client m

Initialize $p \leftarrow 1$; $[K_1] \leftarrow [K]$
while $K_p > 1$ **do**
  Pull each active arm $k \in [K_p]$ for $f(p)$ times
  Calculate the local sample means $\bar{\mu}_{k,m}(p), \forall k \in [K_p]$
  Send local updates $\bar{\mu}_{k,m}(p), \forall k \in [K_p]$ to the server
  Receive global update set E_p from the server
  $[K_{p+1}] \leftarrow [K_p] \backslash E_p$; $p \leftarrow p + 1$
end while
$F \leftarrow$the only element in $[K_p]$; Stay on arm F until T

Algorithm 6 Fed2-UCB: central server

Initialize $p \leftarrow 1$; $[K_1] \leftarrow [K]$
while $K_p > 1$ **do**
  Admit $g(p)$ new clients ▷ *Client sampling*
  Receive local updates $\bar{\mu}_{k,m}(p), \forall k \in [K_p], \forall m \in [M(p)]$
  Calculate $\forall k \in [K_p], \bar{\mu}_k(p) \leftarrow \sum_{m=1}^{M(p)} \bar{\mu}_{k,m}(p)/M(p)$
  $E_p \leftarrow \left\{k \in [K_p] | \bar{\mu}_k(p) + B_{p,2} \leq \max_{l \in [K_p]} \bar{\mu}_l(p) - B_{p,2}\right\}$
  Send global update set E_p to all involved clients
  $[K_{p+1}] \leftarrow [K_p] \backslash E_p$; $p \leftarrow p + 1$
end while

The Fed2-UCB algorithm is performed in phases simultaneously and synchronously at clients and the central server. Clients collect observations and update local estimations for the arms that have not been declared as sub-optimal, i.e., the active arms, while the server admits new clients and aggregates the local estimations as global estimations to eliminate sub-optimal active arms. We denote the set of active arms in the p-th phase by $[K_p]$ with cardinality K_p. The detailed algorithms for the clients and the central server are given in Algorithms 5 and 6, respectively.

At phase p, $g(p)$ new clients are first added into the game by the server. These clients can be viewed as interacting with newly sampled local MAB models. Each client, regardless of newly added or not, performs a sequential arm sampling among the currently active arms for $K_p f(p)$ times on their own local models, which means each active arm is pulled $f(p)$ times by each client. Thus, arm $k \in [K_p]$ is played a total of $M(p)f(p)$ times in phase p, where $M(p) = \sum_{q=1}^{p} g(q)$ is the overall number of clients at phase p. Parameters $g(p)$ and $f(p)$ are flexible and we discuss the impact of these choices on the regret in the next section. It is worth noting that the rate of admitting new clients is determined not only by $g(p)$ but also by $f(p)$, which characterizes the frequency of client sampling. With new observations from arm sampling, each client m updates her local estimations, i.e., sample mean $\bar{\mu}_{k,m}(p), k \in [K_p]$, then sends them to the central server as a local parameter update. Note that uploading sample means instead of raw samples benefits the preservation of privacy, and additional methods for better privacy protection are presented in the supplementary material.

After receiving local parameter updates from the clients, the central server first updates the global estimation as the average of them for each active arm, i.e., $\bar{\mu}_k(p) = \frac{1}{M(p)} \sum_{m=1}^{M(p)} \bar{\mu}_{k,m}(p), k \in [K_p]$. While recognizing two coexisting uncertainties, a "double" confidence bound $B_{p,2}$ is adopted to characterize them simultaneously as:

$$B_{p,2} = \underbrace{\sqrt{6\sigma^2 \eta_p \log(T)}}_{\text{arm sampling}} + \underbrace{\sqrt{6\sigma_c^2 \log(T)/M(p)}}_{\text{client sampling}},$$

where $\eta_p = \frac{1}{M(p)^2} \sum_{q=1}^{p} \frac{g(q)}{F(p)-F(q-1)}$ and $F(p) = \sum_{q=1}^{p} f(q)$ with $F(0) = 0$. The first terms in $B_{p,2}$ characterizes the uncertainty from arm sampling, which illustrates the gap between the averaged sampled local model and the exact averaged local model. The second term represents the uncertainty from client sampling, which captures the gap between the exact averaged local model and the (hidden) global model. Note that these two types of uncertainty are not independent of each other, since more admitted clients can perform more pulls on arms, thus reducing both simultaneously.

With the global estimations and the confidence bound, the elimination set E_p is determined by the server, which contains arms that are sub-optimal with a high probability:

$$E_p = \left\{ k \in [K_p] \middle| \bar{\mu}_k(p) + B_{p,2} \leq \max_{l \in [K_p]} \bar{\mu}_l(p) - B_{p,2} \right\}.$$

The set $[E_p]$ is then sent back to the clients, who then remove these arms from their sets of active arms. This iteration keeps going until there is only one active arm left.

Regret Analysis

The regret of the Fed2-UCB algorithm is the combination of the exploration loss and communication loss and relies on the design of $g(p)$ and $f(p)$.

Theorem 3.1.2. *For $k \neq k_*$, we denote $\Delta_k = \mu_* - \mu_k$ and p_k as the smallest integer p such that*

$$96 \left(\sigma \sqrt{\eta_p} + \sigma_c / \sqrt{M(p)} \right)^2 \log(T) \leq \Delta_k^2, \tag{3.5}$$

and $p_{\max} = \max_{k \neq k_*} \{p_k\}$. If $\max_{t \leq T} \{M_t\} \leq \beta T$, where β is a constant, the regret for the Fed2-UCB algorithm satisfies

$$R_2(T) \leq \sum_{k \neq k_*} \sum_{q=1}^{p_k} \Delta_k M(q) f(q) + C \sum_{q=1}^{p_{\max}} M(q) + 4\beta(1+C)K.$$

Eqn. (3.5) describes the requirement for phase p_k under two types of uncertainty, by which the sub-optimal arm k is guaranteed to be eliminated with a high probability. For it to hold, eventually we need at least $O(\log(T))$ clients in the game, which coincides with Eqn. (3.3).

Theorem 3.1.2 provides a general description, using unspecified $g(p)$ and $f(p)$. A better characterization can be had with more specific choices.

Corollary 3.1.3. *With $f(p) = \kappa$ where κ is a constant, and $g(p) = 2^p$, the asymptotic regret of Fed2-UCB is*

$$R_2(T) = O\left(\sum_{k \neq k_*} \frac{\kappa (\sigma/\sqrt{\kappa} + \sigma_c)^2 \log(T)}{\Delta_k} + C \frac{(\sigma/\sqrt{\kappa} + \sigma_c)^2 \log(T)}{\Delta^2} \right).$$

Corollary 3.1.3 shows that with carefully designed $f(p) = \kappa$ and $g(p) = 2^p$, Fed2-UCB can achieve a regret of $O(\log(T))$. The exploration loss approaches the single-player MAB lower bound (Lai and Robbins, 1985), which shows the effectiveness of exploration in Fed2-UCB. Since at least $O(\log(T))$ clients need to be involved as indicated by Eqn. (3.3), an $O(\log(T))$ communication loss achieved in Corollary 3.1.3 is inevitable, which

demonstrates the communication efficiency. The overall regret in Corollary 3.1.3 proves that Fed2-UCB can effectively deal with two types of uncertainty while balancing the communication loss.

The choice of $g(p) = 2^p$ and $f(p) = \kappa$ leads to an exponentially decreasing $B_{p,2}$, which can be viewed as maintaining an exponentially decreasing estimation $\hat{\Delta}$ of Δ and eliminating arms with a larger gap (Auer and Ortner, 2010); thus, it naturally solves the difficulty associated with the unknown Δ. The regret behavior of several other choices of $f(p)$ and $g(p)$ are given in the supplementary material.

3.1.3 The Exact Model and The Fed1-UCB Algorithm

While the approximate model introduces two types of uncertainty simultaneously, here we study a special case of the *exact model*, where the uncertainty from client sampling does not exist. Correspondingly, the Fed1-UCB algorithm, which degenerates from Fed2-UCB, is designed and analyzed.

The Exact Model

In the exact model, the number of clients is fixed, i.e., $M_t = M$, $\forall t$, and the global model is the *exact* average of all the local models, which means the global arm k has a mean reward of $\mu_k = \frac{1}{M}\sum_{m=1}^{M}\mu_{k,m}$. Thus, the global model can be perfectly reconstructed with information of local models and there only exists the uncertainty from arm sampling. The regret expression can be simplified to

$$R(T) = \mathbb{E}\left[\sum_{t\in[T]} MX_{k_*}(t) - \sum_{t\in[T]}\sum_{m\in[M]} X_{\pi_m(t)}(t) + CMT_c\right].$$

This model focuses on optimizing the performance for a fixed group of clients that do not change throughout the T time steps. In other words, the global model is not exogenously generated but adapts to the involved clients. Taken the recommender system as an example, the overall popularity of one item is the average of its popularity over the potential clients.

The Fed1-UCB Algorithm

Without the uncertainty from client sampling, there is no need of admitting new clients. The same exploration and communication procedure of Fed2-UCB is performed in Fed1-UCB without client admitting. The confidence bound used in arm eliminations is also degenerated from $B_{p,2}$ to $B_{p,1} = \sqrt{6\sigma^2\log(T)/(MF(p))}$, which only characterizes the uncertainty from arm sampling. A complete description of Fed1-UCB is given in the supplementary material.

Regret Analysis

The regret for the Fed1-UCB algorithm only relies on $f(p)$ and is characterized by the following theorem.

Theorem 3.1.4. *For $k \neq k_*$, we denote $\Delta_k = \mu_* - \mu_k$, $F(p) = \sum_{q=1}^{p} f(q)$, p_k as the smallest integer p such that*

$$MF(p) \geq 96\sigma^2 \log(T)/\Delta_k^2, \tag{3.6}$$

and $p_{\max} = \max_{k \neq k_*}\{p_k\}$. The regret of Fed1-UCB satisfies

$$R_1(T) \leq M \sum_{k \neq k_*} \Delta_k F(p_k) + CMp_{\max} + 2(1+C)MK.$$

Somewhat surprisingly, Eqn. (3.6) shows that although involving more clients leads to a faster convergence (i.e., smaller p_k), in general, the overall necessary arm pulls performed by the clients, i.e., $MF(p_k)$, are independent of M. In other words, we can trade off the convergence time with the number of clients without additional exploration loss.

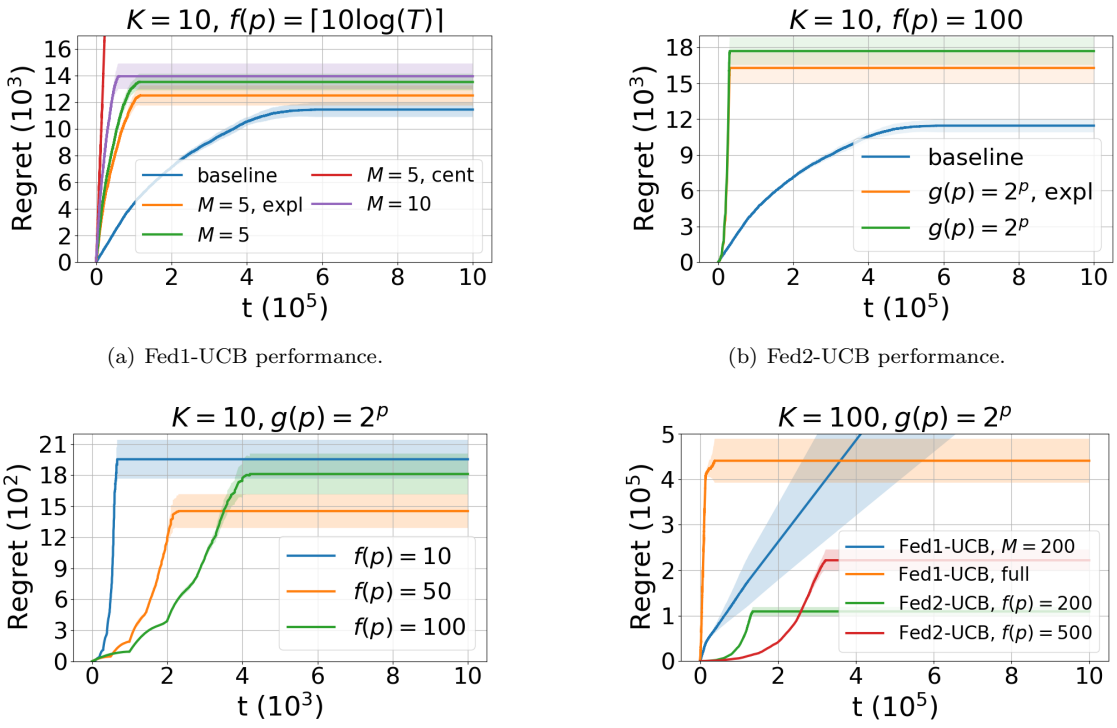
Corollary 3.1.5. *With $f(p) = \lceil \kappa \log(T) \rceil$ where κ is a constant, the asymptotic regret of the Fed1-UCB algorithm is*

$$R_1(T) = O\left(\sum_{k \neq k_*} \frac{\sigma^2 \log(T)}{\Delta_k} \right).$$

Corollary 3.1.5 states that the exploration loss of Fed1-UCB approaches the single-player MAB lower bound (Lai and Robbins, 1985). It is also worth noting that with $f(p) = \lceil \kappa \log(T) \rceil$, the communication loss of Fed-1UCB is a non-dominating constant, which demonstrates its communication efficiency. Furthermore, the regret is independent of M asymptotically. The regret behavior with other choices of $f(p)$ is discussed in the supplementary material.

3.1.4 Experimental Results

Numerical experiments have been carried out under both applications of cognitive radio and recommender systems. Their results are reported in this section to demonstrate the effectiveness and efficiency of Fed2-UCB and Fed1-UCB. For the cognitive radio example, due to the lack of suitable real-world datasets, synthetic datasets are used for simulations (Avner and Mannor, 2014; Bande and Veeravalli, 2019). For the recommender system, real-world evaluations are performed. The performance of a (hypothetical) single-player improved UCB algorithm (Auer and Ortner, 2010) directly performed at the server is used as the baseline (labeled as "baseline"). The communication cost is set to be $C = 1$.



(a) Fed1-UCB performance.

(b) Fed2-UCB performance.

(c) Fed2-UCB with different $f(p)$.

(d) Performance with MovieLens dataset.

Figure 3.3: Regret comparisons between FMAB algorithms. The continuous curves represent the empirical average values, and the shadowed areas represent the standard deviations. (a), (b) and (c) are evaluated with synthetic datasets, and (d) is from the MovieLens dataset.

Synthetic Dataset for Cognitive Radio

A bandit game with $K = 10$ arms is used to mimic 10 candidate channels, and Gaussian distributions with $\sigma = 0.5$ are used to generate local observations of the channel availability. The means of global arms are in the interval $[0.7, 0.8]$ with $\Delta = 0.02$. We first start with the relatively simple exact model, where $M = 5$ clients are involved while arm 1 is not the optimal arm of any of their local models. As shown in Fig. 3.3, with $f(p) = \lceil 10\log(T) \rceil$, if there is no communication loss, Fed1-UCB (labeled as "expl") achieves almost the same performance as the baseline, which proves its effectiveness. When considering the communication loss, the centralized version of Fed1-UCB (labeled as "cent"), where clients send their raw data in every time slot, has a very large regret due to significant yet unnecessary communications. However, with $f(p) = \lceil 10\log(T) \rceil$, Fed1-UCB only incurs a small communication loss, which proves its efficiency. It is also worth noting that Fed1-UCB converges faster than the baseline, which is the result of higher arm sampling rate due to multiple clients simultaneously pulling arms. In other words, the fast convergence over time is due to the increased client dimension. When increasing the number of clients to $M = 10$, the overall regret remains approximately the same as $M = 5$, but with even faster convergence, which corroborates Theorem 3.1.4 and Corollary 3.1.5.

For the approximate model, the same set of global arms is used while the local models are generated by Gaussian distributions with $\sigma_c = 0.02$. Fig. 3.3(b) shows that Fed2-UCB with $f(p) = 100$ and $g(p) = 2^p$ successfully finds the optimal global arm and, without communication loss, has a performance (labeled as "expl") slightly worse than the baseline. Furthermore, the additional communication loss is very limited. Compared with the performance of Fed1-UCB in Fig. 3.3(a), we see that Fed2-UCB achieves almost the same performance for the more challenging approximate model, and the convergence of Fed2-UCB is even faster since the impact of increasing the number of clients is already significant at the very beginning. Under a reduced time horizon $T = 10^4$, Fig. 3.3(c) provides a finer look at the shape of regret curves of Fed2-UCB and illustrates the need to balance two types of uncertainty. With a short update period $f(p) = 10$, new clients are admitted rapidly, which sharply decreases the uncertainty from client sampling, but insufficient local exploration leads to a large uncertainty from arm sampling, which causes a large regret despite the fast convergence. On the other extreme, although local exploration is guaranteed to be sufficient with $f(p) = 100$, it admits new clients slowly, which delays the convergence and causes unnecessary local explorations. $f(p) = 50$ strikes a better balance between two types of uncertainty and thus a better performance.

Real-world Dataset for Recommender System

The MovieLens dataset (Cantador et al., 2011) is used for real-world evaluation as an implementation of a recommender system, which has been widely adopted in MAB studies. It links the movies of the MovieLens dataset with IMDb and Rotten Tomatoes movie review systems and contains 2113 clients and 10197 movies. All the users are assumed to be available while the movies are randomly divided into 100 groups and the observations for clients are defined as their ratings of each group of movies. The suboptimality gap of the pre-processed data is $\Delta \approx 0.0053$. The number of arms and potential clients is much larger than the synthetic dataset. First, as shown in Fig. 3.3(d), if a small fraction of clients ($M = 200$) are used for Fed1-UCB, which can be viewed as only involving a small number of clients at the beginning of Fed2-UCB, the regret curve trends upward, meaning the global optimal arm is not found due to insufficient client sampling. Oppositely, when all clients are involved, Fed1-UCB converges to the optimal arm but with a large regret, which shows the harm of oversampling. Using Fed2-UCB and $f(p) = 200$ with $g(p) = 2^p$, much better performance is achieved since only the necessary amount of clients are sampled to capture the global model faithfully without unnecessary loss. With $f(p) = 500$, new clients are admitted more slowly but it still outperforms Fed1-UCB.

3.1.5 Additional Theoretical Discussions

The regrets in Theorem 3.1.2 and Theorem 3.1.4 are related to the choices at the server, i.e., $f(p)$ and $g(p)$. In this section, we provide a more detailed discussion of the impact of these choices on regret.

Discussion for Theorem 3.1.2

From Eqns. (3.3), (3.4) and (3.5), there are $\Theta(\log(T))$ clients involved in the game eventually, which means that the choice of any $f(p)$ with an order higher than $O(1)$ cannot have an $O(\log(T))$ regret. Also, with $O(\log(T))$ involved clients, a constant communication loss is no longer achievable as shown in Corollary 3.1.5. Thus, we focus on choices of $g(p)$ while fixing $f(p) = \kappa$, and the results are given in Table 3.1. With a linear growth rate $g(p) = \lambda$, we can see that the overall regret is of order $O(\log^2(T))$. While increasing the rate to $g(p) = \lceil \lambda \log(T) \rceil$, an $O(\log(T))$ regret is achieved but the multiplicative factors are far from optimal. By exponentially increasing involving players with $g(p) = 2^p$ or $g(p) = \lceil 2^p \log(T) \rceil$, an exploration loss approaching the lower bound can be achieved, while the communication loss remains sublinear of order $O(\log(T))$.

Table 3.1: Regret of Fed2-UCB algorithm with $f(p) = \kappa$ and different choices of $g(p)$.

$g(p)$	p_k, $k \neq k_*$	$R_2(T)$
λ	$\left\lceil \frac{96(\sigma/\sqrt{\kappa}+\sigma_c)^2 \log(T)}{\lambda(\mu_*-\mu_k)^2} \right\rceil$	$O\left(\sum_{k\neq k_*} \frac{\kappa(\sigma/\sqrt{\kappa}+\sigma_c)^4 \log^2(T)}{\lambda(\mu_*-\mu_k)^3} + C\frac{(\sigma/\sqrt{\kappa}+\sigma_c)^4 \log^2(T)}{\lambda\Delta^4}\right)$
$\lceil \lambda \log(T) \rceil$	$\left\lceil \frac{96(\sigma/\sqrt{\kappa}+\sigma_c)^2}{\lambda(\mu_*-\mu_k)^2} \right\rceil$	$O\left(\sum_{k\neq k_*} \frac{\kappa(\sigma/\sqrt{\kappa}+\sigma_c)^4 \log(T)}{\lambda(\mu_*-\mu_k)^3} + C\frac{(\sigma/\sqrt{\kappa}+\sigma_c)^4 \log(T)}{\Delta^4}\right)$
2^p	$\left\lceil \log\left(\frac{96(\sigma/\sqrt{\kappa}+\sigma_c)^2 \log(T)}{(\mu_*-\mu_k)^2}\right) \right\rceil$	$O\left(\sum_{k\neq k_*} \frac{\kappa(\sigma/\sqrt{\kappa}+\sigma_c)^2 \log(T)}{(\mu_*-\mu_k)} + C\frac{(\sigma/\sqrt{\kappa}+\sigma_c)^2 \log(T)}{\Delta^2}\right)$
$\lceil 2^p \log(T) \rceil$	$\left\lceil \log\left(\frac{96(\sigma/\sqrt{\kappa}+\sigma_c)^2}{(\mu_*-\mu_k)^2}\right) \right\rceil$	$O\left(\sum_{k\neq k_*} \frac{\kappa(\sigma/\sqrt{\kappa}+\sigma_c)^2 \log(T)}{(\mu_*-\mu_k)} + C\frac{(\sigma/\sqrt{\kappa}+\sigma_c)^2 \log(T)}{\Delta^2}\right)$

λ and κ are constants and $\Delta = \min_{k\neq k_*}\{\mu_* - \mu_k\}$ is the suboptimality gap; the p_k column represents its upper bound.

Discussion for Theorem 3.1.4

For Theorem 3.1.4, a few possible choices for $f(p)$ with the corresponding p_k and asymptotic regrets are given in Table 3.2. While $f(p) = \kappa$, the overall asymptotic regret is independent of M; however, the communication loss is of order $O(\log(T))$. The choice of $f(p) = \lceil \kappa \log(T) \rceil$ results in a constant communication loss which scales as $1/\Delta^2$. When the update period grows exponentially, the communication loss is of order $O(\log(\log(T)))$ for $f(p) = 2^p$ and $O(1)$ for $f(p) = \lceil 2^p \log(T) \rceil$, and with these two choices, the communication loss now scales as $\log(1/\Delta)$.

Table 3.2: Regret of Fed1-UCB algorithm with different choices of $f(p)$.

$f(p)$	p_k, $k \neq k_*$	$R_1(T)$
κ	$\left\lceil \frac{96\sigma^2 \log(T)}{\kappa M(\mu_* - \mu_k)^2} \right\rceil$	$O\left(\sum_{k\neq k_*} \frac{\sigma^2 \log(T)}{(\mu_* - \mu_k)} + C\frac{\sigma^2 \log(T)}{\kappa \Delta^2}\right)$
$\lceil \kappa \log(T) \rceil$	$\left\lceil \frac{96\sigma^2}{\kappa M(\mu_* - \mu_k)^2} \right\rceil$	$O\left(\sum_{k\neq k_*} \frac{\sigma^2 \log(T)}{(\mu_* - \mu_k)} + C\frac{\sigma^2}{\kappa \Delta^2}\right)$
2^p	$\left\lceil \log\left(\frac{96\sigma^2 \log(T)}{M(\mu_* - \mu_k)^2}\right) \right\rceil$	$O\left(\sum_{k\neq k_*} \frac{\sigma^2 \log(T)}{(\mu_* - \mu_k)} + CM\log\left(\frac{\sigma^2 \log(T)}{M\Delta^2}\right)\right)$
$\lceil 2^p \log(T) \rceil$	$\left\lceil \log\left(\frac{96\sigma^2}{M(\mu_* - \mu_k)^2}\right) \right\rceil$	$O\left(\sum_{k\neq k_*} \frac{\sigma^2 \log(T)}{(\mu_* - \mu_k)} + CM\log\left(\frac{\sigma^2}{M\Delta^2}\right)\right)$

κ is a constant and $\Delta = \min_{k\neq k_*}\{\mu_* - \mu_k\}$ is the suboptimality gap;
the p_k column represents its upper bound.

3.1.6 Full Proofs

Proof of Theorem 3.1.1

Proof. With a union bound, we have

$$P_z = P\left(\hat{\mu}_{k_*}^M \leq \max_{k\neq k_*} \hat{\mu}_k^M\right) = P\left(\bigcup_{k\neq k_*} \left(\hat{\mu}_{k_*}^M \leq \hat{\mu}_k^M\right)\right) \leq \sum_{k\neq k_*} P\left(\hat{\mu}_{k_*}^M \leq \hat{\mu}_k^M\right). \tag{3.7}$$

For a given arm $k \neq k_*$, we further have

$$\begin{aligned}
P\left(\hat{\mu}_{k_*}^M > \hat{\mu}_k^M\right) &\geq P\left(\hat{\mu}_{k_*}^M \geq \frac{1}{2}(\mu_k + \mu_*) \geq \hat{\mu}_k^M\right) \\
&= P\left(\hat{\mu}_{k_*}^M \geq \frac{1}{2}(\mu_k + \mu_*)\right) P\left(\frac{1}{2}(\mu_k + \mu_*) \geq \hat{\mu}_k^M\right) \\
&\overset{(i)}{\geq} \left(1 - \exp\left\{\frac{-M(\mu_* - \mu_k)^2}{8\sigma_c^2}\right\}\right)\left(1 - \exp\left\{\frac{-M(\mu_* - \mu_k)^2}{8\sigma_c^2}\right\}\right) \\
&= 1 - O\left(\exp\left\{\frac{-M(\mu_* - \mu_k)^2}{\sigma_c^2}\right\}\right).
\end{aligned}$$

Inequality (i) is because $\hat{\mu}_{k_*}^M$ and $\hat{\mu}_k^M$ are $\frac{\sigma_c}{\sqrt{M}}$-subgaussian random variables. Thus, each term in the summation of Eqn. (3.7) can be bounded as

$$P\left(\hat{\mu}_{k_*}^M \leq \hat{\mu}_k^M\right) \leq O\left(\exp\left\{-\frac{M(\mu_* - \mu_k)^2}{\sigma_c^2}\right\}\right).$$

Finally Theorem 3.1.1 can be derived as

$$P_z \leq \sum_{k\neq k_*} P\left(\hat{\mu}_{k_*}^M \leq \hat{\mu}_k^M\right) \leq O\left(\sum_{k\neq k_*} \exp\left\{-\frac{M(\mu_* - \mu_k)^2}{\sigma_c^2}\right\}\right) \leq O\left(K\exp\left\{-\frac{M\Delta^2}{\sigma_c^2}\right\}\right).$$

□

Proof of Theorem 3.1.2

Step 1: Confidence Bound for the Estimations

We first analyze the probability guarantee of the interval for the averaged local mean estimation. In the Fed2-UCB algorithm, with two types of uncertainty, the following lemma provides an upper bound for the gap between averaged local means and the exact global means for each arm.

Lemma 3.1.6. *At phase p, for any active arm $k \in [K_p]$, it holds that*

$$P\left(|\bar{\mu}_k(p) - \mu_k| \geq B_{p,2}\right) \leq \frac{4}{T^3}.$$

Proof. The gap between $\bar{\mu}_k(p)$ and μ_k can be bounded as follows:

$$\begin{aligned}
&P\left(|\bar{\mu}_k(p) - \mu_k| \geq B_{p,2}\right) \\
&= P\left(\left|\bar{\mu}_k(p) - \hat{\mu}_k^{M(p)}(p) + \hat{\mu}_k^{M(p)}(p) - \mu_k\right| \geq B_{p,2}\right) \\
&\leq P\left(\left|\bar{\mu}_k(p) - \hat{\mu}_k^{M(p)}(p)\right| + \left|\hat{\mu}_k^{M(p)}(p) - \mu_k\right| \geq B_{p,2}\right) \\
&= P\left(\left|\bar{\mu}_k(p) - \hat{\mu}_k^{M(p)}(p)\right| + \left|\hat{\mu}_k^{M(p)}(p) - \mu_k\right| \geq \sqrt{6\sigma^2\eta_p \log(T)} + \sqrt{\frac{6\sigma_c^2 \log(T)}{M(p)}}\right) \\
&\leq P\left(\left|\bar{\mu}_k(p) - \hat{\mu}_k^{M(p)}(p)\right| \geq \sqrt{6\sigma^2\eta_p \log(T)}\right) + P\left(\left|\hat{\mu}_k^{M(p)}(p) - \mu_k\right| \geq \sqrt{\frac{6\sigma_c^2 \log(T)}{M(p)}}\right).
\end{aligned}$$

For the first part, at phase p, the averaged local mean is $\bar{\mu}_k(p) = \frac{1}{M(p)} \sum_{m=1}^{M(p)} \bar{\mu}_{k,m}(p)$, while $\bar{\mu}_{k,m}(p)$ is the sample mean collected by client m. It can be observed that arm k is pulled for $\sum_{q=1}^{p} f(q) = F(p)$ times by $g(1)$ clients (referred as "group 1"), $\sum_{q=2}^{p} = F(p) - F(1)$ times by $g(2)$ clients ("group 2"), and so on until $f(p) = F(p) - F(p-1)$ times by $g(p)$ clients ("group p"). We also have that for clients in groups 1, $\bar{\mu}_{k,m}$ is a $\frac{\sigma}{\sqrt{F(p)}}$-subgaussian random variable, while it is a $\frac{\sigma}{\sqrt{F(p)-F(1)}}$-subgaussian random variable for clients in group 2, and so on. We further have that the overall average $\bar{\mu}_k(p)$ is a $\frac{\sigma}{M(p)}\sqrt{\sum_{q=1}^{p} \frac{g(q)}{F(p)-F(q-1)}}$-subgaussian random variable. With the sub-gaussian property and $\eta_p = \frac{1}{M(p)^2} \sum_{q=1}^{p} \frac{g(q)}{F(p)-F(q-1)}$, it holds that

$$P\left(\left|\bar{\mu}_k(p) - \hat{\mu}_k^{M(p)}(p)\right| \geq \sqrt{6\sigma^2\eta_p \log(T)}\right) \leq 2\exp\left\{-\frac{6\sigma^2\eta_p \log(T)}{2\frac{\sigma^2}{M(p)^2}\sum_{q=1}^{p} \frac{g(q)}{F(p)-F(q-1)}}\right\} = \frac{2}{T^3}.$$

For the second part, $\hat{\mu}_k^{M(p)}(p) = \frac{1}{M(p)} \sum_{m=1}^{M(p)} \mu_{k,m}$, which is a $\frac{\sigma_c}{\sqrt{M(p)}}$-subgaussian random variable. Thus, with the subgaussian property, we have

$$P\left(\left|\hat{\mu}_k^{M(p)}(p) - \mu_k\right| \geq \sqrt{\frac{6\sigma_c^2 \log(T)}{M(p)}}\right) \leq 2\exp\left\{-\frac{\frac{6\sigma_c^2 \log(T)}{M(p)}}{2\frac{\sigma_c^2}{M(p)}}\right\} = \frac{2}{T^3}.$$

By combining the two parts together, the lemma is proved. □

Denote event $B = \{\forall p, \forall k \in [K_p], |\bar{\mu}_k(p) - \mu_k| \leq B_{p,2}\}$ and $P_b = \mathbb{P}(B)$. Since there are at most T rounds and K arms, with a simple union bound, we have

$$P_b \geq 1 - \frac{4K}{T^2}.$$

Step 2: Required Number of Pulls

Based on that event B happens, the following lemma provides an upper bound for the required number of pulls to eliminate any sub-optimal arm.

Lemma 3.1.7. *Assuming that event B happens, for any sub-optimal arm $k \neq k_*$, there are at most p_k rounds before arm k is eliminated or the overall time runs out, where p_k is the smallest integer satisfying*

$$96\left(\sigma\sqrt{\eta_p} + \sigma_c \frac{1}{\sqrt{M(p)}}\right)^2 \log(T) \leq (\mu_* - \mu_k)^2.$$

Proof. Let $\Delta_k = \mu_* - \mu_k$ be the gap between the sub-optimal arm k and the optimal arm and p_k be the smallest integer such that $4B_{p_k,2} \leq \Delta_k$. Thus p_k is the smallest integer satisfying

$$96\left(\sigma\sqrt{\eta_p} + \sigma_c \frac{1}{\sqrt{M(p)}}\right)^2 \log(T) \leq (\mu_* - \mu_k)^2.$$

For any $p \geq p_k$, we have

$$\Delta_k \geq 4B_{p,2};$$
$$|\bar{\mu}_k(p) - \mu_k| \leq B_{p,2};$$
$$|\bar{\mu}_{k_*}(p) - \mu_*| \leq B_{p,2}.$$

With the above inequalities, we have

$$\begin{aligned}\bar{\mu}_k(p) + B_{p,2} &\leq \mu_k + 2B_{p,2} \leq \mu_k + 2B_{p,2} + \bar{\mu}_{k_*}(p) - \mu_* + B_{p,2} \\ &= -(\Delta_k - 4B_{p,2}) + \bar{\mu}_{k_*}(p) - B_{p,2} \leq \bar{\mu}_{k_*}(p) - B_{p,2},\end{aligned}$$

which means arm k is eliminated. Thus, arm k is pulled at most p_k rounds before elimination or the overall time runs out. □

Step 3: Overall Regret

When event B holds, the overall regret, denoted as $R_{s,2}(T)$, can be decomposed as

$$R_{s,2}(T) = \sum_{k=1}^{K}(\mu_* - \mu_k)\mathbb{E}[N(k)] + C\mathbb{E}\left[\sum_{\tau=1}^{T_c} M_{\gamma_\tau}\right],$$

where $N(k)$ is the overall number of pulls on arm k. For the first term, i.e. the exploration loss, for any sub-optimal arm k, with p_k defined in Lemma 3.1.7, we have

$$(\mu_* - \mu_k)\mathbb{E}[N(k)] \leq (\mu_* - \mu_k)\sum_{p=1}^{p_k} M(p)f(p).$$

The communication loss is similarly determined by $p_{\max} = \max_{k\neq k_*}\{p_k\}$, which satisfies

$$C\sum_{\tau=1}^{T_c} M_{\gamma_\tau} \leq C\sum_{q=1}^{p_{\max}} M(q).$$

Then $R_{s,2}(T)$ can be bounded as

$$R_{s,2}(T) \leq \sum_{k\neq k_*}(\mu_* - \mu_k)\sum_{p=1}^{p_k} M(p)f(p) + C\sum_{q=1}^{p_{\max}} M(q).$$

If event B does not hold, with βT as an upper bound for the number of clients, the exploration regret and communication regret are upper bounded by a linear loss βT^2 and $\beta C T^2$ respectively. Thus, the regret of this case, denoted as $R_{f,2}(T)$, can be bounded as

$$R_{f,2}(T) \leq \beta(1+C)T^2.$$

With $R_{s,2}(T)$ and $R_{f,2}(T)$, the overall regret $R_2(T)$ can be finally bounded as

$$\begin{aligned} R_2(T) &= P_b R_{s,2}(T) + (1-P_b)R_{f,2}(T) \\ &\leq R_{s,2}(T) + (1-P_b)R_{f,2}(T) \\ &\leq \sum_{k\neq k_*}(\mu_* - \mu_k)\sum_{p=1}^{p_k} M(p)f(p) + C\sum_{q=1}^{p_{\max}} M(q) + 4\beta(1+C). \end{aligned}$$

Proof of Theorem 3.1.4

Step 1: Confidence Bound for the Estimations

Lemma 3.1.8. *At phase p, for any active arm $k \in [K_p]$, it holds that*

$$P\left(|\bar{\mu}_k(p) - \mu_k| \geq B_{p,1}\right) \leq \frac{2}{T^2}.$$

Proof. Since $\bar{\mu}_k(p) = \frac{1}{M}\sum_{m=1}^{M} \bar{\mu}_{k,m}(p)$ while $\bar{\mu}_{k,m}(p)$ is the sample mean collected by client m through $F(p)$ pulls, which thus is a $\frac{\sigma}{\sqrt{F(p)}}$-subgaussian random variable, $\bar{\mu}_k(p)$ is a $\frac{\sigma}{\sqrt{MF(p)}}$-subgaussian random variable. Thus, with the subgaussian property, we have

$$P\left(|\bar{\mu}_k(p) - \mu_k| \geq B_{p,1}\right) \leq 2\exp\left\{-\frac{MF(p)B_{p,1}^2}{2\sigma^2}\right\} = 2\exp\left\{-\frac{MF(p)\frac{6\sigma^2\log(T)}{MF(p)}}{2\sigma^2}\right\} = \frac{2}{T^3}.$$

□

Denoting event $A = \{\forall p, \forall k \in [K_p], |\bar{\mu}_k(p) - \mu_k| \leq B_{p,1}\}$ and $P_a = \mathbb{P}(A)$, since there are at most T rounds and K arms, with a simple union bound, we have

$$P_a \geq 1 - \frac{2K}{T^2}.$$

Step 2: Required Number of Pulls

Lemma 3.1.9. *Assuming that event A happens, for any sub-optimal arm $k \neq k_*$, there are at most p_k rounds before it is eliminated or the overall time runs out, where p_k is the smallest integer satisfying*

$$MF(p_k) \geq \frac{96\sigma^2\log(T)}{(\mu_* - \mu_k)^2}.$$

Proof. Let $\Delta_k = \mu_* - \mu_k$ be the gap between the sub-optimal arm k and the optimal arm, and p_k be the smallest integer such that $4B_{p_k,1} \leq \Delta_k$. We have

$$MF(p_k) \geq \frac{96\sigma^2\log(T)}{(\mu_* - \mu_k)^2}.$$

If $p_k \leq F^{-1}(T)$, for such $p \geq p_k$, it leads to

$$\Delta_k \geq 4B_{p,1};$$
$$|\bar{\mu}_k(p) - \mu_k| \leq B_{p,1};$$
$$|\bar{\mu}_{k_*}(p) - \mu_*| \leq B_{p,1}.$$

With the above inequalities, we can further derive

$$\begin{aligned}\bar{\mu}_k(p) + B_{p,1} &\leq \mu_k + 2B_{p,1} \leq \mu_k + 2B_{p,1} + \bar{\mu}_{k_*}(p) - \mu_* + B_{p,1} \\ &= -(\Delta_k - 4B_{p,1}) + \bar{\mu}_{k_*}(p) - B_{p,1} \leq \bar{\mu}_{k_*}(p) - B_{p,1},\end{aligned}$$

which means arm k is eliminated. Thus, arm k is pulled for at most p_k phases by each client before elimination or the overall time runs out. □

Step 3: Overall Regret

When event A holds, the overall regret, denoted as $R_{s,1}(T)$, can be decomposed as

$$R_{s,1}(T) = \sum_{k=1}^{K}(\mu_* - \mu_k)\mathbb{E}[N(k)] + \mathbb{E}\left[CMT_c\right],$$

where $N(k)$ is the overall number of pulls on arm k by all the clients. For the first term, i.e. the exploration loss, with Lemma 3.1.9, for any sub-optimal arm k, if it holds that

$$\frac{96\sigma^2 \log(T)}{(\mu_* - \mu_k)^2} \leq MF(p_k)$$

at round p_k, we can conclude arm k is eliminated in this round. Thus,

$$(\mu_* - \mu_k)\mathbb{E}[N(k)] \leq M(\mu_* - \mu_k)F(p_k).$$

Since there are no more communications after the optimal arm is found or the overall time runs out, the communication loss is determined by $p_{\max} = \max_{k \neq k_*}\{p_k\}$ and can be bounded as

$$CMT_c \leq CM \min\left\{\max_{k \neq k_*}\{p_k\}, F^{-1}(T)\right\} \leq CMp_{\max}.$$

Then $R_{s,1}(T)$ can be bounded as

$$R_{s,1}(T) \leq M \sum_{k \neq k_*} (\mu_* - \mu_k) F(p_k) + CMp_{\max}.$$

If event A does not hold, the exploration regret can be simply upper bounded by a linear loss MT while the communication loss is also simply upper bounded linearly by CMT. The regret of this case, denoted as $R_{f,1}(T)$, can be bounded as

$$R_{f,1}(T) \leq (1 + C)MT.$$

With $R_{s,1}(T)$ and $R_{f,1}(T)$, the overall regret $R_1(T)$ can finally be bounded as

$$\begin{aligned} R_1(T) &= P_a R_{s,1}(T) + (1 - P_a) R_{f,1}(T) \\ &\leq R_{s,1}(T) + (1 - P_a) R_{f,1}(T) \\ &\leq M \sum_{k \neq k_*} (\mu_* - \mu_k) F(p_k) + CMp_{\max} + 2(1 + C)MK/T. \end{aligned}$$

3.2 Federated Multi-armed Bandits: Flexible Tradeoffs between Generalization and Personalization

Earlier FL approaches focus on training a single global model that can perform well on the aggregated global dataset. However, the performance of the FL-trained global model on an individual client dataset degrades dramatically when significant heterogeneity among the local datasets exists, which raises the concern of using one global model for all individual clients in edge inference. To address this issue, FL with personalization (Smith et al., 2017) has been proposed. Instead of learning a single global model, each device aims at learning a mixture of the global model and its own local model (Hanzely and Richtárik, 2020; Deng et al., 2020), which provides an explicit trade-off between the two potentially competing learning goals. Following these attempts, the following discussions extend the concept of "personalization" to the study of federated multi-armed bandits, especially as a framework of personalized federated multi-armed bandits (PF-MAB).

3.2.1 Problem Fomulation

Clients and local models.

In the PF-MAB framework, there are M clients interacting with the same set of K arms (referred as "local arms"). The clients are labeled from 1 to M to facilitate the discussion (labelling is not used in the algorithm). For client m, arm k generates local rewards $X_{k,m}(t)$ independently from a σ-subgaussian distribution with mean $\mu_{k,m}$. Without loss of generality, we assume $\sigma = 1$. For different clients, their local models are non-IID, i.e., in general $\mu_{k,n} \neq \mu_{k,m}$ when $n \neq m$. A client can only interact with her own local MAB model by choosing arm $\pi_m(t)$ and receiving reward $X_{\pi_m(t),m}(t)$ at time t. Also, there is no direct communication between clients.

A global stochastic MAB model with the same set of K arms (referred to as "global arms") coexists with the local models, where the global reward $X_k(t)$ for the global arm k is the average of local rewards, i.e., $X_k(t) = \frac{1}{M}\sum_{m=1}^{M} X_{k,m}(t)$. The global reward can be thought of as the *virtual* averaged reward had all M clients pulled the same arm k at time t. Correspondingly, the mean reward of global arm k is $\mu_k = \frac{1}{M}\sum_{m=1}^{M}\mu_{k,m}$. We note that although the global model is the average of local models, the global rewards are not directly observable by any client.

In decentralized multi-player multi-armed bandits (MP-MAB), clients are prohibited from having *explicit* communication with each other (Liu and Zhao, 2010; Boursier and Perchet, 2019). We modify this constraint to enable client-server periodic communication that is similar to FL. Specifically, the clients can send "local model updates" to a central server, which then aggregates and broadcasts the updated "global model" to the

clients. (We will specify these components later.) Note that just as in FL, communication is one of the major bottlenecks and the algorithm has to be conscious of its usage. This constraint is incorporated by imposing a loss C each time a communication round happens, which will be accounted for in the regret. We also make the assumption that clients and the server are fully synchronized.

Personalization vs Generalization

With the coexistence of local and global models, two extreme scenarios exist for bandit learning: local-only and global-only. In the first case, clients only care about their own local performance, which is characterized by the local cumulative reward $r_l(T)$ as

$$r_l(T) := \mathbb{E}\left[\sum_{t=1}^{T}\sum_{m=1}^{M} X_{\pi_m(t),m}(t)\right].$$

$r_l(T)$ is equivalent to the sum rewards of M clients who play M decoupled and non-interacting MAB games. Obviously, the optimal choice for client m is arm $k_{*,m}$ with $\mu_{*,m}:=\mu_{k_{*,m},m}=\max_{k\in[K]}\mu_{k,m}$. However, only pursuing the locally optimal arm severely limits the ability to generalization across clients, especially when the degree of heterogeneity is significant.

On the other extreme, clients only focus on learning the global model, which means maximizing the global cumulative reward:

$$r_g(T) := \mathbb{E}\left[\sum_{t\in[T]}\sum_{m\in[M]} X_{\pi_m(t)}(t)\right].$$

In this case, although the client's action and observation are both on her local arms, the reward is defined with respect to the global arm (Shi and Shen, 2021a). Ideally, the optimal choice to maximize $r_g(T)$ is to let all the clients play the optimal global arm k_* with $\mu_*:=\mu_{k_*}=\max_{k\in[K]}\mu_k$. We note that this problem has recently been proposed and studied in Zhu et al. (2020); Shi and Shen (2021a), which calls for efficient coordination among clients since no client can solve the global model individually. However, any efficient solution for this extreme case may lead to poor individual performance due to the non-IID local models.

To balance the need for both personalization and generalization, we hereby introduce a new learning objective that mixes $r_g(T)$ and $r_l(T)$ by a parameter $\alpha\in[0,1]$. This learning objective is referred to as the *mixed cumulative reward*, which is defined as:

$$r(T) := \alpha r_l(T) + (1-\alpha) r_g(T). \tag{3.8}$$

The parameter α provides a flexible choice of personalization: with $\alpha=1$, $r(T)$ becomes the sum rewards of

M individual single-player MAB games (full personalization); with $\alpha = 0$, $r(T)$ only considers the global model (no personalization); with $0 < \alpha < 1$, both the global and local models are simultaneously taken into consideration by $r(T)$.

The Equivalent Mixed Model

An equivalent view of the mixed cumulative reward $r(T)$ in Eqn. (3.8) is provided here, which facilitates our subsequent discussion. By unfolding $r_l(T)$ and $r_g(T)$, $r(T)$ can be rewritten as

$$r(T) = \mathbb{E}\left[\sum\nolimits_{t\in[T]} \sum_{m\in[M]} X'_{\pi_m(t),m}(t)\right],$$

where $X'_{\pi_m(t),m}(t)$ is a hypothetical reward that combines the local and global rewards, defined as:

$$X'_{\pi_m(t),m}(t) := \alpha X_{\pi_m(t),m}(t) + (1-\alpha) X_{\pi_m(t)}(t). \tag{3.9}$$

Thus, maximizing the mixed cumulative reward can be equivalently viewed as playing a new MAB game with $X'_{k,m}(t)$ as rewards for the clients. However, since clients cannot directly observe the global reward, $X'_{k,m}(t)$ is only *partially observable* at each individual client. We refer to this hypothetical game as the *mixed model*. A similar reward definition using the weighted sum of clients' rewards has been adopted in Brânzei and Peres (2019), albeit from a game theory perspective.

In client m's mixed model, the mean reward $\mu'_{k,m} := \mathbb{E}\left[X'_{k,m}(t)\right]$ for arm k can be calculated as:

$$\mu'_{k,m} = \underbrace{\left(\alpha + \frac{1-\alpha}{M}\right)\mu_{k,m}}_{\text{local info}} + \underbrace{\frac{1-\alpha}{M}\sum\nolimits_{n\neq m}\mu_{k,n}}_{\text{global info}}. \tag{3.10}$$

Since the global information in $\mu'_{k,m}$ is determined by other clients and cannot be accessed directly at client m, communication between clients and the server is of critical importance.

With the mixed models, the notion of regret in single-agent MAB can be generalized to $r(T)$ as

$$R(T) = T\sum_{m=1}^{M}\mu'_{*,m} - \mathbb{E}\left[\sum_{t=1}^{T}\sum_{m=1}^{M} X'_{\pi_m(t),m}(t)\right] + CMT_c, \tag{3.11}$$

where $\mu'_{*,m}$ is the mean reward from the optimal arm $k'_{*,m}$ of client m's mixed model, i.e., $\mu'_{*,m} := \mu'_{k'_{*,m},m} = \max_{k\in[K]}\mu'_{k,m}$. The additional loss term CMT_c in Eqn. (3.11) represents the communication loss, where T_c is the total amount of communication slots. Without loss of generality, we assume that the optimal arm of each client on her mixed model is unique. We further note that the optimal arms of different clients are likely

to be different because in general non-IID local models lead to $k'_{*,m} \neq k'_{*,n}$ when $m \neq n$. We further denote $\Delta'_{k,m} = \mu'_{*,m} - \mu'_{k,m}$.

3.2.2 Lower Bound Analysis

A regret lower bound of PF-MAB is characterized by the following theorem.

Theorem 3.2.1. *For any consistent*[1] *algorithm* Π*, the regret* $R(T)$ *in Eqn.* (3.11) *can be lower bounded as*

$$\liminf_{T\to\infty} \frac{R_\Pi(T)}{\log(T)} \geq \sum_{m=1}^{M} \sum_{k\neq k'_{*,m}} \max\left\{ \frac{\Delta'_{k,m}}{\mathrm{kl}(Y_{k,m}, Y_{k'_{*,m},m})}, \frac{\Delta'_{k,m}}{\min_{n:n\neq m, k'_{*,n}\neq k} \mathrm{kl}(Z^m_{k,n}, Z^m_{k'_{*,n},n})} \right\}, \tag{3.12}$$

where $Y_{k,m} = \left(\alpha + \frac{1-\alpha}{M}\right) X_{k,m} + \mu'_{k,m} - \left(\alpha + \frac{1-\alpha}{M}\right) \mu_{k,m}$ *and* $Z^m_{k,n} = \frac{1-\alpha}{M} X_{k,m} + \mu'_{k,n} - \frac{1-\alpha}{M} \mu_{k,m}$.

The communication cost is ignored in the analysis (i.e., $C = 0$), but naturally, this lower bound still holds for $C > 0$. The lower bound in Eqn. (3.12) sums over the maximum of two terms for all clients and suboptimal arms. First, random variable $Y_{k,m}$ with mean $\mu'_{k,m}$ represents an idealized degenerated game of client m's mixed model where information from other clients, i.e., $\{\mu_{k,l}\}_{l\neq m}$, is perfectly known. With $Y_{k,m}$, a lower bound for the regret of client m learning arm k for her mixed model can be obtained. Second, random variable $Z^m_{k,n}$ with mean $\mu'_{k,n}$ represents another idealized degenerated game of client n's mixed model, where we assume full information of arm k from all other clients except client m, i.e., $\{\mu_{k,l}\}_{l\neq m}$. With $Z^m_{k,n}$, the regret of client m providing information of arm k to client n is characterized. Then, building on this characterization, the regret of client m providing information of arm k to all other clients can be lower bounded by taking the worst case among them, i.e., the minimization term. This worst-case argument corresponds to the client who requires the most global information of arm k. To summarize, the first and second terms in the maximization characterize the necessary loss for learning local (for the client herself) and global (for all other clients) information of client m's arm k, respectively. We also note that in the case of $\alpha = 1$, i.e., local-only, Eqn. (3.12) recovers the lower bound in Lai and Robbins (1985), summed over M local models.

More light can be shed on the lower bound by limiting the attention to Gaussian distributed rewards.

Corollary 3.2.2. *For any consistent algorithm* Π*, if the rewards follow Gaussian distributions with unit variance, the regret is lower bounded as*

$$\liminf_{T\to\infty} \frac{R_\Pi(T)}{\log(T)} \geq \sum_{m=1}^{M} \sum_{k\neq k'_{*,m}} \max\left\{ \frac{2\beta^2}{\Delta'_{k,m}}, \frac{2\gamma^2 \Delta'_{k,m}}{(\Delta'_k)^2} \right\},$$

[1]The consistent algorithm is defined the same way as in Lai and Robbins (1985) but with the regret of Eqn. (3.11).

where $\beta = \alpha + \frac{1-\alpha}{M}$, $\gamma = \frac{1-\alpha}{M}$ *and* $\Delta'_k = \min_{n:k'_{*,n} \neq k} \Delta'_{k,n}$.

Corollary 3.2.2 shows that the second term in the maximum is determined by Δ'_k, which corroborates that the loss of learning global information for arm k is determined by the hardest mixed model.

We note that, as will be evident in the PF-UCB algorithm, the lower bound analysis reveals important guidelines for balancing global and local explorations. Nevertheless, neither Theorem 3.2.1 nor Corollary 3.2.2 establishes a *universally tight* lower bound (for all α). Characterizing the precise lower bound dependency on α is an interesting open problem, and we have the following conjecture.

Conjecture 3.2.3. *For any consistent algorithm* Π, *as* $T \to \infty$, $\forall m \in [M]$ *and* $\forall k : k \neq k'_{*,m}$, *it holds that*

$$\frac{\beta^2}{T_{k,m}} + \sum_{n:n\neq m, k'_{*,n} \neq k} \frac{\gamma^2}{T_{k,n}} \leq \frac{\eta^2 \mathrm{kl}(X'_{k,m}, X'_{k'_{*,m},m})}{\log(T)},$$

where $T_{k,m}$ *is the expected number of pulls on arm* k *by client* m *in the* T *time slots,* $\beta = \alpha + \frac{1-\alpha}{M}$, $\gamma = \frac{1-\alpha}{M}$ *and* $\eta = (\beta^2 + (M-1)\gamma^2)^{\frac{1}{2}}$.

Conjecture 3.2.3 also recovers single-agent lower bound in Lai and Robbins (1985)with $\alpha = 1$. Furthermore, with $\alpha = 0$ (global-only), it implies that $\liminf_{T\to\infty} \frac{R(T)}{\log(T)} \geq \sum_{k\neq k_*} \frac{M\Delta_k}{\mathrm{kl}(X_k, X_{k_*})}$, where $\Delta_k = \mu_* - \mu_k$. This result is reasonable as it is equivalent to the lower bound of a centralized client who directly maximizes the cumulative global reward.

3.2.3 The PF-UCB Algorithm

Challenges

Solving the PF-MAB model faces several new challenges. The first challenge is that in order to maximize the mixed reward, both local and global information are essential. On one hand, the overall decision can be compromised (depending on the choice of α) as long as one type of information is insufficiently learned. On the other hand, providing global information for other clients may degrade individual performance since the additional exploration does not directly benefit the client. The key challenge is how to gain *sufficient but not excessive* local and global information simultaneously based on the required degree of personalization.

A second challenge is that the game difficulties vary across clients. It is highly likely that different clients would need different amounts of global information. In other words, some clients may find their optimal arms much slower than others, which is similar to the client heterogeneity problem in FL (Li et al., 2020a). How to handle the resulting synchronization problem caused by client heterogeneity in PF-MAB becomes an important issue.

Lastly, although communication is fundamental to providing global information, it incurs additional losses in regret. This benefit-cost balance needs to be addressed in the algorithm design.

Algorithm Design

The Personalized Federated Upper Confidence Bound (PF-UCB) algorithm operates in phases (analogous to communication rounds in FL), and each phase consists of three sub-phases: global exploration, local exploration, and exploitation. The set of arms for global (resp. local) exploration is referred to as the set of global (resp. local) active arms. Specifically, at phase p, $A_m(p)$ (with cardinality $K_m(p)$) and $A(p) = \cup_{m\in[M]} A_m(p)$ (with cardinality $K(p)$) denote the set of local and global active arms respectively, which are both initialized as $[K]$. PF-UCB for clients and the central server are presented in Algorithms 7 and 8, respectively.

Algorithm 7 PF-UCB: client m

Require: T, M, K, α

Initialize $p \leftarrow 1$; $A_m(1), A(1) \leftarrow [K]$; $\forall k \in [K], s_{k,m} \leftarrow 0$, $T_{k,m} \leftarrow 0$; $g, h \leftarrow 1$; $O_m \leftarrow 0$
while $A(p) \neq \emptyset$ **do** ▷ `Global exploration`
  for $g \leq K(p)\lceil(1-\alpha)f(p)\rceil$ **do**
    $\pi \leftarrow (g \bmod K(p))$-th arm in $A(p)$
    Pull arm π and receive reward r_π
    $s_{\pi,m} \leftarrow s_{\pi,m} + r_\pi$; $T_{\pi,m} \leftarrow T_{\pi,m} + 1$; $g \leftarrow g + 1$
  end for
  for $h \leq K_m(p)\lceil M\alpha f(p)\rceil$ **do** ▷ `Local exploration`
    $\pi \leftarrow (h \bmod K_m(p))$-th arm in $A_m(p)$
    Pull arm π and receive reward r_π
    $s_{\pi,m} \leftarrow s_{\pi,m} + r_\pi$; $T_{\pi,m} \leftarrow T_{\pi,m} + 1$; $h \leftarrow h + 1$
  end for
  Update $\bar{\mu}_{k,m}(p) \leftarrow s_{k,m}/T_{k,m}, \forall k \in A(p)$
  Send $\bar{\mu}_{k,m}(p), \forall k \in A(p)$ to the server
  if $O_m = 0$ **then** ▷ `Exploitation`
    $\bar{k}'_{*,m}(p) \leftarrow \arg\max_{k\in A_m(p)}\{\bar{\mu}'_{k,m}(p-1)\}$
  else $\bar{k}'_{*,m}(p) \leftarrow O_m$
  end if
  Pull arm $\bar{k}'_{*,m}(p)$ until receiving $\bar{\mu}_k(p), k \in A(p)$
  $\forall k \in A_m(p), \bar{\mu}'_{k,m}(p) \leftarrow \alpha\bar{\mu}_{k,m}(p) + (1-\alpha)\bar{\mu}_k(p)$
  Update $E_m(p)$ ▷ *Arm elimination*
  $A_m(p+1) \leftarrow A_m(p) \backslash E_m(p)$
  if $|A_m(p+1)| = 1$ **then**
    $O_m \leftarrow$ the only arm in $A_m(p+1)$; $A_m(p+1) \leftarrow \emptyset$
  end if
  Send $A_m(p+1)$ to the server
  Receive $A(p+1)$ from the server; $p \leftarrow p+1$; $g, h \leftarrow 1$
  $K(p+1) \leftarrow |A(p+1)|$, $K_m(p+1) \leftarrow |A_m(p+1)|$
end while
Stay on arm O_m until T ▷ *Exploitation*

Algorithm 8 PF-UCB: central server

Require: T, M, K

Initialize $p \leftarrow 1$; $A(1) \leftarrow [K]$
while $A(p) \neq \emptyset$ **do**
  Receive $\bar{\mu}_{k,m}(p), \forall k \in A(p)$ from all clients $m \in [M]$
  Update $\bar{\mu}_k(p) \leftarrow \frac{1}{M}\sum_{m=1}^{M} \bar{\mu}_{k,m}(p)$, $\forall k \in A(p)$
  Send $\bar{\mu}_k(p), \forall k \in A(p)$ to all clients
  Receive $A_m(p+1)$ from all clients
  Send $A(p+1) \leftarrow \cup_{m\in[M]} A_m(p+1)$ to all clients
  $p \leftarrow p+1$
end while

In phase p, global exploration is first performed in order to collect statistics to update the global information. Client m explores each arm $k \in A(p)$, i.e., global active arms, for $n^g_{k,m}(p) = \lceil (1-\alpha) f(p) \rceil$ times, and thus the entire global exploration sub-phase lasts for $K(p)\lceil (1-\alpha) f(p) \rceil$ time slots. Note that $f(p)$ is a flexible exploration length determined by the phase index p, and its impact on the regret is analyzed later. Since all clients share the same global active arm set $A(p)$, the global exploration length is also the same for them.

After the global exploration, the clients perform local exploration to update the local information. Each arm $k \in A_m(p)$ is played by client m for $n^l_{k,m}(p) = \lceil M\alpha f(p) \rceil$ times, which means the local exploration lasts for $K_m(p)\lceil M\alpha f(p) \rceil$ time slots at client m. It is important to note that since different clients may have local exploration sets of different sizes, i.e., $K_m(p)$ can be different across m, the local exploration length may also vary across clients.

Note that the lengths of global and local explorations are carefully designed. For each arm $k \in A_m(p)$, it is explored for $\lceil (1-\alpha) f(p) \rceil$ times during global exploration (recall that $A_m(p) \subseteq A(p) = \cup_{m\in[M]} A_m(p)$) and $\lceil M\alpha f(p) \rceil$ times during local exploration, leading to a total of $n_{k,m}(p) = \lceil (1-\alpha) f(p) \rceil + \lceil M\alpha f(p) \rceil$ pulls by client m. At the same time, client m is also assured that arm k is pulled by every other client n for at least $n^g_{k,n}(p) = \lceil (1-\alpha) f(p) \rceil$ times since they share the same $A(p)$. Thus, the proportion between local and global information is $\frac{(1-\alpha)+M\alpha}{(1-\alpha)}$, which coincides with the desired allocation in Eqn. (3.10).

After completing both global and local explorations, client m first sends the updated local sample means of all global active arms $k \in A(p)$, denoted as $\bar{\mu}_{k,m}(p)$ for arm k at phase p, as the "local model updates" to the server. Since the local exploration length may vary, the server may not receive the updates from all clients at the same time. Thus, it has to wait until the updated sample means from all the clients are received and then sends the aggregated "global model" $\bar{\mu}_k(p) = \frac{1}{M}\sum_{m=1}^{M} \bar{\mu}_{k,m}(p)$ back to the clients. While this waiting time is necessary to synchronize the clients, it also leads to an increased regret, i.e., all clients have to wait for the slowest client before the next iteration.

In PF-MAB, The celebrated exploration-exploitation tradeoff in MAB is embraced to keep the regret caused by this waiting time low. The idea is that clients who have already sent local updates can begin

exploitation while the server still waits to collect information from other clients. Specifically, before $\bar{\mu}_k(p)$ are sent back, client m keeps playing her empirically best arm $\bar{k}'_{*,m} = \arg\max_{k \in A_m(p)}\{\bar{\mu}'_{k,m}(p-1)\}$, where $\bar{\mu}'_{k,m}(p-1)$ is the estimation of $\mu'_{k,m}$ in the preceding phase. Regret analysis shows that this is essential in keeping clients update periodically synchronized while achieving a low regret.

After the global sample means $\bar{\mu}_k(p)$ are broadcast to the clients, the estimation for $\mu'_{k,m}$ is updated as $\bar{\mu}'_{k,m}(p) = \alpha\bar{\mu}_{k,m}(p) + (1-\alpha)\bar{\mu}_k(p)$. Then, a local arm elimination procedure is performed such that the arms that are sub-optimal with a high probability are eliminated. With the newly calculated $\bar{\mu}'_{k,m}(p)$, the elimination set $E_m(p)$ can be constructed as:

$$\left\{k : k \in A_m(p), \max_{l \in A_m(p)} \bar{\mu}'_{l,m}(p) - \bar{\mu}'_{k,m}(p) \geq 2B_p\right\},$$

where $B_p = \sqrt{4\log(T)/(MF(p))}$ is the confidence bound and $F(p) = \sum_{q=1}^{p} f(q)$. Note that the simple and clean form of B_p comes from the carefully designed lengths of global and local explorations. The local active set $A_m(p+1)$ for the next phase is updated as $A_m(p+1) = A_m(p) \backslash E_m(p)$. Finally, all the clients send $A_m(p+1)$ to the server and subsequently receive the global active set $A(p+1) = \cup_{m\in[M]} A_m(p+1)$ from the server. As long as an arm is in the local active set of at least one client, it is contained in the global active set because more global information regarding this arm is still needed to help (at least) that client make decisions.

When the local active set contains only one arm, i.e., $|A_m(q)| = 1$, client m marks the only left arm in $A_m(q)$ as the fixed arm O_m and sets $A_m(q) = \emptyset$. Then, she only sends an empty set to the server for the local active set update since her optimal arm is found. Also, with $A_m(q) = \emptyset$, client m does not perform local explorations any more. Nevertheless, global exploration is still necessary for client m as long as $A(q)$ is not empty, because other clients still need information from her. In the exploitation phase, she also directly plays the fixed arm O_m. When all clients have found their optimal arms, i.e., $A(q) = \emptyset$, they all fixate on their identified arms until the end of T without any further communication.

Remarks. It can be observed that the choice of local exploration length scales linearly with the number of clients, i.e., $n^l_{k,m}(p) = \lceil M\alpha f(p)\rceil \propto M$, which may not be desirable when M is large. It is possible to simultaneously scale down the local and global exploration lengths as M increases, e.g., $n^l_{k,m}(p) = \lceil \alpha f(p)\rceil$ and $n^g_{k,m}(p) = \lceil (1-\alpha)f(p)/M\rceil$, to further trade off exploration and communication, but this does not fundamentally change the regret behavior that is to be discussed. A final note is that only sample means and active sets are communicated in the entire procedure – no raw samples and number of pulls are shared. This is similar to sharing model updates instead of raw data samples in FL, which helps preserve privacy.

3.2.4 Regret Analysis

The theoretical analysis for PF-UCB is presented in this section. In particular, Theorem 3.2.4 characterizes a regret upper bound of PF-UCB.

Theorem 3.2.4. *$\forall m \in [M]$ and $\forall k \neq k'_{*,m}$, suppose $p'_{k,m}$ is the smallest integer that satisfies*

$$MF(p'_{k,m}) \geq \frac{64\log(T)}{(\Delta'_{k,m})^2}. \tag{3.13}$$

The regret of PF-UCB can be bounded as

$$\begin{aligned} R(T) \leq & \sum_{m=1}^{M} \sum_{k \neq k'_{*,m}} \Delta'_{k,m} \sum_{p=1}^{p'_{k,m}} \lceil \alpha M f(p) \rceil \\ & + \sum_{m=1}^{M} \sum_{k \neq k'_{*,m}} \Delta'_{k,m} \sum_{p=1}^{p'_{k}} \lceil (1-\alpha) f(p) \rceil \\ & + \sum_{m=1}^{M} \sum_{k \neq k'_{*,m}} \Delta'_{k,m} \sum_{p=1}^{p'_{k,m}} K \lceil \alpha M f(p) \rceil P'_{k,m}(p) \\ & + 2CMp'_{\max} + 2(1+2C)M^2K, \end{aligned} \tag{3.14}$$

where $p'_k = \max_{m\in[M]}\{p'_{k,m}\}$, $p'_{\max} = \max_{k\in[K]}\{p'_k\}$ and $P'_{k,m}(p) = \exp\{-\Delta'^2_{k,m} MF(p-1)/4\}$.

Detailed proof of Theorem 3.2.4 can be found in the supplementary material, which shows that total regret can be decomposed into local and global exploration losses, exploitation loss, and communication loss. Note that the local exploration loss (the first term) is determined individually by each client's local model, i.e., $p'_{k,m}$, while the global exploration loss (the second term) is determined globally, i.e., p'_k. This coincides with Theorem 3.2.1 and Corollary 3.2.2. In addition, there is no global (resp. local) exploration loss in the local-only (resp. global-only) scenario, i.e., $\alpha = 1$ (resp. 0). Furthermore, although the constant term $2(1+2C)M^2K$ in Eqn. (3.14) has a dependence on M^2, one may trade off this term with other regret terms by adjusting the confidence bound, e.g., specifying $B_p = \sqrt{4\log(MT)/(MF(p))}$.

While Theorem 3.2.4 provides a general characterization with unspecified $f(p)$, the following corollary gives an explicit form of regret with $f(p) = 2^p \log(T)$.

Corollary 3.2.5. *With $f(p) = 2^p \log(T)$, it holds that*

$$R(T) = O\left(\sum_{m=1}^{M} \sum_{k \neq k'_{*,m}} \left[\frac{\alpha}{\Delta'_{k,m}} + \frac{\frac{1-\alpha}{M}\Delta'_{k,m}}{(\Delta'_k)^2}\right] \log(T)\right),$$

where $\Delta'_k = \min_{n:k'_{*,n}\neq k}\{\Delta'_{k,n}\}$.

With this choice, PF-UCB achieves an $O(\log(T))$ regret *regardless of* α. It also has a similar instance dependency on $\Delta'_{k,m}$ and Δ'_k as shown in Corollary 3.2.2. Although the α-dependency does not match Corollary 3.2.2, which is not necessarily a tight lower bound, Corollary 3.2.5 does match the sum of single-player lower bound when $\alpha = 1$. Interestingly, when $\alpha = 0$, the achievable upper bound in Corollary 3.2.5 approaches the conjectured lower bound in Conjecture 3.2.3. It is also worth noting that the communication and exploitation losses when $f(p) = 2^p \log(T)$ are both of order $O(1)$, which demonstrates its efficiency. Regrets with other choices of $f(p)$ can be found in the supplementary material.

We highlight the key components in the proof of Theorem 3.2.4 and Corollary 3.2.5 in the remainder of this section. A typical event

$$G = \{|\bar{\mu}'_{k,m}(p) - \mu'_{k,m}| \leq B_p, \forall p, \forall m \in [M], \forall k \in A_m(p)\}$$

is first established, and we can show that event G happens with high probability.

Lemma 3.2.6. *It holds that* $\mathbb{P}(G) := P_G \geq 1 - \frac{2MK}{T}$.

We then analyze the different loss components of the total regret in the following.

Exploration Loss

First, Lemma 3.2.7 bounds the number of pulls at clients on their sub-optimal arms.

Lemma 3.2.7. *Suppose event G happens. For client m, sub-optimal arm $k \neq k'_{*,m}$ is guaranteed to be eliminated by phase $p'_{k,m}$ as defined in Theorem 3.2.4.*

Then, the local and global exploration losses, denoted as $R_l^{expr}(T)$ and $R_g^{expr}(T)$, respectively, can be bounded by the following lemma.

Lemma 3.2.8. *Suppose event G happens. With $p'_{k,m}$ and p'_k defined in Theorem 3.2.4, $R_l^{expr}(T)$ and $R_g^{expr}(T)$ can be bounded, respectively, as*

$$R_l^{expr}(T) \leq \sum_{m=1}^{M} \sum_{k \neq k'_{*,m}} \Delta'_{k,m} \sum_{p=1}^{p'_{k,m}} \lceil \alpha M f(p) \rceil,$$

$$R_g^{expr}(T) \leq \sum_{m=1}^{M} \sum_{k \neq k'_{*,m}} \Delta'_{k,m} \sum_{p=1}^{p'_k} \lceil (1-\alpha) f(p) \rceil.$$

Note that $R_g^{expr}(T)$ for arm k is determined by p'_k, which is from the hardest local model for arm k. It also matches Theorem 3.2.1 and Corollary 3.2.2.

Exploitation Loss

The exploitation loss $R^{expt}(T)$ is caused by the exploitations when a client has to wait for other clients. Noting that this loss stops once the optimal arm is declared. $R^{expt}(T)$ can be bounded as follows.

Lemma 3.2.9. *Suppose event G happens. With $p'_{k,m}$ and $P'_{k,m}(p)$ defined in Theorem 3.2.4, $R^{expt}(T)$ can be bounded as*

$$R^{expt}(T) \leq \sum_{m=1}^{M} \sum_{k \neq k'_{*,m}} \Delta'_{k,m} \sum_{p=1}^{p'_{k,m}} K \lceil M\alpha f(p) \rceil P'_{k,m}(p).$$

Communication Loss

Since communication stops once all the optimal arms are declared, the communication loss is bounded as:

Lemma 3.2.10. *Suppose event G happens. With $p'_{\max}$ defined in Theorem 3.2.4, the communication loss $R^{comm}(T)$ can be bounded as*

$$R^{comm}(T) \leq 2CMp'_{\max}.$$

With Lemmas 3.2.6 to 3.2.10, Theorem 3.2.4 can be proved.

3.2.5 Algorithm Enhancement

While the exploration length in Section 3.2.3 can be viewed as evenly splitting the workload among clients (especially for the global exploration), it ignores the fact that the same action results in different losses at different clients. We propose an enhancement to adaptively adjust the exploration lengths for client m, as follows:

$$n^l_{k,m}(p) \propto \frac{\alpha M f(p)}{(\Delta'_{k,m})^{1/2}}, \forall k \in A_m(p), k \neq k'_{*,m};$$
$$n^g_{k,m}(p) \propto \frac{(1-\alpha) f(p)}{(\Delta'_{k,m})^{1/2}}, \forall k \in A(p), k \neq k'_{*,m}.$$

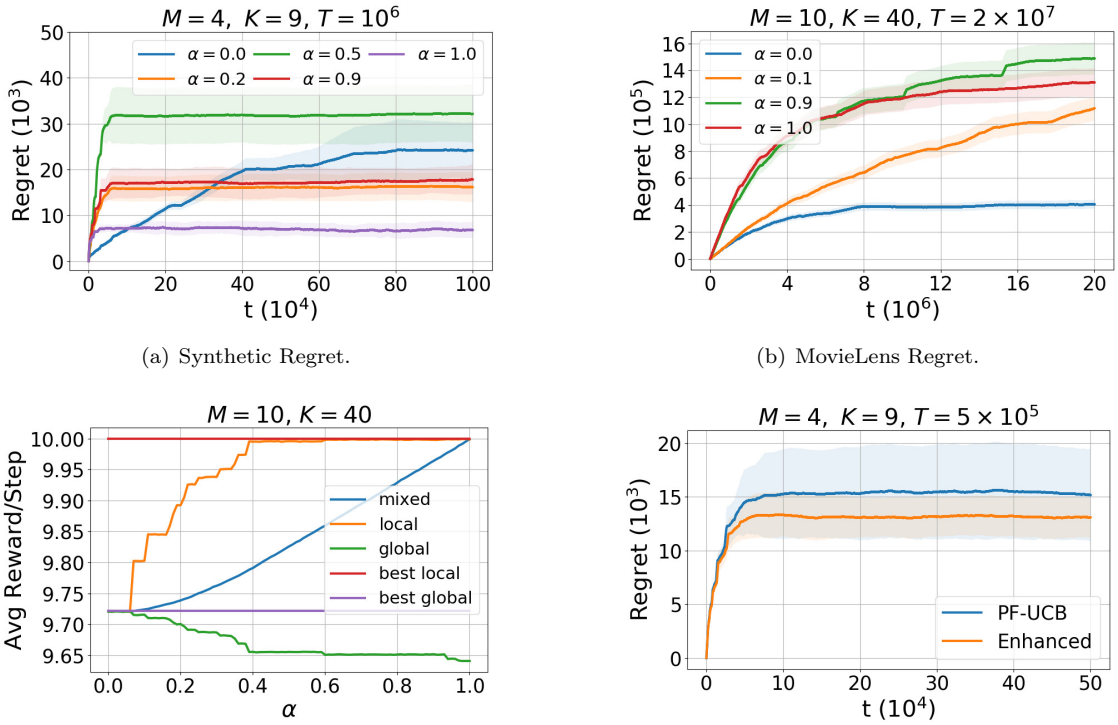
More details on designing this enhancement can be found in the supplementary material. Note that the exploration length for arm k is now proportional to $1/(\Delta'_{k,m})^{1/2}$, which coincides with the intuition that the workload should decrease for those clients who suffer large losses, i.e., with large $\Delta'_{k,m}$'s. However, this is difficult to implement without the knowledge of $\Delta'_{k,m}$. One way to resolve this is to assume all of the sub-optimal gaps are the same, which results in the chosen length in Section 3.2.3. In this enhancement, however, we propose to replace $\Delta'_{k,m}$ by an estimation $\bar{\Delta}'_{k,m}(p)$ in phase p, which can be specified as

$$\bar{\Delta}'_{k,m}(p) = \max_{l \in [K]} \bar{\mu}'_{l,m}(p-1) - \bar{\mu}'_{k,m}(p-1) + 2B_{p-1}.$$

Rigorously analyzing the regret of this enhancement turns out to be difficult, and we evaluate it only through experiments.

3.2.6 Experimental Results

Experiment results using both synthetic and real-world datasets are reported in this section to evaluate PF-UCB and the proposed enhancement. The communication loss is set as $C = 1$ and $f(p)$ is set to be $2^p \log(T)$. Details of the experiments (including the implementation codes) and additional results can be found in the supplementary material.



(a) Synthetic Regret.

(b) MovieLens Regret.

(c) MovieLens Reward.

(d) Enhancement

Figure 3.4: Experimental results for PF-MAB. (a) and (d) is evaluated with synthetic datasets, (b) and (c) are evaluated with the real-world MovieLens dataset.

First, PF-UCB is evaluated with various choices of α under a synthetic bandit game with 4 clients and 9 arms. The game is carefully designed such that all clients have different local optimal arms and the global optimal arm is also sub-optimal locally. Fig. 3.4(a) shows that PF-UCB successfully converges to the optimal choices across different values of α, which proves its effectiveness in handling different combinations of personalization and generalization. The varying overall regrets and convergence speeds are the result of different game difficulties associated with different α.

We then return to one of the motivating examples – the recommender system – and utilize the real-world MovieLens dataset (Cantador et al., 2011) for an empirical study of PF-MAB. The 2113 clients and 10197 movies in the dataset are randomly divided into 10 and 40 groups, respectively, and the averaged movie ratings from each group of clients are used to construct their local rewards, which vary across the groups of clients and naturally lead to non-IID local models. This game is larger and harder than the previous synthetic game. Especially, some groups have suboptimality gaps on their mixed models at around 10^{-4}. As shown in Fig. 3.4(b), sub-linear regrets are achieved by PF-UCB with different values of α. Note that in some cases (e.g., $\alpha = 0.1$), the algorithm does not completely converge within the given horizon; however, the regret curve only increases slowly at the end, which suggests that most of the suboptimal arms are eliminated.[2]

We also evaluate the rewards instead of regrets in the same MovieLens experiments. Fig. 3.4(c) reports the averaged per-step reward that PF-UCB achieves with varying α. The optimal global and local rewards (labeled as "best global" and "best local") represent the theoretically highest global and local mean rewards, respectively. The mixed, global, and local rewards (labeled as "mixed", "global", and "local") generated by the actions of clients with PF-UCB are plotted. Fig. 3.4(c) shows that the mixed and global rewards almost meet the optimal global rewards with $\alpha = 0$ (global-only), while the local rewards are highly sub-optimal. With an increase of α, the mixed and local rewards trend up, indicating the focus is gradually shifted towards the local rewards, and simultaneously the global rewards trend down. At $\alpha = 1$ (local-only), the mixed and local rewards almost achieve the optimal local rewards, while the global rewards are poor. This gradual shifting shows that introducing α provides a smooth tradeoff between local and global rewards.

Lastly, the algorithm enhancement in Section 3.2.5 is evaluated. With a 4-client 9-arm game, the performance of the original and enhanced PF-UCB is compared in Fig. 3.4(d) with $\alpha = 0.5$. It can be observed that both algorithms converge but the enhanced design has a lower regret, demonstrating its effectiveness.

3.2.7 Omitted Algorithmic Details

As stated in Section 3.2.3, the key challenge to solving PF-MAB is how to gain *sufficient but not excessive* local and global information simultaneously based on the required degree of personalization. Sections 3.2.3 and 3.2.5 provide two choices and here the details behind these choices are elaborated.

From client m's perspective on a locally active arm $k \neq k'_{*,m}$, in order to maintain the convergence rate of $1/(MF(p))$ (as specified in Section 3.2.3) while reducing the loss, an optimization problem over $N_{k,m}(p)$ and

[2]We note that in practice, the dataset is likely to be structured more carefully, e.g., grouping movies by categories instead of randomly, which would in general lead to easier games and faster convergence.

$N^g_{k,n}(p), \forall n \neq m$ can be formulated as:

$$\begin{aligned}&\text{minimize } N_{k,m}(p)\Delta'_{k,m} + \sum\nolimits_{n\neq m, k'_{*,n}\neq k} N^g_{k,n}(p)\Delta'_{k,n}\\ &\text{subject to } \frac{\left[\alpha + (1-\alpha)/M\right]^2}{N_{k,m}(p)} + \sum_{n\neq m}\frac{\left[(1-\alpha)/M\right]^2}{N^g_{k,n}(p)} \leq \frac{1}{MF(p)}\end{aligned}$$

where $N_{k,m}(p)$ is the number of pulls on arm k at client m up to phase p, and $N^g_{k,n}(p)$ is the guaranteed number of global pulls on arm k at a different client n up to phase p. The optimization objective is the loss associated with client m's local and global information estimation for arm k, while the constraint is a sufficient condition for $B_p = \sqrt{4\log(T)/(MF(p))}$ and Lemma 3.2.6 to hold. Note that the convergence rate constraint can have many forms, and the choice here is to match the discussion in the main paper.

Using the Cauchy-Schwarz inequality, the exploration length described in Section 3.2.5 can be obtained as:

$$\begin{aligned}n^l_{k,m}(p) &\propto \frac{\alpha M f(p)}{(\Delta'_{k,m})^{1/2}}, \forall k \in A_m(p), k \neq k'_{*,m};\\ n^g_{k,m}(p) &\propto \frac{(1-\alpha) f(p)}{(\Delta'_{k,m})^{1/2}}, \forall k \in A(p), k \neq k'_{*,m},\end{aligned}$$

and $N^l_{k,m}(p) = \sum_{q=1}^p n^l_{k,m}(q)$, $N^g_{k,m}(p) = \sum_{q=1}^p n^g_{k,m}(q)$ and $N_{k,m}(p) = N^l_{k,m}(p) + N^g_{k,m}(p)$. This result is the key to choosing exploration lengths as it builds up the relationship between local and global explorations.

The issue however is that the knowledge of $\Delta'_{k,m}$ is unavailable. An easy way to tackle this problem is to assume all the sub-optimal gaps are the same, which results in the chosen length in PF-UCB in Section 3.2.3. The alternative way proposed in Section 3.2.5 is to use $\bar{\Delta}'_{k,m}(p) = \max_{l\in[K]} \bar{\mu}'_{l,m}(p-1) - \bar{\mu}'_{k,m}(p-1) + 2B_{p-1}$ in place of $\Delta'_{k,m}(p)$. This approach leverages information collected in the game. However, $\bar{\Delta}'_{k,m}(p)$ needs to be communicated to the server and then broadcast to maintain synchronization among clients, which may increase the risk of privacy leak.

3.2.8 Full Proofs

Proofs for the Lower Bound Analysis in Theorem 3.2.1

Proof. First, the following lemma recalls the classic result from the single-player MAB (Lai and Robbins, 1985), which directly leads to the following lower bound.

Lemma 3.2.11. *For any consistent policy* Π*, for any arm* k *such that* $\mu_k < \mu_{k_*}$*, it holds that*

$$\liminf_{T\to\infty} \frac{T_k}{\log(T)} \geq \frac{1}{\mathrm{kl}\left(X_k, X_{k_*}\right)},$$

where T_k is the expected number of pulls performed on arm k during T.

Then, from client m's perspective of her suboptimal arm $k \neq k_{*,m}$ on the mixed model, the mixed reward in Eqn. (3.9) can be decomposed as

$$X'_{k,m} = \left(\alpha + \frac{1-\alpha}{M}\right) X_{k,m} + \frac{1-\alpha}{M} \sum_{n \neq m} X_{k,n}.$$

The difficulty is that $X'_{k,m}$ involves the rewards from all M clients, which are M sources of randomness. Next, we attempt to isolate these sources of randomness.

First, if we assume client m has perfect knowledge of $\{\mu_{k,n}\}_{n \neq m}$, a new random variable $Y_{k,m}$ is constructed as

$$Y_{k,m} = \left(\alpha + \frac{1-\alpha}{M}\right) X_{k,m} + \frac{1-\alpha}{M} \sum_{n \neq m} \mu_{k,n} = \left(\alpha + \frac{1-\alpha}{M}\right) X_{k,m} + \mu'_{k,m} - \left(\alpha + \frac{1-\alpha}{M}\right) \mu_{k,m}.$$

Under this construction, $Y_{k,m}$ shares the same mean with $X'_{k,m}$ while the randomness only comes from $X_{k,m}$. Then, $Y_{k,m}$ forms a new hypothetical bandit game degenerated from client m's mixed model, where the mean rewards and the optimal arm remain the same. With Lemma 3.2.11, if client m individually interacts with this new game, her pulls on arm k can be bounded as

$$\liminf_{T \to \infty} \frac{T_{k,m}}{\log(T)} \geq \frac{1}{\mathrm{kl}\left(Y_{k,m}, Y_{k'_{*,m},m}\right)}.$$

On the other hand, from a different client n's perspective, whose arm k is also sub-optimal, she also needs information of client m's arm k. However, client n's mixed reward is constructed as

$$X'_{k,n} = \left(\alpha + \frac{1-\alpha}{M}\right) X_{k,n} + \frac{1-\alpha}{M} X_{k,m} + \frac{1-\alpha}{M} \sum_{l \neq m,n} X_{k,l},$$

which is different from $X'_{k,m}$. Following a similar idea of isolating randomness, if we assume client n has perfect knowledge of $l \neq m, \mu_{k,l}$, including $\mu_{k,n}$, a new random variable $Z^m_{k,n}$ can be constructed as

$$Z^m_{k,n} = \left(\alpha + \frac{1-\alpha}{M}\right) \mu_{k,n} + \frac{1-\alpha}{M} X_{k,m} + \frac{1-\alpha}{M} \sum_{l \neq m,n} \mu_{k,l} = \frac{1-\alpha}{M} X_{k,m} + \mu'_{k,n} - \frac{1-\alpha}{M} \mu_{k,m}.$$

Under this construction, $Z^m_{k,n}$ shares the same mean as $X_{k,n}$ while the randomness only comes from $X_{k,m}$. Then $Z^m_{k,n}$ forms another new hypothetical bandit game degenerated from client n's mixed model, where the optimal arm remains the same and client m has to provide information to help client n distinguish arm k.

Similarly, with Lemma 3.2.11, if client m individually interacts with this new game, her pulls on arm k can be bounded as

$$\liminf_{T\to\infty} \frac{T_{k,m}}{\log(T)} \geq \frac{1}{\mathrm{kl}\left(Z^m_{k,n}, Z^m_{k'_{*,n},n}\right)}.$$

Since $Z^m_{k,n}$ can be constructed for any client, it must hold that

$$\liminf_{T\to\infty} \frac{T_{k,m}}{\log(T)} \geq \max_{n:n\neq m,k'_{*,n}\neq k} \left\{ \frac{1}{\mathrm{kl}\left(Z^m_{k,n}, Z^m_{k'_{*,n},n}\right)} \right\} = \frac{1}{\min_{n:n\neq m,k'_{*,n}\neq k} \left\{ \mathrm{kl}\left(Z^m_{k,n}, Z^m_{k'_{*,n},n}\right) \right\}}.$$

Combining the above results, we can have

$$\liminf_{T\to\infty} \frac{T_{k,m}}{\log(T)} \geq \max \left\{ \frac{1}{\mathrm{kl}\left(Y_{k,m}, Y_{k'_{*,m},m}\right)}, \frac{1}{\min_{n:n\neq m,k'_{*,n}\neq k} \left\{ \mathrm{kl}\left(Z^m_{k,n}, Z^m_{k'_{*,n},n}\right) \right\}} \right\}.$$

Since the regret can be decomposed as

$$R(T) = \sum_{m=1}^{M} \sum_{k:k\neq k'_{*,m}} T_{k,m}\Delta'_{k,m},$$

Theorem 3.2.1 can be established. □

Note that the randomness isolation utilized in the proof reduces the hardness of the problem, which results in a relaxed lower bound. Although it can recover the single-player stochastic MAB lower bound with $\alpha = 1$, when α moves away from 1, the lower bound becomes less tight.

Discussions for Theorem 3.2.4

Table 3.3: Regret of PF-UCB algorithm with different choices of $f(p)$

$f(p)$	$p_{k,m}$, $k \neq k'_{*,m}$	$R(T)$
λ	$O\left(\frac{\log(T)}{M\lambda(\Delta'_{k,m})^2}\right)$	$O\left(\sum_{m=1}^{M}\sum_{k\neq k'_{*,m}}\left[\frac{\alpha}{\Delta'_{k,m}} + \frac{\frac{1-\alpha}{M}\Delta'_{k,m}}{\Delta'^2_k}\right]\log(T) + \frac{C\log(T)}{\lambda(\Delta'_{\min})^2}\right)$
$\lambda\log(T)$	$O\left(\frac{1}{M\lambda(\Delta'_{k,m})^2}\right)$	$O\left(\sum_{m=1}^{M}\sum_{k\neq k'_{*,m}}\left[\frac{\alpha}{\Delta'_{k,m}} + \frac{\frac{1-\alpha}{M}\Delta'_{k,m}}{\Delta'^2_k}\right]\log(T) + \frac{C}{\lambda(\Delta'_{\min})^2}\right)$
2^p	$O\left(\log\left(\frac{\log(T)}{M(\Delta'_{k,m})^2}\right)\right)$	$O\left(\sum_{m=1}^{M}\sum_{k\neq k'_{*,m}}\left[\frac{\alpha}{\Delta'_{k,m}} + \frac{\frac{1-\alpha}{M}\Delta'_{k,m}}{\Delta'^2_k}\right]\log(T) + CM\log\left(\frac{\log(T)}{M(\Delta'_{\min})^2}\right)\right)$
$2^p\log(T)$	$O\left(\log\left(\frac{1}{M(\Delta'_{k,m})^2}\right)\right)$	$O\left(\sum_{m=1}^{M}\sum_{k\neq k'_{*,m}}\left[\frac{\alpha}{\Delta'_{k,m}} + \frac{\frac{1-\alpha}{M}\Delta'_{k,m}}{(\Delta'_k)^2}\right]\log(T) + CM\log\left(\frac{1}{M(\Delta'_{\min})^2}\right)\right)$

λ is a constant; $\Delta'_k = \min_{n:k'_{*,n}\neq k}\{\Delta'_{k,n}\}$; $\Delta'_{\min} = \min_k\{\Delta'_k\}$.

Table 3.3 summarizes the regrets under several different choices of $f(p)$, including $f(p) = 2^p \log(T)$ in Corollary 3.2.5. All choices listed in Table 3.3 achieve a similar exploration regret and a non-dominating exploitation loss (which is omitted in the regret expression). However, they lead to varying communication losses. With $f(p) = \lambda$, the communication loss is of order $O(\log(T))$ and scales with $1/(\Delta'_{\min})^2$, which actually dominates the exploration loss. This is the result of the unnecessary communications with $f(p) = \lambda$. With $f(p) = \lambda \log(T)$, the communication loss is no longer of order $O(\log(T))$; however, it still scales with $1/(\Delta'_{\min})^2$. The dependency of communication loss on $\Delta'_{\min}$ is improved with an exponential $f(p)$, as both $f(p) = 2^p$ and $f(p) = 2^p \log(T)$ have communication losses that scale only with $\log(1/\Delta'_{\min})$, which greatly reduces the communication burden. Furthermore, with $f(p) = 2^p \log(T)$, the communication cost is a constant that is independent of T. Thus, among all considered choices of $f(p)$, the most preferable one is $f(p) = 2^p \log(T)$.

We further note that all the choices of $f(p)$ listed in Table 3.3 do not depend on the communication loss parameter C. This is made to simplify the problem, as otherwise, the analysis will have a convoluted relationship between the exploration loss and the communication loss. Intuitively, with a larger C, it is better to increase $f(p)$ to reduce the communication frequency and lower the communication loss, e.g., adding a $1/C$ multiplicative factor to the listed choice of $f(p)$.

Proof of Lemma 3.2.6

Proof. To decouple the randomness of $A_m(p)$, we assume a virtual system without elimination, i.e., in this virtual system $\forall m \in [M], \forall p, A_m(p) = [K]$. At phase p, $\forall m \in [M], \forall k \in A_m(p)$, $\bar{\mu}'_{k,m}(p)$ can be decomposed as

$$\bar{\mu}'_{k,m}(p) = \left(\alpha + \frac{1-\alpha}{M}\right) \bar{\mu}_{k,m}(p) + \frac{1-\alpha}{M} \sum_{n \neq m} \bar{\mu}_{k,n}(p).$$

It can be shown that $\bar{\mu}_{k,m}(p)$ is a $\sqrt{\frac{1}{N_{k,m}(p)}}$-subgaussian random variable, since client m has explored arm k for $N_{k,m}(p) = \sum_{q=1}^{p} n_{k,m}(q)$ times in the global and local exploration sub-phases. However, $\forall n \in [M], n \neq m$, client m can only make sure that $\bar{\mu}_{k,n}(p)$ is a $\sqrt{\frac{1}{N^g_{k,n}(p)}}$-subgaussian random variable, where $N^g_{k,n}(p) = \sum_{q=1}^{p} n^g_{k,n}(q)$, since she is only assured that each other client has explored arm k in the global exploration sub-phases. Overall, we can claim that $\bar{\mu}'_{k,m}(p)$ is a $\sigma'_{k,m}(p)$-subgaussian random variable where

$$\begin{aligned}
\sigma'_{k,m}(p) &= \sqrt{\left(\alpha + \frac{1-\alpha}{M}\right)^2 \frac{1}{N_{k,m}(p)} + \left(\frac{1-\alpha}{M}\right)^2 \sum_{n \neq m} \frac{1}{N^g_{k,n}(p)}} \\
&\leq \sqrt{\left(\alpha + \frac{1-\alpha}{M}\right)^2 \frac{1}{[(1-\alpha) + M\alpha]F(p)} + \left(\frac{1-\alpha}{M}\right)^2 \sum_{n \neq m} \frac{1}{(1-\alpha)F(p)}}
\end{aligned}$$

$$= \sqrt{\frac{1}{MF(p)}}.$$

With the concentration inequality for subgaussian random variables, we have

$$\mathbb{P}\left(|\bar{\mu}'_{k,m}(p) - \mu'_{k,m}| \geq B_p\right) \leq 2\exp\left\{-\frac{B_p^2}{2(\sigma'_{k,m}(p))^2}\right\} \leq 2\exp\left\{-\frac{\frac{4\log(T)}{MF(p)}}{2\frac{1}{MF(p)}}\right\} = \frac{2}{T^2}.$$

Thus, with the union bound, P_G can be bounded as

$$\begin{aligned} P_G &= 1 - \mathbb{P}\left\{\exists p, \exists m \in [M], \exists k \in A_m(p), |\bar{\mu}'_{k,m}(p) - \mu'_{k,m}| \geq B_p\right\} \\ &\geq 1 - \sum_{p=1}^{T}\sum_{m=1}^{M}\sum_{k=1}^{K}\mathbb{P}\left(|\bar{\mu}'_{k,m}(p) - \mu'_{k,m}| \geq B_p\right) \\ &\geq 1 - \frac{2MK}{T}. \end{aligned}$$

Since this argument applies to $k \in [K]$, it also applies to all arms in the local active arm set $A_m(p)$ of the real system, which concludes the proof. □

Proof of Lemma 3.2.7

Proof. Recall that $\forall k \neq k'_{*,m}$, $p'_{k,m}$ is the smallest integer such that

$$MF(p'_{k,m}) \geq \frac{64\log(T)}{(\Delta'_{k,m})^2},$$

which ensures that $\forall p \geq p'_{k,m}, B_p \leq \frac{\Delta'_{k,m}}{4}$. Thus, based on that event G happens, at phase $p'_{k,m}$, we have

$$\begin{aligned} \bar{\mu}'_{k,m}(p'_{k,m}) + B_{p'_{k,m}} &\overset{(i)}{\leq} \mu'_{k,m} + 2B_{p'_{k,m}} \leq \mu'_{k,m} + \frac{\Delta'_{k,m}}{2} \\ &= \mu'_{*,m} - \frac{\Delta'_{k,m}}{2} \overset{(ii)}{\leq} \bar{\mu}'_{k'_{*,m},m}(p'_{k'_{*,m},m}) + B_{p'_{k,m}} - \frac{\Delta'_{k,m}}{2} \leq \bar{\mu}'_{k'_{*,m},m}(p'_{k'_{*,m},m}) - B_{p'_{k,m}}, \end{aligned}$$

where inequalities (i) and (ii) are guaranteed by event G. Thus, arm k is guaranteed to be eliminated at phase $p'_{k,m}$ by client m. □

Proof of Lemma 3.2.8

Proof. Lemma 3.2.7 indicates for a sub-optimal arm k, after phase $p'_{k,m}$, it is guaranteed to be eliminated from set $A_m(p)$. Thus, it is pulled for at most $\sum_{p=1}^{p'_{k,m}} \lceil \alpha M f(p) \rceil$ times in the local exploration sub-phases,

which leads to the local exploration loss as

$$R_l^{expr}(T) \leq \sum_{m=1}^{M} \sum_{k \neq k'_{*,m}} \Delta'_{k,m} \sum_{p=1}^{p'_{k,m}} \lceil \alpha M f(p) \rceil.$$

However, arm k is still pulled in the global exploration sub-phases until $k \notin A(p)$, i.e., arm k is eliminated by all of the clients whose optimal arm is not it. Since arm k is guaranteed to be eliminated globally by phase $p'_k = \max_{m \in [M]}\{p'_{k,m}\}$, it is pulled for at most $\sum_{p=1}^{p'_k} \lceil (1-\alpha) f(p) \rceil$ times in the global exploration sub-phases. Thus, the global exploration loss can be bounded as:

$$R_g^{expr}(T) \leq \sum_{m=1}^{M} \sum_{k \neq k'_{*,m}} \Delta'_{k,m} \sum_{p=1}^{p'_k} \lceil (1-\alpha) f(p) \rceil.$$

□

Proof of Lemma 3.2.9

Proof. At phase p, the exploitation time for client m is at most $\max_n\{|A_n(p)| - A_m(p)\} \lceil M\alpha f(p) \rceil$, which is the difference between the longest local exploration duration and her local exploration duration. The probability that the exploited arm in the exploitation phase, i.e., arm $\bar{k}'_{*,m}$, is arm k instead of $k'_{*,m}$ can be bounded as:

$$\begin{aligned}
\mathbb{P}\left(\bar{k}'_{*,m} = k\right) &\leq P\left(\bar{\mu}'_{k'_{*,m},m}(p-1) \leq \bar{\mu}_{k,m}(p-1)\right) \\
&= P\left(\bar{\mu}'_{k'_{*,m},m}(p-1) - \bar{\mu}_{k,m}(p-1) - \Delta'_{k,m} \leq -\Delta'_{k,m}\right) \\
&\overset{(i)}{\leq} 2\exp\left\{-\frac{(\Delta'_{k,m})^2}{2(\sigma'^2_{k,m}(p-1) + \sigma'^2_{k'_{*,m},m}(p-1))}\right\} \\
&\leq 2\exp\left\{-\frac{(\Delta'_{k,m})^2 MF(p-1)}{4}\right\} \\
&= P'_{k,m}(p).
\end{aligned}$$

Thus, it can be shown that the exploration loss caused by arm k for client m is bounded as

$$\begin{aligned}
R_{k,m}^{expt}(T) &\leq \Delta'_{k,m} \sum_{p=1}^{p'_{k,m}} \left(\max_n\{|A_n(p)| - A_m(p)\}\right) \lceil M\alpha f(p) \rceil P'_{k,m}(p) \\
&\leq \Delta'_{k,m} \sum_{p=1}^{p'_{k,m}} K \lceil M\alpha f(p) \rceil \exp\left\{-\frac{(\Delta'_{k,m})^2 MF(p-1)}{4}\right\}.
\end{aligned}$$

The overall exploration loss can be obtained by summing over all of the clients and arms:

$$R^{expt}(T) = \sum_{m=1}^{M}\sum_{k=1}^{K}\Delta'_{k,m}R^{expt}_{k,m}(T) \leq \sum_{m=1}^{M}\sum_{k\neq k'_{*,m}}\sum_{p=1}^{p'_{k,m}} K\left\lceil M\alpha f(p)\right\rceil \Delta'_{k,m}\exp\left\{-\frac{(\Delta'_{k,m})^2MF(p-1)}{4}\right\}.$$

In addition, we note that in phase $p = 1$, all the players share the same global and local active arm sets, i.e., $\forall m \in [M], A_m(p) = A(p) = [K]$, which means there would be no exploration loss. Thus, the sum of index p in the exploitation loss above can start from 2 instead of 1. This fact does not change the scaling of the overall regret, but would be useful in deriving Corollary 3.2.5 from Theorem 3.2.4. □

Proof of Lemma 3.2.10

Proof. As designed in the PF-UCB algorithm, clients do not communicate anymore after they find their optimal arms. Thus, there is no more communication after phase $p'_{\max} = \max_{k\in[K]}\{p'_{k,m}\} = \max_{m\in[M]}\max_{k\neq k'_{*,m}}\{p'_{k,m}\}$. Before phase $p'_{\max}$, there are two communications in each phase for arm statistics and active sets, respectively, which leads to the communication loss upper bound as:

$$R^{comm}(T) \leq 2CMp'_{\max}.$$

□

Proof of Theorem 3.2.4

Proof. Lemmas 3.2.8, 3.2.9 and 3.2.10 are all based on the condition that event G happens, which has probability P_G as shown in Lemma 3.2.6. When event G does not happen, the regret is directly upper bounded by $MT + 2CMT$, which assumes full exploration and communication loss. Thus, Theorem 3.2.4 follows by putting everything together as:

$$\begin{aligned}
R(T) &= P_G\left(R^{expr}(T) + R^{expt}(T) + R^{comm}(T)\right) + (1-P_G)(1+2C)MT \\
&\leq R^{expr}_l(T) + R^{expr}_g(T) + R^{expt}(T) + R^{comm}(T) + 2M^2K(1+2C) \\
&\leq \sum_{m=1}^{M}\sum_{k\neq k'_{*,m}}\Delta'_{k,m}\sum_{p=1}^{p'_{k,m}}\lceil\alpha Mf(p)\rceil + \sum_{m=1}^{M}\sum_{k\neq k'_{*,m}}\Delta'_{k,m}\sum_{p=1}^{p'_k}\lceil(1-\alpha)f(p)\rceil \\
&+ \sum_{m=1}^{M}\sum_{k\neq k'_{*,m}}\Delta'_{k,m}\sum_{p=1}^{p'_{k,m}} K\left\lceil M\alpha f(p)\right\rceil \exp\left\{-\frac{(\Delta'_{k,m})^2MF(p-1)}{4}\right\} + 2CMp'_{\max} + 2M^2K(1+2C).
\end{aligned}$$

□

Proof of Corollary 3.2.5

Proof. With $f(p) = 2^p \log(T)$, $p'_{k,m}$ can be bounded from Eqn. (3.13) as

$$p'_{k,m} = O\left(\log_2\left(\frac{64}{M(\Delta'_{k,m})^2}\right)\right).$$

Plugging this into Theorem 3.2.4, Corollary 3.2.5 follows. □

3.3 Federated Contextual Bandits: A General Modulized Design

3.3.1 Problem Formulation

This section presents a concise formulation of federated contextual bandits (FCB).

Agents.

In the FCB setting, a total of M agents simultaneously participate in a contextual bandit (CB) system. For generality, we consider an asynchronous system: each of the M agents has a clock indicating her time step, which is denoted as $t_m = 1, 2, \cdots$ for agent m. For convenience, we also introduce a global time step t. Denote by $t_m(t)$ the agent m's local time step when the global time is t, and $t(t_m, m)$ the global time step when the agent m's local time is t_m.

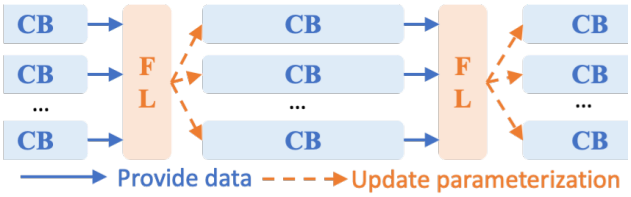
Agent m at each of her local time step $t_m = 1, 2, \cdots$ observes a context x_{m,t_m}, selects an action a_{m,t_m} from an action set $\mathcal{A}_{m,t_m}$, and then receives the associated reward $r_{m,t_m}(a_{m,t_m})$ (possibly depends on both x_{m,t_m} and a_{m,t_m}) as in the standard CB (Lattimore and Szepesvári, 2020). Each agent's goal is to collect as many rewards as possible given a time horizon, which is often formulated as minimizing her regret.

Federation.

While many efficient single-agent (centralized) designs have been proposed for CB (Lattimore and Szepesvári, 2020), FCB targets building a federation among agents to perform collaborative learning such that the performance can be improved from learning independently. Especially, common interests shared among agents motivate their collaboration. Thus, FCB studies typically assume that the environment models of the agents are either fully (Wang et al., 2020b; Huang et al., 2021b; Dubey and Pentland, 2020; Li and Wang, 2022a; He et al., 2022; Amani et al., 2022; Li et al., 2022, 2023; Li and Wang, 2022b; Dai et al., 2023) or partially (Li and Wang, 2022a; Agarwal et al., 2020) shared in the global federation.

In federated learning, the following two modes are commonly considered: (1) There exists a central server in the system, and the agents can share information with the server, which can then broadcast aggregated information back to the agents. (2) There exists a communication graph between agents, who can share information with their neighbors in the graph. The to-be-discussed unified principle can effectively encompass both models, while in the later development of FedIGW, we mainly consider the first scenario, i.e., collaborating through server, which is also the main focus in FL.

Table 3.4: An illustration of the FCB design philosophy of alternating between CB and FL and a compact summary of investigations on FCB with their adopted FL and CB schemes



Principle: FCB = FL + CB			
Ref.	Setting	FL	CB
Globally Shared Full Model			
Wang et al. (2020b)	Tabular	Mean Averaging	AE
Wang et al. (2020b); Huang et al. (2021b)	Linear	Linear Regression	AE
Wang et al. (2020b); Dubey and Pentland (2020) He et al. (2022); Amani et al. (2022)	Linear	Ridge Regression	UCB
Li and Wang (2022b)	Gen. Lin.	Distributed AGD	UCB
Li et al. (2022, 2023)	Kernel	Nyström approx.	UCB
Dai et al. (2023)	Neural	NTK approx.	UCB
FedIGW	Realizable	Flexible	IGW
Globally Shared Partial Model			
Li and Wang (2022a)	Linear	AM	UCB
Agarwal et al. (2020)	Realizable	FedRes.SGD	ε-greedy
FedIGW	Realizable	Flexible	IGW

AE: arm elimination; Gen. Linear: generalized linear model;
AGD: accelerated gradient descent; NTK: neural tangent kernel; AM: Alternating Minimization

3.3.2 A Unified Principle: FCB = FL + CB

The study on FCB dates back to distributed multi-armed bandits (Wang et al., 2020b) on the tabular setting, and FCB studies have mostly focused on how to obtain better performances in a more general problem setting, especially different types of reward functions including linear (Wang et al., 2020b; Huang et al., 2021b; Dubey and Pentland, 2020; Li and Wang, 2022a; He et al., 2022; Amani et al., 2022), kenerlized (Li et al., 2022, 2023), generalized linear (Li and Wang, 2022b) and neural (Dai et al., 2023).

Upon reviewing these works as a whole, it becomes apparent that each of them focuses on a specific CB method and employs a particular communication protocol to update the parameterization required by CB. We thus can summarize that these papers all implicitly follow a unified principle that "**FCB = FL + CB**". This principle is important as it first reveals FCB is fundamentally designed to share agents' local information via FL such that their CB strategies can be updated. Notably, the CB algorithm benefits from having more data to update its parameterized policy for improved decision-making, while FL facilitates this implicit "access" to other agents' data through the designed communications of aggregating local parameters.

Furthermore, this principle indicates that as long as the two black boxes of CB and FL schemes are

compatible with each other, their combination would lead to a functional FCB design. The chosen FL scheme should possess the capability to effectively update the necessary parameterization in the employed CB method. Conversely, the CB approach should provide appropriate datasets to facilitate the execution of FL. To be more specific, as current FL studies typically learn from batched datasets, it is more desirable to have a periodically alternating scheme between CB and FL: CB (collects one epoch of data in parallel) $\rightarrow$ FL (proceeds with CB data together and outputs CB's parameterization) $\rightarrow$ updated CB (collects another epoch of data in parallel) $\rightarrow \cdots$, which is illustrated at the top of Table 3.4. In fact, this philosophy has been implicitly adopted by previous FCB designs to varying degrees. For example, in federated linear bandits (Wang et al., 2020b; Dubey and Pentland, 2020; Li and Wang, 2022a; He et al., 2022; Amani et al., 2022), the CB algorithm is often selected as a batched version of LinUCB (Abbasi-Yadkori et al., 2011), while the adopted FL typically solves a ridge regression problem (although there are differences in the adopted communication protocols, e.g., synchronous or asynchronous). As LinUCB is parameterized by both model estimates and confidence bounds, FL needs to update both, e.g., via aggregating local covariance matrices. The extensions to different reward functions (Li et al., 2022, 2023; Li and Wang, 2022b; Dai et al., 2023) and partially shared global models (Li and Wang, 2022a; Agarwal et al., 2020) also follow similar routines with varying FL modifications to match the considered scenarios. A compact summary has been given in Table 3.4, where the alignment with the design philosophy can be observed.

More importantly, the formalization of this principle can serve as a guiding framework for the development of novel FCB designs. In particular, the FL components in the previous FCB works have some mismatches from canonical FL designs (McMahan et al., 2017; Konečnỳ et al., 2016): most of them adopt specific communication protocols with *one-shot aggregation* of *compressed local data* per epoch (e.g., combining local covariance matrices). Such choices are rare (and even undesirable) in canonical FL designs, where agents typically communicate and aggregate their *model parameters* (e.g., gradients) for *multiple rounds*. Guided by the unified principle and motivated by the deficiency of existing FCB designs, we propose a new method, FedIGW, in the following sections. It leverages IGW as the CB scheme and enables the integration of any flexible FL routine as long as it solves the standard FL problem. The intimate connection to FL is important as it allows us to effectively leverage advances in FL studies, including but not limited to canonical algorithm designs, convergence analyses, and useful appendages, which are discussed in the following sections.

3.3.3 A New Design: FedIGW

System Model

Built on the formulation in Sec. 3.3.1, for each agent $m \in [M]$, let $\mathcal{X}_m$ denote a context space, and $\mathcal{A}_m$ a finite set of K_m actions. We consider that at each time step t_m of each agent m, the environment samples a context $x_{m,t_m} \in \mathcal{X}_m$ and a context-dependent reward vector $r_{m,t_m} \in [0,1]^{\mathcal{A}_m}$ according to a fixed but unknown distribution $\mathcal{D}_m$. Then, as in Sec. 3.3.1, the agent m observes the context x_{m,t_m}, picks an action $a_{m,t_m} \in \mathcal{A}_m$, and observes the reward $r_{m,t_m}(a_{m,t_m})$. The expected reward of playing action a_m facing context x_m is further denoted as $\mu_m(x_m, a_m) := \mathbb{E}[r_{m,t_m}(a_m)|x_{m,t_m} = x_m]$.

With no prior information about the rewards, the agents gradually learn their optimal policies, denoted as $\pi_m^*(x_m) := \arg\max_{a_m \in \mathcal{A}_m} \mu_m(x_m, a_m)$ for agent m with context x_m. Following the standard notation (Wang et al., 2020b; Huang et al., 2021b; Dubey and Pentland, 2020; Li and Wang, 2022a; He et al., 2022; Amani et al., 2022; Li and Wang, 2022b; Li et al., 2022, 2023; Dai et al., 2023), the overall regret of all M agents in this environment is

$$\mathrm{Reg}(T) := \mathbb{E}\left[\sum_{m\in[M]}\sum_{t_m\in[T_m]} [\mu_m(x_{m,t_m}, \pi_m^*(x_{m,t_m})) - \mu_m(x_{m,t_m}, a_{m,t_m})]\right],$$

where $T_m = t_m(T)$ is the effective time horizon for agent m given a global horizon T and the expectation is taken over the randomness in contexts and rewards and the agents' algorithms. This overall regret can be interpreted as the sum of each agent m's individual regret with respect to (w.r.t.) her optimal strategy π_m^*. Hence, it is ideal to be sub-linear w.r.t. the number of agents M, which indicates the agents' learning processes are accelerated on average due to federation.

Despite not knowing the true expected reward functions, we consider the scenario that they are globally shared and are within a function class $\mathcal{F}$, to which the agents have access. This assumption, rigorously stated in the following, is often referred to as the *realizability* assumption.

Assumption 3.3.1 (Realizability)**.** *There exists f^* in $\mathcal{F}$ such that $f^*(x_m, a_m) = \mu_m(x_m, a_m)$ for all $m \in [M]$, $x_m \in \mathcal{X}_m$ and $a_m \in \mathcal{A}_m$.*

This assumption is a natural extension from its commonly adopted single-agent version (Agarwal et al., 2012; Simchi-Levi and Xu, 2022; Xu and Zeevi, 2020; Sen et al., 2021) to a federated one. Note that it does not imply that the agents' environments are the same since they may face different contexts $\mathcal{X}_m$, arms $\mathcal{A}_m$, and distributions $\mathcal{D}_m^{\mathcal{X}_m}$, where $\mathcal{D}_m^{\mathcal{X}_m}$ is the marginal distribution of the joint distribution $\mathcal{D}_m$ on the context space $\mathcal{X}_m$. We study a general FCB setting only with this assumption, which incorporates many previously studied FCB scenarios as special cases. For example, the federated linear bandits (Huang et al., 2021b; Dubey

and Pentland, 2020; Li and Wang, 2022a; He et al., 2022; Amani et al., 2022) are with a linear function class $\mathcal{F}$.

Algorithm 9 FedIGW (Agent m)

Require: epoch number $l = 1$, reward function $\widehat{f}_m^l(\cdot,\cdot) = 0$, local dataset $\mathcal{S}_m^l = \emptyset$

for time step $t_m = 1, 2, \cdots$ **do**
  observe context x_{m,t_m} ▷ *CB: IGW*
  compute $\widehat{a}_m^* = \arg\max_{a_m \in \mathcal{A}_m} \widehat{f}^l(a_m, x_{m,t_m})$ and set action selection distribution as

$$p_m^l(a_m|x_{m,t_m}) \leftarrow \begin{cases} 1/\left(K_m + \gamma^l \left(\widehat{f}^l(\widehat{a}_m^*, x_{m,t_m}) - \widehat{f}^l(a_m, x_{m,t_m})\right)\right) & \text{if } a_m \neq \widehat{a}_m^* \\ 1 - \sum_{a'_m \neq \widehat{a}_m^*} p_m^l(a'_m|x_{m,t_m}) & \text{if } a_m = \widehat{a}_m^* \end{cases}$$

  select action $a_{m,t_m} \sim p_m^l(\cdot|x_{m,t_m})$; observe reward $r_{m,t_m}(a_{m,t_m})$
  update the local dataset, $\mathcal{S}_m^l \leftarrow \mathcal{S}_m^l \cup \{(x_{m,t_m}, a_{m,t_m}, r_{m,t_m}(a_{m,t_m}))\}$
  if $t_m = t_m(\tau^l)$ **then** ▷ *FL*
    perform FL $\widehat{f}^{l+1} \leftarrow \texttt{FLroutine}_m(\mathcal{S}_m^l)$
    update dataset $\mathcal{S}_m^{l+1} \leftarrow \emptyset$; update epoch $l \leftarrow l + 1$
  end if
end for

Algorithm Design

Guided by the unified principle of "FCB = FL + CB", we design a novel algorithm FedIGW proceeding in epochs. The epochs are separated at time slots $\tau^1, \tau^2, \cdots$ w.r.t. the global time step t, i.e., the l-th epoch starts from $t = \tau^{l-1} + 1$ and ends at $t = \tau^l$, and the overall number of epochs is denoted as $l(T)$. In each epoch l, we describe the FL and CB designs, respectively, as follows, while emphasizing on how they are compatible yet decoupled.

CB: Inverse Gap Weighting (IGW). For CB, we use the method of inverse gap weighting (Abe and Long, 1999), which has received growing interest in the single-agent setting recently (Foster and Rakhlin, 2020; Simchi-Levi and Xu, 2022; Krishnamurthy et al., 2021; Ghosh et al., 2021) but has not been fully investigated in the federated setting. At any time step in epoch l, when encountering the context x_m, agent m first estimates the optimal arm by $\widehat{a}_m^* = \arg\max_{a_m \in \mathcal{A}_m} \widehat{f}^l(x_m, a_m)$ from an estimated function $\widehat{f}^l$ (provided by the to-be-discussed FL). Then, she randomly selects her action a_m according to the following distribution, which is inversely proportional to each action's estimated reward gap from the estimated optimal action $\widehat{a}_m^*$:

$$p_m^l(a_m|x_m) \leftarrow \begin{cases} 1/\left(K_m + \gamma^l \left(\widehat{f}^l(\widehat{a}_m^*, x_m) - \widehat{f}^l(a_m, x_m)\right)\right) & \text{if } a_m \neq \widehat{a}_m^* \\ 1 - \sum_{a'_m \neq \widehat{a}_m^*} p_m^l(a'_m|x_m) & \text{if } a_m = \widehat{a}_m^* \end{cases},$$

where γ^l is the learning rate in epoch l that controls the exploration-exploitation tradeoff.

FL: Flexible Designs. By IGW, all agents perform stochastic arm sampling, and thus each agent m collects a set of data samples $\mathcal{S}_m^l := \{(x_{m,t_m}, a_{m,t_m}, r_{m,t_m} : t_m \in [t_m(\tau^{l-1}+1), t_m(\tau^l)])\}$ in epoch l. In order to enhance the CB interactions with IGW in the subsequent epoch $l+1$, an improved estimate $\widehat{f}^{l+1}$ based on all agents' data is desired. This objective aligns precisely with the aim of standard FL, which aggregates local models for better global estimates (McMahan et al., 2017; Konečnỳ et al., 2016).

With this match, the agents can perform a standard FL routine (e.g., FedAvg (McMahan et al., 2017) or Scaffold (Karimireddy et al., 2020)) with the server. To highlight the flexibility and generality, we denote the adopted FL scheme as $\texttt{FLroutine}(\cdot)$ with datasets $\mathcal{S}_{[M]}^l := \{\mathcal{S}_m^l : m \in [M]\}$. $\texttt{FLroutine}(\mathcal{S}_{[M]}^l)$ targets at solving the following standard FL problem:

$$\min_{f \in \mathcal{F}} \widehat{\mathcal{L}}(f; \mathcal{S}_{[M]}^l) := \sum_{m \in [M]} (n_m/n) \cdot \widehat{\mathcal{L}}_m(f; \mathcal{S}_m^l), \tag{3.15}$$

where $n_m := |\mathcal{S}_m^l|$ is the number of samples in dataset $\mathcal{S}_m^l$, $n := \sum_{m \in [M]} n_m$ is the total number of samples, and $\widehat{\mathcal{L}}_m(f; \mathcal{S}_m^l) := (1/n_m) \cdot \sum_{i \in [n_m]} \ell_m(f(x_m^i, a_m^i); r_m^i)$ is the empirical local loss of agent m with $\ell_m(\cdot;\cdot) : \mathbb{R}^2 \to \mathbb{R}$ as the loss function and (x_m^i, a_m^i, r_m^i) as the i-th sample in $\mathcal{S}_m^l$. The output function of this FL process is then used as the estimated reward function $\widehat{f}^{l+1}$ for IGW sampling in the next epoch $l+1$.

The FedIGW algorithm for agent m is summarized in Alg. 9. The key is that the adopted schemes of FL and CB are largely decoupled: IGW only needs an estimated reward function from FL, which provides many theoretical and practical conveniences.

3.3.4 Regret Analysis

We theoretically analyze the performance of the FedIGW algorithm. First, for the output function from the adopted FL routine, we characterize its performance via the following assumption on its excess risk, which is common in the analysis of IGW-type CB algorithms (Simchi-Levi and Xu, 2022; Sen et al., 2021; Ghosh et al., 2021).

Assumption 3.3.2. *Let $p_{[M]} := \{p_m : m \in [M]\}$ be a set of M arbitrary independent arm selection distributions. Given an overall dataset $\mathcal{S}_{[M]} := \{\mathcal{S}_m : m \in [M]\}$ where each dataset $\mathcal{S}_m$ consists of n_m training samples of the form $(x_m, a_m; r_m(a_m))$ independently and identically drawn according to $(x_m, r_m) \sim \mathcal{D}_m$, $a_m \sim p_m(\cdot|x_m)$, the federated routine $\texttt{FLroutine}(\mathcal{S}_{[M]}) = \{\texttt{FLroutine}_m(\mathcal{S}_m) : m \in [M]\}$ returns a predictor $\widehat{f}(\cdot)$. There exists a known parameter $\mathcal{E}(\mathcal{F}; n_{[M]})$ such that*

$$\mathcal{E}(\mathcal{F}; n_{[M]}) \geq \mathbb{E}_{S_{[M]}, \xi}\left[\sum\nolimits_{m \in [M]} \frac{n_m}{n} \cdot \mathbb{E}_{x_m \sim \mathcal{D}_m^{x_m}, a_m \sim p_m(\cdot|x_m)}\left[\left(\widehat{f}(x_m, a_m) - f^*(x_m, a_m)\right)^2\right]\right],$$

where $n_{[M]} := \{n_m : m \in [M]\}$ and ξ denotes the random source in the potentially stochastic FL algorithm. We often abbreviate $\mathcal{E}(\mathcal{F}; n_{[M]})$ as $\mathcal{E}(n_{[M]})$ to simplify notations.

This assumption indicates that in expectation (w.r.t. the random data generation and the stochastic FL process), the output of the adopted FL scheme is close to the true reward function on the weighted data distribution from all agents. Note that the excess risk bound $\mathcal{E}(n_{[M]})$ would typically rely on some other parameters in the adopted FL routine (e.g., the step size and the number of iterations in gradient-based methods), which are currently not specified for generality.

Then, for the asynchronous time steps, we denote $E_m^l := t_m(\tau^l) - t_m(\tau^{l-1})$ as the length of epoch l for agent m and $\underline{c} := \min_{m\in[M],l\in[2,l(T)]} E_m^l/E_m^{l-1}$, $\overline{c} := \max_{m\in[M],l\in[2,l(T)]} E_m^l/E_m^{l-1}$ and $c := \overline{c}/\underline{c}$. At last, under Assumption 3.3.2, we can obtain the following global regret guarantee.

Theorem 3.3.3. *Using a learning rate*

$$\gamma^l = O\left(\sqrt{\textstyle\sum_{m\in[M]} E_m^{l-1} K_m / (\sum_{m\in[M]} E_m^{l-1} \mathcal{E}(E_{[M]}^{l-1}))}\right)$$

in epoch l, denoting $\bar{K}^l := \sum_{m\in[M]} E_m^l K_m / \sum_{m\in[M]} E_m^l$, the regret of FedIGW can be bound as

$$\mathrm{Reg}(T) = O\left(\sum_{m\in[M]} E_m^1 + \sum_{l\in[2,l(T)]} c^{\frac{5}{2}} \sqrt{\bar{K}^l \mathcal{E}(E_{[M]}^{l-1})} \sum_{m\in[M]} E_m^l\right).$$

This performance guarantee is general in the sense that as long as an excess risk bound stated in Assumption 3.3.2 can be established (often via standard convergence and generalization analyses) for a certain class of reward functions and the chosen FL routine, a corresponding regret can be established. Furthermore, the regret incurred within each epoch (i.e., the term inside the sum over l) can be interpreted as the epoch length times the expected per-step suboptimality, which then relates to the estimation quality of $\widehat{f}^l$ and thus $\mathcal{E}(E_{[M]}^{l-1})$ as $\widehat{f}^l$ is learned with the data from epoch $l-1$.

While Theorem 3.3.3 provides a general guarantee, FedIGW can be further specified in different forms when facing different FCB problems. A few instances are discussed in the following two subsections. To ease the notation, we discuss synchronous systems with a shared number of arms, i.e., $t_m = t, \forall m \in [M]$, and $K_m = K, \forall m \in [M]$, while noting similar results can be easily obtained for general systems. With this simplification, we can unify all E_m^l as E^l and $\bar{K}^l$ as K.

Reward Function Classes with Finite Cardinalities.

We first consider that the function class $\mathcal{F}$ is finite, i.e., $|\mathcal{F}| \leq \infty$. Then, if `FLroutine`$(\cdot)$ can provide an exact minimizer $\widehat{f}$ of the optimization problem with quadratic losses, i.e., $\ell_m(f(x_m, a_m); r_m) = (f(x_m, a_m) - r_m)^2$, we can establish the following excess risk bound and the corresponding regret.

Lemma 3.3.4. *If $|\mathcal{F}| < \infty$ and the adopted FL routine provides an exact minimizer for Eqn.* (3.15) *with quadratic losses, Assumption 3.3.2 holds with $\mathcal{E}(n_{[M]}) = O(\log(|\mathcal{F}|n)/n)$.*

Corollary 3.3.5. *If $|\mathcal{F}| < \infty$ and the adopted FL routine provides an exact minimizer for Eqn.* (3.15) *with quadratic losses, with $\tau^l = 2^l$, FedIGW incurs a regret of $\mathrm{Reg}(T) = O(\sqrt{KMT\log(|\mathcal{F}|MT)})$ and a total $O(\log(T))$ calls of the adopted FL routine.*

We note that the obtained regret of order $O(\sqrt{KMT\log(|\mathcal{F}|MT)})$ approaches the optimal regret $\Omega(\sqrt{KMT\log(|\mathcal{F}|)/\log(K)})$ of a single agent playing for MT rounds (Agarwal et al., 2012) up to logarithmic factors, which demonstrates the *statistical efficiency* of the proposed FedIGW. Moreover, the total $O(\log(T))$ times call of the FL routine indicates that only a limited number of agents-server information-sharing are required, which further illustrates its *communication efficiency*.

Reward Function Classes with Convex and Smooth Losses.

Furthermore, we consider that each $f \in \mathcal{F}$ is parameterized by a d-dimensional parameter $\omega \in \mathbb{R}^d$ as f_ω. To facilitate discussions, we abbreviate $\mathcal{S} := \mathcal{S}_{[M]}$ while denoting $\omega^*_{\mathcal{S}} := \arg\min_\omega \widehat{\mathcal{L}}(f_\omega; \mathcal{S})$ as the empirical optimal parameter given a fixed dataset $\mathcal{S}$ and $\widehat{\omega}_{\mathcal{S}}$ as the output of the adopted FL routine. We further assume f^* is parameterized by the true model parameter ω^*, and for a fixed ω, define $\mathcal{L}(f_\omega) := \mathbb{E}_{\mathcal{S}}[\widehat{\mathcal{L}}(f_\omega; \mathcal{S})]$ as its expected loss w.r.t. the data distribution. Following standard learning-theoretic analysis, we recognize that the excess risk in Assumption 3.3.2 can be broken down into a combination of errors stemming from optimization and generalization.

Lemma 3.3.6. *If the loss function $l_m(\cdot;\cdot)$ is μ_f-strongly convex in its first coordinate for all $m \in [M]$, Assumption 3.3.2 holds with*

$$\mathcal{E}(\mathcal{F}; n_{[M]}) = 2\left(\varepsilon_{opt}(\mathcal{F}; n_{[M]}) + \varepsilon_{gen}(\mathcal{F}; n_{[M]})\right)/\mu_f,$$

where $\varepsilon_{gen}(\mathcal{F}; n_{[M]}) := \mathbb{E}_{\mathcal{S},\xi}[\mathcal{L}(f_{\widehat{\omega}_{\mathcal{S}}}) - \widehat{\mathcal{L}}(f_{\widehat{\omega}_{\mathcal{S}}}; \mathcal{S})]$ and $\varepsilon_{opt}(\mathcal{F}; n_{[M]}) := \mathbb{E}_{\mathcal{S},\xi}[\widehat{\mathcal{L}}(f_{\widehat{\omega}_{\mathcal{S}}}; \mathcal{S}) - \widehat{\mathcal{L}}(f_{\omega^*_{\mathcal{S}}}; \mathcal{S})]$.

For the generalization error term, we can use many standard results in learning theory (e.g., uniform convergence). For the sake of simplicity, we here leverage a distributional-independent upper bound on the

Rademacher complexity:

$$\mathfrak{R}(\mathcal{F}; n_{[M]}) = \sup \left\{ \mathbb{E}_{\mathcal{S},\boldsymbol{\sigma}} \left[\sup_{\omega} \left\{ \sum_{m\in[M]} \frac{1}{n} \sum_{i\in[n_m]} \sigma_{m,i} \cdot \ell_m(f_\omega(x_m^i, a_m^i); r_m^i) \right\} \right] \right\},$$

where the outside supremum is over possible distributions of dataset $\mathcal{S}$ defined in Assumption 3.3.2 and the expectation is w.r.t. the generation of dataset $\mathcal{S}$ following a fixed distribution and independent Rademacher random variables $\boldsymbol{\sigma} := \{\sigma_{m,i} : m \in [M], i \in [n_m]\}$. We do not further particularize this upper bound while noting it can be specified following standard procedures (Mohri et al., 2018; Bartlett et al., 2005). Then, the classical uniform convergence result indicates the following lemma.

Lemma 3.3.7. *It holds that* $\varepsilon_{gen}(\mathcal{F}; n_{[M]}) \leq 2\mathfrak{R}(\mathcal{F}; n_{[M]})$.

On the other hand, the optimization error term is related to the specific FL routine adopted in FedIGW. Under standard assumptions in FL studies, the following lemma establishes the optimization error for the considered FL problem with FedAvg (McMahan et al., 2017) as the adopted FL routine. FedAvg is the most standard and commonly adopted FL design, which is also used in our experiments.

Lemma 3.3.8 (Theorem V Karimireddy et al. (2020))**.** *For any dataset* $\mathcal{S}$*, if* $\widehat{\mathcal{L}}_m(f_\omega; \mathcal{S})$ *is* μ_ω*-strongly convex and* β_ω*-smooth w.r.t.* ω *for all* $m \in [M]$ *while the gradients are* σ_b^2*-bounded and have* G_b*-bounded dissimilarity, with FedAvg as the adopted FL routine, the output* $\widehat{\omega}$ *satisfies that*

$$\varepsilon_{opt}(\mathcal{F}; n_{[M]}) \leq \tilde{O}\left(\sigma_b^2(\mu_\omega \rho\kappa M)^{-1} + \beta_\omega G_b^2(\mu_\omega \rho)^{-2}\right),$$

when $\rho \geq \Omega(\beta_\omega/\mu_\omega)$*, where* ρ *denotes the round of communications (i.e., global aggregations) and* κ *denotes the number of local updates (i.e., SGD) between each communication.*

Combining the above two lemmas, the following performance guarantee can be established.

Corollary 3.3.9. *Under the conditions of Lemmas 3.3.6 and 3.3.8, if FedAvg is used as the FL routine, the regret of FedIGW can be bounded as*

$$\mathrm{Reg}(T) = O\left(ME^1 + \sum\nolimits_{l\in[2,l(T)]} \sqrt{\frac{K}{\mu_f} \cdot \left(\mathfrak{R}^{l-1} + \frac{\sigma_b^2}{\mu_\omega \rho^{l-1}\kappa^{l-1}M} + \frac{\beta_\omega G_b^2}{\mu_\omega^2(\rho^{l-1})^2}\right) ME^l}\right),$$

where $\mathfrak{R}^l := \mathfrak{R}(\mathcal{F}; \{E^l : m \in [M]\})$ *while* ρ^l *and* κ^l *the round of agents-server communications and local updates between in epoch* l*, respectively.*

This corollary not only provides a more concrete description of Theorem 3.3.3 but also guides the adopted FL design. As the generalization error is an inherent property that cannot be bypassed by providing better

optimization results, there is no need to further proceed with the FL process as long as the optimization error does not dominate the generalization error, i.e., we can stop the FL process when $\varepsilon_{\text{opt}} = O(\varepsilon_{\text{gen}})$. Following this idea, we provide a more particularized corollary in the following.

Corollary 3.3.10. *Under the conditions of Lemmas 3.3.6 and 3.3.8, with FedAvg as the adopted FL routine, FedIGW incurs a regret of*

$$\text{Reg}(T) = O\left(ME^1 + \sum_{l\in[2,l(T)]} \sqrt{K\mathfrak{R}^{l-1}/\mu_f} ME^l\right)$$

with

$$\tilde{O}\left(\sum_{l\in[l(T)]} \beta_\omega \mu_\omega^{-1} + \sigma_b^2(\mu_\omega \kappa^l M\mathfrak{R}^l)^{-1} + \sqrt{\beta_\omega G_b^2(\mu_\omega^2\mathfrak{R}^l)^{-1}}\right)$$

rounds of communications.

A Linear Reward Function Class. As a more specified instance, we consider linear reward functions as in federated linear bandits, i.e., $f_\omega(\cdot) = \langle \omega, \phi(\cdot)\rangle$ and $f^*(\cdot) = \langle \omega^*, \phi(\cdot)\rangle$, where $\phi(\cdot) \in \mathbb{R}^d$ is a known feature mapping. In this case, the FL problem can be formulated as a standard ridge regression with $\ell_m(f_\omega(x_m, a_m); r_m) := (\langle \omega, \phi(x_m, a_m)\rangle - r_m)^2 + \lambda\|\omega\|_2^2$. With a properly chosen regularization parameter $\lambda = O(1/n)$, the generalization error can be bounded as $\varepsilon_{\text{gen}}(n_{[M]}) = \tilde{O}(d/n)$ (Hsu et al., 2012), while a same-order optimization error can be achieved by many efficient distributed algorithms (Nesterov, 2003) with roughly $O(\sqrt{n}\log(n/d))$ rounds of communications. Then, with an exponentially growing epoch length, FedIGW can have a regret of $\tilde{O}(\sqrt{dMKT})$ with at most $\tilde{O}(\sqrt{MT})$ rounds of communications, both of which are efficient with sublinear dependencies on the number of agents M and time horizon T. It is worth noting that during this process, no raw or compressed data is communicated – only processed model parameters (e.g., gradients) are exchanged. This aligns with FL studies while is distinctive from previous federated linear bandits studies (Wang et al., 2020b; Dubey and Pentland, 2020; Li and Wang, 2022a; He et al., 2022; Amani et al., 2022), which often communicate covariance matrices or aggregated rewards. More discussions can be found in the appendix.

Remark 3.3.11. = From the above results and derivations, we can see that FedIGW provides a general framework to leverage theoretical advances in FL. Thus, beyond these two instances, it is possible to incorporate more advanced results. For example, Huang et al. (2021a) provides a characterization of the optimization and generalization errors of a variant of FedAvg with overparameterized neural networks via NTK analyses, which is conceivably compatible with FedIGW.

3.3.5 Experimental Results

In this section, we report the empirical performances of FedIGW on two distinct real-world multi-label classification datasets, Bibtex (Katakis et al., 2008) and Delicious (Tsoumakas et al., 2008), which are also used in other practical CB investigations such as Cortes (2018). The aim of CB in these experiments is considered to be recommending one of the correct labels at any given time. Especially, in the experiments, at each time step, a context is randomly sampled from the dataset while the true labels are concealed from the agents. The agents then determine which label to select (i.e., pull one arm) with their CB algorithms; thus, the number of arms is the number of possible labels in each dataset. Upon pulling one arm, a reward of 1 is granted if the pulled arm corresponds to one of the true labels, while a reward of 0 is granted otherwise.

Varying FL choices. The reported Fig. 3.5 first compares the averaged rewards collected by each agent with FedIGW using different FL choices, including FedAvg (McMahan et al., 2017), SCAFFOLD (Karimireddy et al., 2020), and FedProx (Li et al., 2020a). This is the first time, to the best of our knowledge, that FedAvg is practically integrated with FCB experiments, let alone other FL protocols, which largely demonstrate the generality and flexibility of FedIGW. It can be observed that using the more developed SCAFFOLD and FedProx provides improved performance (i.e., collects more rewards) compared with the basic FedAvg, which credits to that FedIGW can flexibly leverage algorithmic advances in FL protocols.

Comparison with baselines. To further evaluate the performance of FedIGW, experiments are conducted to compare it with several baselines as described in the following.

- **FN-UCB (Dai et al., 2023).** The federated neural-upper confidence bound (FN-UCB) design proposed in Dai et al. (2023) is adopted as a strong FCB baseline due to its capability of leveraging neural networks to approximate rewards and the previously reported good performance. Instead of being compatible with canonical FL protocols, FN-UCB requires a specifically developed communication design, where local neural tangent features are transmitted to the server for global aggregation in a one-shot fashion.

- **Greedy and softmax.** Besides IGW, two other regression-based CB algorithms, greedy selection and softmax selection, are also adopted for empirical validations using FedAvg to collaboratively learn the reward function. In particular, the action is selected as $a_{m,t_m} \leftarrow \arg\max_{a_m \in \mathcal{A}_m} \widehat{f}^l(a_m, x_{m,t_m})$ for greedy and $a_{m,t_m} \sim \text{softmax}(\widehat{f}^l(\cdot, x_{m,t_m})/\zeta)$ for softmax, where ζ is a tempurate parameter.

In Fig. 3.5, all methods leverage the same-size MLPs to approximate reward functions for fair comparisons. It can be observed that after convergence, FedIGW (even with the basic FedAvg) significantly outperforms FN-UCB with about twice the rewards collected by each agent on average, demonstrating its remarkable

superiority. Also, under the FL protocol (i.e., FedAvg), FedIGW exhibits much stronger performance than greedy and softmax, further illustrating the advantage of using IGW as the CB algorithm.

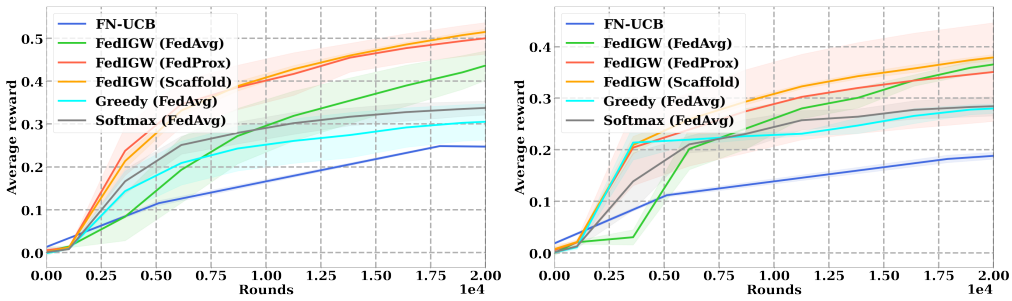


Figure 3.5: Experiments of FCB with Bibtex (left) and Delicious (right).

3.3.6 Flexible Extensions: FedIGW + FL Appendages

Another notable advantage offered by the decoupled FL choices is to bring appropriate appendages from FL that directly benefit FCB, as illustrated in Fig. 3.6. In the following, we discuss how to leverage techniques of personalization, robustness, and privacy from FL in FedIGW, while presenting intriguing avenues for future exploration.

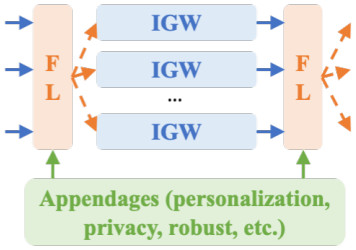


Figure 3.6: Flexible FL appendages in FedIGW.

Personalized Learning

We first consider that each agent m' true reward function $\mu_m(\cdot,\cdot)$ is not globally realizable as in Assumption 3.3.1, but instead only locally realizable in her own function class $\mathcal{F}_m$.

Assumption 3.3.12 (Local Realizability). *For each $m \in [M]$, there exists f_m^* in $\mathcal{F}_m$ such that $f_m^*(x_m, a_m) = \mu_m(x_m, a_m)$ for all $x_m \in \mathcal{X}_m$ and $a_m \in \mathcal{A}_m$.*

Following previous discussions, we consider that each function f in $\mathcal{F}_m$ is parameterized by a d_m-dimensional parameter $\omega_m \in \mathbb{R}^{d_m}$, which is denoted as f_{ω_m}. Correspondingly, the true reward function f_m^* is parameterized by ω_m^* and denoted as $f_{\omega_m^*}$.

We further consider a middle case where only partial parameters are globally shared among $\{f_{\omega_m^*} : m \in [M]\}$ while other parameters are heterogeneous among agents. This setting is aligned with the popular personalized FL studies (Hanzely et al., 2021; Agarwal et al., 2020) and can be formulated via the following assumption.

Assumption 3.3.13. *For all $m \in [M]$, the true parameter ω_m^* can be decomposed as $[\omega^{\alpha,*}, \omega_m^{\beta,*}]$ with $\omega^{\alpha,*} \in \mathbb{R}^{d^\alpha}$ and $\omega_m^{\beta,*} \in \mathbb{R}^{d_m^\beta}$, where $d^\alpha \leq \min_{m\in[M]} d_m$ and $d_m^\beta := d_m - d^\alpha$. In other words, there are d^α-dimensional globally shared parameters among $\{\omega_m^* : m \in [M]\}$.*

A similar setting is studied in Li and Wang (2022a) for linear reward functions and in Agarwal et al. (2020) for realizable cases with a naive ε-greedy design for CB. With FedIGW, we can directly adopt a personalized FL routine, which targets solving a standard personalized FL problem

$$\min_{\omega^\alpha, \omega_{[M]}^\beta} \widehat{\mathcal{L}}(f_{\omega^\alpha, \omega_{[M]}^\beta}; \mathcal{S}_{[M]}) := \sum_{m\in[M]} n_m \widehat{\mathcal{L}}_m(f_{\omega^\alpha, \omega_m^\beta}; \mathcal{S}_m)/n$$

with outputs $\widehat{\omega}^\alpha$ and $\widehat{\omega}_{[M]}^\beta$. Then, the corresponding M output functions $\{f_{\widehat{\omega}^\alpha, \widehat{\omega}_m^\beta} : m \in [M]\}$ (instead of the single one $\widehat{f}$) can be used by the M agents, separately, for their CB interactions following the IGW scheme. More details are in the appendix.

We can bound the generalization error similarly via a distributional-independent Rademacher upper bound defined as $\mathfrak{P}(\mathcal{F}_{[M]}; n_{[M]}) = \sup\{\mathbb{E}_{\mathcal{S},\boldsymbol{\sigma}}[\sup_{\omega^\alpha, \omega_{[M]}^\beta} \{\sum_{m\in[M]} \frac{1}{n} \sum_{i\in[n_m]} \sigma_{m,i} \cdot \ell_m(f_{\omega_m}(x_m^i, a_m^i); r_m^i)\}]\}$. Also, the optimization error of LSGD-PFL (Hanzely et al., 2021), a general design for personalized FL, is characterized in the following lemma.

Lemma 3.3.14 (Theorem 1, Hanzely et al. (2021))**.** *For any dataset $\mathcal{S}$, if $\widehat{\mathcal{L}}_m(f_{\omega_m}; \mathcal{S})$ is μ_ω-strongly convex w.r.t. ω_m, β_{ω^α}-smooth w.r.t. ω^α, and $M\beta_{\omega^\beta}$-smooth w.r.t. ω_m^β for all $m \in [M]$ while the gradients are σ_b^2-bounded and have G_b-bounded dissimilarity, with LSGD-PFL as the adopted FL routine, the output $\widehat{\omega}$ has $\varepsilon_{opt}(\mathcal{F}_{[M]}; n_{[M]}) \leq \varepsilon'$ after*

$$\tilde{O}\left(\max\{\beta_{\omega^\beta}\kappa^{-1}, \beta_{\omega^\alpha}\}\mu_\omega^{-1} + \sigma_b^2(\mu_\omega \kappa M \varepsilon')^{-1} + \sqrt{\beta_{\omega^\alpha}(G^2+\sigma^2)(\mu_\omega^2\varepsilon')^{-1}}\right)$$

rounds of communications, where κ is the number of local updates.

Then, following the idea of having the optimization error approximately the same as the generalization error in Corollary 3.3.10, the following performance guarantee can be established.

Corollary 3.3.15. *Under the conditions of Lemmas 3.3.6 and 3.3.14, with LSGD-PFL as the adopted personalized FL routine, FedIGW incurs a regret of*

$$\mathrm{Reg}(T) = O(ME^1 + \sum\nolimits_{l\in[2,l(T)]} \sqrt{K\mathfrak{P}^{l-1}/\mu_f} ME^l)$$

with

$$\tilde{O}\left(\sum_{l\in[l(T)]} \max\{\beta_{\omega^\beta}(\kappa^l)^{-1}, \beta_{\omega^\alpha}\}\mu_\omega^{-1} + \sigma_b^2(\mu_\omega \kappa^l M\mathfrak{P}^l)^{-1} + \sqrt{\beta_{\omega^\alpha}(G^2+\sigma^2)(\mu_\omega^2\mathfrak{P}^l)^{-1}}\right)$$

rounds of communications, where $\mathfrak{P}^l := \mathfrak{P}(\mathcal{F}_{[M]}, \{E^l : m \in [M]\})$ *and* κ^l *is the number of local updates in epoch* l*.*

A Linear Reward Function Class. We also consider linear reward functions; however, in the personalized setting here, we specify $f_m^*(\cdot) := \langle \omega_m^*, \phi(\cdot)\rangle$ with $\{\omega_m^* : m \in [M]\}$ satisfying Assumption 3.3.13. Then, FedIGW can have a regret of $\tilde{O}(\sqrt{\tilde{d}MKT})$ with at most $\tilde{O}(\sqrt{MT})$ rounds of communications, where $\tilde{d} := d^\alpha + \sum_{m\in[M]} d_m^\beta$. More details are discussed in the appendix.

Robustness, Privacy, and Beyond

Another important direction in FCB studies is to improve robustness against malicious attacks and provide privacy guarantees for local agents. A few progresses have been achieved in attaining these desirable properties. For example, robust aggregation schemes are studied in Demirel et al. (2022); Jadbabaie et al. (2022); Mitra et al. (2022), while different ways of inserting noises to FCB are investigated in Dubey and Pentland (2020); Zhou and Chowdhury (2023); Li and Song (2022) for privacy guarantees.

With the FL and CB largely decoupled in the design of FedIGW, it is more convenient to achieve these properties as suitable techniques from FL studies can be directly applied with only minor modifications. Especially, robustness and privacy protection have been extensively studied for FL in Yin et al. (2018); Pillutla et al. (2022); Fu et al. (2019); Li et al. (2021); Zhu et al. (2023) and Wei et al. (2020); Yin et al. (2021); Liu et al. (2022), respectively, among other works. As long as such FL designs can provide an estimated function (which is a common goal of FL), they can be adopted in FedIGW to achieve additional robustness and privacy guarantees in FCB; see more details in the appendix.

Other Possibilities. There have been many studies on fairness guarantees (Mohri et al., 2019; Du et al., 2021), client selections (Balakrishnan et al., 2022; Fraboni et al., 2021), and practical communication designs (Chen et al., 2021; Wei and Shen, 2022; Zheng et al., 2020) in FL among many other directions, which are all conceivably applicable in FedIGW. In addition, a recent work (Marfoq et al., 2023) studies the FL with data streams, i.e., data comes sequentially instead of being static, which is a suitable design for FCB as CB essentially provides data streams. If similar ideas can be leveraged in FCB, the two components of CB and FL can truly be parallel, instead of being performed alternately.

3.3.7 Full Proofs of the General Analysis

Notations

We first introduce notations that are repeatedly used in the proofs. First, let Υ^l denote the sigma-algebra generated by the history up to epoch l, i.e., $\{(x_{m,t_m}, a_{m,t_m}, r_{m,t_m}) : m \in [M], t_m \in [t_m(\tau^l)]\}$, and the randomness in the adopted FL routine up to epoch l, i.e., $\{\xi_i : i \in [l]\}$, where ξ_i denotes the random source in epoch i. Then, we denote $l_m(t_m) := \min\{l \in \mathbb{N} : t_m \leq t_m(\tau^l)\}$ as the epoch that agent m's t_m belongs to. Also, let $\Psi_m := \mathcal{A}_m^{\mathcal{X}_m}$ denote the set of deterministic functions from $\mathcal{X}_m$ to $\mathcal{A}_m$ for agent m and $\Psi_{[M]} := \times_{m\in[M]}\Psi_m$ the Cartesian product of $\{\Psi_m : m \in [M]\}$. Furthermore, for any action selection kernel $p_{[M]} = \{p_m : m \in [M]\}$, where $p_m(a_m|x_m)$ is the probability of selecting action $a_m \in \mathcal{A}$ given convext x_m, and any policy $\pi_{[M]} = \{\pi_m : m \in [M]\} \in \Psi$, we define

$$
\begin{aligned}
V_m(p_m, \pi_m) &:= \mathbb{E}_{x_m\sim\mathcal{D}_m^{\mathcal{X}_m}}\left[\frac{1}{p_m(\pi_m(x_m)|x_m)}\right], \\
\mathcal{R}_m(\pi_m) &:= \mathbb{E}_{x_m\sim\mathcal{D}_m^{\mathcal{X}_m}}\left[f^*(x_m, \pi_m(x_m))\right], \\
\widehat{\mathcal{R}}_m^l(\pi_m \mid \Upsilon^{l-1}) &:= \mathbb{E}_{x_m\sim\mathcal{D}_m^{\mathcal{X}_m}}\left[\widehat{f}^l(x_m, \pi_m(x_m)) \mid \Upsilon^{l-1}\right], \\
\mathrm{Reg}_m(\pi_m) &:= \mathcal{R}_m(\pi_m^*) - \mathcal{R}_m(\pi_m), \\
\widehat{\mathrm{Reg}}_m^l(\pi_m \mid \Upsilon^{l-1}) &:= \widehat{\mathcal{R}}_{m,t_m}^l(\widehat{\pi}_m^l \mid \Upsilon^{l-1}) - \widehat{\mathcal{R}}_{m,t_m}^l(\pi_m \mid \Upsilon^{l-1}).
\end{aligned}
$$

where $\widehat{\pi}_m^l(x_m) := \arg\max_{a_m\in\mathcal{A}_m} \widehat{f}^l(x_m, a_m)$ for a given $\widehat{f}^l$ (determined by Υ^{l-1}).

The following proofs are largely inspired by the single-agent contextual bandits work (Simchi-Levi and Xu, 2022), while major changes have been made to accommodate the more complex federated system considered in this work.

Proofs of Theorem 3.3.3

First, the following lemma characterizes the relation between the excess errors and the selected learning rates.

Lemma 3.3.16. *For all $l > 1$, it holds that*

$$\mathbb{E}_{\Upsilon^{l-1}}\left[\sum_{m\in[M]}\frac{E_m^{l-1}}{\sum_{m'\in[M]}E_{m'}^{l-1}}\cdot\mathbb{E}_{x_m\sim\mathcal{D}_m^{\mathcal{X}_m},a_m\sim p_m^{l-1}(\cdot|x_m)}\left[\left(\widehat{f}^l(x_m,a_m)-f^*(x_m,a_m)\right)^2\mid\Upsilon^{l-1}\right]\right]$$
$$\leq\mathcal{E}(\mathcal{F};E_{[M]}^{l-1})=\frac{\sum_{m\in[M]}E_m^{l-1}K_m}{\sum_{m\in[M]}E_m^{l-1}(\gamma^l)^2}.$$

Proof. The first inequality is from the Assumption 3.3.2, while the second is based on the choice of γ^l in Theorem 3.3.3, i.e.,

$$\gamma^l=\sqrt{\frac{\sum_{m\in[M]}E_m^{l-1}K_m}{\sum_{m\in[M]}E_m^{l-1}\mathcal{E}(\mathcal{F};E_{[M]}^{l-1})}},$$

which leads to the lemma. □

Then, the following lemma bounds the estimated rewards $\widehat{\mathcal{R}}_m^l$ and true rewards $\mathcal{R}_m$.

Lemma 3.3.17. *For any epoch $l > 1$, for any $\pi_m \in \Psi_m$, conditioned on Υ^{l-1}, it holds that*

$$\left|\widehat{\mathcal{R}}_m^l(\pi_m\mid\Upsilon^{l-1})-\mathcal{R}_m(\pi_m)\right|\leq\sqrt{V_m(p_m^{l-1},\pi_m\mid\Upsilon^{l-1})}\sqrt{\mathcal{E}_m^{l-1}(\Upsilon^{l-1})},$$

where $\mathcal{E}_m^{l-1}(\Upsilon^{l-1}):=\mathbb{E}_{x_m\sim\mathcal{D}_m^{\mathcal{X}_m},a_m^{l-1}\sim p_m^{l-1}(\cdot|x_m)}\left[\left(\widehat{f}^l(x_m,a_m^{l-1})-f^*(x_m,a_m^{l-1})\right)^2\mid\Upsilon^{l-1}\right]$.

Proof. For simplicity, we abbreviate $\mathbb{E}_{x_m\sim\mathcal{D}_m^{\mathcal{X}_m},a_m^{l-1}\sim p_m^{l-1}(\cdot|x_m)}[\cdot]$ as $\mathbb{E}_{x_m,a_m^{l-1}}[\cdot]$, and for any policy $\pi_m\in\Psi_m$, and any epoch $l>1$, we define

$$\Delta_m^l(\pi_m(x_m)):=\widehat{f}^l(x_m,\pi_m(x_m))-f^*(x_m,\pi_m(x_m))$$

which indicates that

$$\widehat{\mathcal{R}}_m^l(\pi_m\mid\Upsilon^{l-1})-\mathcal{R}_m(\pi_m)=\mathbb{E}_{x_m}\left[\Delta_m^l(\pi_m(x_m)\mid\Upsilon^{l-1}\right],$$

and

$$\mathbb{E}_{x_m,a_m^{l-1}}\left[\left(\Delta_m^l(a_m^{l-1})\right)^2\mid\Upsilon^{l-1}\right]\geq\mathbb{E}_{x_m}\left[p_m^{l-1}(\pi_m(x_m)|x_m)\left(\Delta_m^l(\pi_m(x_m))\right)^2\mid\Upsilon^{l-1}\right].$$

Furthermore, conditioned on Υ^{l-1}, we can obtain that

$$
\begin{aligned}
&V_m(p_m^{l-1}, \pi_m \mid \Upsilon^{l-1}) \cdot \mathbb{E}_{x_m, a_m^{l-1}}\left[\left(\Delta_m^l(a_m^{l-1})\right)^2 \mid \Upsilon^{l-1}\right] \\
&= \mathbb{E}_{x_m}\left[\frac{1}{p_m^{l-1}(\pi_m(x_m)|x_m)} \mid \Upsilon^{l-1}\right] \mathbb{E}_{x_m, a_m^{l-1}}\left[\left(\Delta_m^l(a_m^{l-1})\right)^2 \mid \Upsilon^{l-1}\right] \\
&\geq \left(\mathbb{E}_{x_m}\left[\sqrt{\frac{1}{p_m^{l-1}(\pi_m(x_m)|x_m)}\mathbb{E}_{a_m^{l-1}}\left[\left(\Delta_m^l(a_m^{l-1})\right)^2\right]} \mid \Upsilon^{l-1}\right]\right)^2 \\
&\geq \left(\mathbb{E}_{x_m}\left[\sqrt{\frac{1}{p_m^{l-1}(\pi_m(x_m)|x_m)} p_m^{l-1}(\pi_m(x_m)|x_m)\left(\Delta_m^l(\pi_m(x_m))\right)^2} \mid \Upsilon^{l-1}\right]\right)^2 \\
&= \left(\mathbb{E}_{x_m}\left[\left|\Delta_m^l(\pi_m(x_m))\right| \mid \Upsilon^{l-1}\right]\right)^2 \\
&\geq \left|\widehat{\mathcal{R}}_m^l(\pi_m \mid \Upsilon^{l-1}) - \mathcal{R}_m(\pi_m)\right|^2.
\end{aligned}
$$

As a result, it holds that

$$
\left|\widehat{\mathcal{R}}_m^l(\pi_m \mid \Upsilon^{l-1}) - \mathcal{R}_m(\pi_m)\right| \leq \sqrt{V_m(p_m^{l-1}, \pi_m \mid \Upsilon^{l-1})}\sqrt{\mathcal{E}_m^{l-1}(\Upsilon^{l-1})},
$$

where the last step we use the definition that

$$
\mathcal{E}_m^{l-1}(\Upsilon^{l-1}) = \mathbb{E}_{x_m, a_m^{l-1}}\left[\left(\widehat{f}^l(x_m, a_m^{l-1}) - f^*(x_m, a_m^{l-1})\right)^2 \mid \Upsilon^{l-1}\right].
$$

This concludes the proof. □

Furthermore, the following lemma provides a characterization of the relation between the virtual loss $\widehat{\mathrm{Reg}}_m^l$ and the true loss Reg_m^l.

Lemma 3.3.18. *For any epochs $l \geq 1$, for any policies $\pi_{[M]} \in \Psi_{[M]}$, it holds that*

$$
\begin{aligned}
\sum_{m\in[M]} E_m^l \mathrm{Reg}_m(\pi_m) &\leq 2\sum_{m\in[M]} E_m^l \mathbb{E}_{\Upsilon^{l-1}}\left[\widehat{\mathrm{Reg}}_m^l(\pi_m \mid \Upsilon^{l-1})\right] + \eta^l, \\
\sum_{m\in[M]} E_m^l \mathbb{E}_{\Upsilon^{l-1}}\left[\widehat{\mathrm{Reg}}_m^l(\pi_m \mid \Upsilon^{l-1})\right] &\leq 2\sum_{m\in[M]} E_m^l \mathrm{Reg}_m(\pi_m) + \eta^l,
\end{aligned}
$$

with

$$
\eta^l := \frac{9c^2}{\gamma^l}\sum_{m\in[M]} E_m^l K_m.
$$

Proof. First, we note that for $l = 1$, it holds that

$$\sum_{m\in[M]} E_m^1 \mathrm{Reg}_m(\pi_m) \le \sum_{m\in[M]} E_m^1 \le \eta^1 = 9c^2 \sum_{m\in[M]} E_m^1 K_m;$$
$$\sum_{m\in[M]} E_m^1 \widehat{\mathrm{Reg}}_m^l(\pi_m) = 0 \le \eta^1 = 9c^2 \sum_{m\in[M]} E_m^1 K_m,$$

which means the lemma holds for the first epoch.

We then perform an inductive proof and start by assuming that for epoch $l-1$ and any policies $\pi_m \in \Psi_m$, it holds that

$$\sum_{m\in[M]} E_m^{l-1} \mathrm{Reg}_m(\pi_m) \le 2 \sum_{m\in[M]} E_m^{l-1} \mathbb{E}_{\Upsilon^{l-2}} \left[\widehat{\mathrm{Reg}}_m^{l-1}(\pi_m \mid \Upsilon^{l-2})\right] + \eta^{l-1}$$
$$\sum_{m\in[M]} E_m^{l-1} \mathbb{E}_{\Upsilon^{l-2}} \left[\widehat{\mathrm{Reg}}_m^{l-1}(\pi_m \mid \Upsilon^{l-2})\right] \le 2 \sum_{m\in[M]} E_m^{l-1} \mathrm{Reg}_m(\pi_m) + \eta^{l-1}.$$

Then, it can be observed that

$$\begin{aligned}
&\mathrm{Reg}_m(\pi_m) - \widehat{\mathrm{Reg}}_m^l(\pi_m \mid \Upsilon^{l-1}) \\
&= \mathcal{R}_m(\pi_m^*) - \mathcal{R}_m(\pi_m) - \left(\widehat{\mathcal{R}}_m^l(\widehat{\pi}_m^l \mid \Upsilon^{l-1}) - \widehat{\mathcal{R}}_m^l(\pi_m \mid \Upsilon^{l-1})\right) \\
&\le \mathcal{R}_m(\pi_m^*) - \mathcal{R}_m(\pi_m) - \left(\widehat{\mathcal{R}}_m^l(\pi_m^* \mid \Upsilon^{l-1}) - \widehat{\mathcal{R}}_m^l(\pi_m \mid \Upsilon^{l-1})\right) \\
&= \mathcal{R}_m(\pi_m^*) - \widehat{\mathcal{R}}_m^l(\pi_m^* \mid \Upsilon^{l-1}) + \widehat{\mathcal{R}}_m^l(\pi_m \mid \Upsilon^{l-1}) - \mathcal{R}_m(\pi_m) \\
&\overset{(a)}{\le} \sqrt{V_m(p_m^{l-1}, \pi_m^* \mid \Upsilon^{l-1})}\sqrt{\mathcal{E}_m^{l-1}(\Upsilon^{l-1})} + \sqrt{V_m(p_m^{l-1}, \pi_m \mid \Upsilon^{l-1})}\sqrt{\mathcal{E}_m^{l-1}(\Upsilon^{l-1})} \\
&\le \frac{V_m(p_m^{l-1}, \pi_m^* \mid \Upsilon^{l-1})}{8c\gamma^l} + \frac{V_m(p_m^{l-1}, \pi_m \mid \Upsilon^{l-1})}{8c\gamma^l} + 4c\gamma^l \mathcal{E}_m^{l-1}(\Upsilon^{l-1}) \\
&\overset{(b)}{\le} \frac{K_m + \gamma^{l-1}\widehat{\mathrm{Reg}}_m^{l-1}(\pi_m^* \mid \Upsilon^{l-1})}{8c\gamma^l} + \frac{K_m + \gamma^{l-1}\widehat{\mathrm{Reg}}_m^{l-1}(\pi_m \mid \Upsilon^{l-1})}{8c\gamma^l} + 4c\gamma^l \mathcal{E}_m^{l-1}(\Upsilon^{l-1}),
\end{aligned}$$

where inequality (a) is from Lemma 3.3.17 and inequality (b) is from Lemma 3.3.24.

Then, summing over all M agents, we can obtain that

$$\begin{aligned}
&\mathbb{E}_{\Upsilon^{l-1}}\left[\sum_{m\in[M]} E_m^l \left(\mathrm{Reg}_m(\pi_m) - \widehat{\mathrm{Reg}}_m^l(\pi_m \mid \Upsilon^{l-1})\right)\right] \\
&\le \frac{\sum_{m\in[M]} E_m^l K_m}{4c\gamma^l} + \frac{\gamma^{l-1}}{8c\gamma^l} \sum_{m\in[M]} E_m^l \mathbb{E}_{\Upsilon^{l-1}}\left[\widehat{\mathrm{Reg}}_m^{l-1}(\pi_m^* \mid \Upsilon^{l-1})\right] \\
&\quad + \frac{\gamma^{l-1}}{8c\gamma^l} \sum_{m\in[M]} E_m^l \mathbb{E}_{\Upsilon^{l-1}}\left[\widehat{\mathrm{Reg}}_m^{l-1}(\pi_m \mid \Upsilon^{l-1})\right] + 4c\gamma^l \sum_{m\in[M]} E_m^l \mathbb{E}_{\Upsilon^{l-1}}\left[\mathcal{E}_m^{l-1}(\Upsilon^{l-1})\right]
\end{aligned}$$

$$
\begin{aligned}
&\overset{(d)}{\leq} \frac{\sum_{m\in[M]} E_m^l K_m}{4c\gamma^l} + \frac{\bar{c}\gamma^{l-1}}{8c\gamma^l} \sum_{m\in[M]} E_m^{l-1}\mathbb{E}_{\Upsilon^{l-1}}\left[\widehat{\mathrm{Reg}}_m^{l-1}(\pi_m^* \mid \Upsilon^{l-1})\right] \\
&\quad + \frac{\bar{c}\gamma^{l-1}}{8c\gamma^l} \sum_{m\in[M]} E_m^{l-1}\mathbb{E}_{\Upsilon^{l-1}}\left[\widehat{\mathrm{Reg}}_m^{l-1}(\pi_m \mid \Upsilon^{l-1})\right] + 4c\gamma^l \sum_{m\in[M]} E_m^l \mathbb{E}_{\Upsilon^{l-1}}\left[\mathcal{E}_m^{l-1}(\Upsilon^{l-1})\right] \\
&\overset{(e)}{\leq} \frac{\sum_{m\in[M]} E_m^l K_m}{4c\gamma^l} + \frac{\bar{c}\gamma^{l-1}}{4c\gamma^l} \sum_{m\in[M]} E_m^{l-1}\mathrm{Reg}_m(\pi_m) + \frac{\bar{c}\gamma^{l-1}}{4c\gamma^l}\cdot\eta^{l-1} \\
&\quad + 4c\gamma^l \sum_{m\in[M]} E_m^l \mathbb{E}_{\Upsilon^{l-1}}\left[\mathcal{E}_m^{l-1}(\Upsilon^{l-1})\right] \\
&\overset{(f)}{\leq} \frac{\sum_{m\in[M]} E_m^l K_m}{4c\gamma^l} + \frac{1}{4}\sum_{m\in[M]} E_m^l \mathrm{Reg}_m(\pi_m) + \frac{9c^2\sum_{m\in[M]} E_m^l K_m}{4\gamma^l} + \frac{4c^2\sum_{m\in[M]} E_m^l K_m}{\gamma^l},
\end{aligned}
$$

where inequality (d) is from the definition $\bar{c} := \max_{m\in[M],l\in[2,l(T)]} E_m^l/E_m^{l-1}$. Inequality (e) is from the induction assumption that

$$
\begin{aligned}
\sum_{m\in[M]} E_m^{l-1}\mathbb{E}_{\Upsilon^{l-1}}\left[\widehat{\mathrm{Reg}}_m^{l-1}(\pi_m^* \mid \Upsilon^{l-1})\right] &= \sum_{m\in[M]} E_m^{l-1}\mathbb{E}_{\Upsilon^{l-2}}\left[\widehat{\mathrm{Reg}}_m^{l-1}(\pi_m^* \mid \Upsilon^{l-2})\right] \\
&\leq 2\sum_{m\in[M]} E_m^{l-1}\mathrm{Reg}_m(\pi_m^*) + \eta^{l-1} = \eta^{l-1}, \\
\sum_{m\in[M]} E_m^{l-1}\mathbb{E}_{\Upsilon^{l-1}}\left[\widehat{\mathrm{Reg}}_m^{l-1}(\pi_m \mid \Upsilon^{l-1})\right] &= \sum_{m\in[M]} E_m^{l-1}\mathbb{E}_{\Upsilon^{l-2}}\left[\widehat{\mathrm{Reg}}_m^{l-1}(\pi_m \mid \Upsilon^{l-2})\right] \\
&\leq 2\sum_{m\in[M]} E_m^{l-1}\mathrm{Reg}_m(\pi_m) + \eta^{l-1}.
\end{aligned}
$$

Inequality (f) is based on the definition $\underline{c} := \min_{m\in[M],l\in[2,l(T)]} E_m^l/E_m^{l-1}$, $c := \bar{c}/\underline{c}$ and $\eta^l := 9c^2\sum_{m\in[M]} E_m^l K_m/\gamma^l$, also the assumption that $\gamma^l \geq \gamma^{l-1}$ and Lemma 3.3.16, which indicates that

$$
\mathbb{E}_{\Upsilon^{l-1}}\left[\sum_{m\in[M]} E_m^{l-1}\mathcal{E}_m^{l-1}(\Upsilon^{l-1})\right] \leq \frac{\sum_{m\in[M]} E_m^{l-1}K_m}{(\gamma^l)^2}.
$$

Thus, we can obtain that

$$
\begin{aligned}
\frac{3}{4}\sum_{m\in[M]} E_m^l \mathrm{Reg}_m(\pi_m) &\leq \sum_{m\in[M]} E_m^l \mathbb{E}_{\Upsilon^{l-1}}\left[\widehat{\mathrm{Reg}}_m^{l}(\pi_m \mid \Upsilon^{l-1})\right] + \frac{\sum_{m\in[M]} E_m^l K_m}{4c\gamma^l} \\
&\quad + \frac{25c^2\sum_{m\in[M]} E_m^l K_m}{4\gamma^l} \\
\Rightarrow \sum_{m\in[M]} E_m^l \mathrm{Reg}_m(\pi_m) &\leq \frac{4}{3}\sum_{m\in[M]} E_m^l \mathbb{E}_{\Upsilon^{l-1}}\left[\widehat{\mathrm{Reg}}_m^{l}(\pi_m \mid \Upsilon^{l-1})\right] + \frac{\sum_{m\in[M]} E_m^l K_m}{3c\gamma^l} \\
&\quad + \frac{25c^2\sum_{m\in[M]} E_m^l K_m}{4\gamma^l}
\end{aligned}
$$

$$\leq 2 \sum_{m\in[M]} E_m^l \mathbb{E}_{\Upsilon^{l-1}} \left[\widehat{\mathrm{Reg}}_m^l(\pi_m \mid \Upsilon^{l-1})\right] + \eta^l$$

Also, it similarly holds that

$$\begin{aligned}
&\widehat{\mathrm{Reg}}_m^l(\pi_m \mid \Upsilon^{l-1}) - \mathrm{Reg}_m(\pi_m) \\
&= \widehat{\mathcal{R}}_m^l(\widehat{\pi}_m^l \mid \Upsilon^{l-1}) - \widehat{\mathcal{R}}_m^l(\pi_m \mid \Upsilon^{l-1}) - (\mathcal{R}_m(\pi_m^*) - \mathcal{R}_m(\pi_m)) \\
&\leq \widehat{\mathcal{R}}_m^l(\widehat{\pi}_m^l \mid \Upsilon^{l-1}) - \widehat{\mathcal{R}}_m^l(\pi_m \mid \Upsilon^{l-1}) - \big(\mathcal{R}_m(\widehat{\pi}_m^l) - \mathcal{R}_m(\pi_m)\big) \\
&= \widehat{\mathcal{R}}_m^l(\widehat{\pi}_m^l \mid \Upsilon^{l-1}) - \mathcal{R}_m(\widehat{\pi}_m^l) + \mathcal{R}_m(\pi_m) - \widehat{\mathcal{R}}_m^l(\pi_m \mid \Upsilon^{l-1}) \\
&\leq \sqrt{V_m(p_m^{l-1}, \widehat{\pi}_m^l \mid \Upsilon^{l-1})}\sqrt{\mathcal{E}_m^{l-1}(\Upsilon^{l-1})} + \sqrt{V_m(p_m^{l-1}, \pi_m \mid \Upsilon^{l-1})}\sqrt{\mathcal{E}_m^{l-1}(\Upsilon^{l-1})} \\
&\leq \frac{K_m + \gamma^{l-1}\widehat{\mathrm{Reg}}_m^{l-1}(\widehat{\pi}_m^l \mid \Upsilon^{l-1})}{8c\gamma^l} + \frac{K_m + \gamma^{l-1}\widehat{\mathrm{Reg}}_m^{l-1}(\pi_m \mid \Upsilon^{l-1})}{8c\gamma^l} + 4c\gamma^l \mathcal{E}_m^{l-1}(\Upsilon^{l-1}).
\end{aligned}$$

Then, summing over M agents, we can obtain that

$$\begin{aligned}
&\mathbb{E}_{\Upsilon^{l-1}}\left[\sum_{m\in[M]} E_m^l \left(\widehat{\mathrm{Reg}}_m^l(\pi_m \mid \Upsilon^{l-1}) - \mathrm{Reg}_m(\pi_m)\right)\right] \\
&\leq \frac{\sum_{m\in[M]} E_m^l K_m}{4c\gamma^l} + \frac{\overline{c}\gamma^{l-1}}{8c\gamma^l} \sum_{m\in[M]} E_m^{l-1} \mathbb{E}_{\Upsilon^{l-1}}\left[\widehat{\mathrm{Reg}}_m^{l-1}(\widehat{\pi}_m^l \mid \Upsilon^{l-1})\right] \\
&+ \frac{\overline{c}\gamma^{l-1}}{8c\gamma^l} \sum_{m\in[M]} E_m^{l-1} \mathbb{E}_{\Upsilon^{l-1}}\left[\widehat{\mathrm{Reg}}_m^{l-1}(\pi_m \mid \Upsilon^{l-1})\right] + 4c\gamma^l \sum_{m\in[M]} E_m^l \mathbb{E}_{\Upsilon^{l-1}}\left[\mathcal{E}_m^{l-1}(\Upsilon^{l-1})\right] \\
&\leq \frac{\sum_{m\in[M]} E_m^l K_m}{4c\gamma^l} + \frac{\overline{c}\gamma^{l-1}}{4c\gamma^l} \sum_{m\in[M]} E_m^{l-1} \mathbb{E}_{\Upsilon^{l-1}}\left[\mathrm{Reg}_m(\widehat{\pi}_m^l \mid \Upsilon^{l-1})\right] \\
&+ \frac{\overline{c}\gamma^{l-1}}{4c\gamma^l} \sum_{m\in[M]} E_m^{l-1} \mathrm{Reg}_m(\pi_m) + \frac{\overline{c}\gamma^{l-1}}{4c\gamma^l} \cdot \eta^{l-1} + 4c\gamma^l \sum_{m\in[M]} E_m^l \mathbb{E}_{\Upsilon^{l-1}}\left[\mathcal{E}_m^{l-1}(\Upsilon^{l-1})\right] \\
&\overset{(g)}{\leq} \frac{\sum_{m\in[M]} E_m^l K_m}{4c\gamma^l} + \frac{\gamma^{l-1}}{4\gamma^l} \cdot \eta^l + \frac{\gamma^{l-1}}{4\gamma^l} \sum_{m\in[M]} E_m^l \mathrm{Reg}_m(\pi_m) \\
&+ \frac{\overline{c}\gamma^{l-1}}{4c\gamma^l} \cdot \eta^{l-1} + 4c\gamma^l \sum_{m\in[M]} E_m^l \mathbb{E}_{\Upsilon^{l-1}}\left[\mathcal{E}_m^{l-1}(\Upsilon^{l-1})\right] \\
&\leq \frac{\sum_{m\in[M]} E_m^l K_m}{4c\gamma^l} + \frac{9c^2 \sum_{m\in[M]} E_m^l K_m}{4\gamma^l} + \frac{1}{4} \sum_{m\in[M]} E_m^l \mathrm{Reg}_m(\pi_m) \\
&+ \frac{9c^2 \sum_{m\in[M]} E_m^l K_m}{4\gamma^l} + \frac{4c^2 \sum_{m\in[M]} E_m^l K_m}{\gamma^l},
\end{aligned}$$

where inequality (g) is from the previous derivation that

$$\sum_{m\in[M]} E_m^{l-1}\mathrm{Reg}_m(\widehat{\pi}_m^l \mid \Upsilon^{l-1}) \le 2\underline{c}\sum_{m\in[M]} E_m^l\widehat{\mathrm{Reg}}_m^l(\widehat{\pi}_m^l \mid \Upsilon^{l-1}) + \underline{c}\eta^l = \underline{c}\eta^l$$

Thus, it holds that

$$\begin{aligned}\sum_{m\in[M]} E_m^l\mathbb{E}_{\Upsilon^{l-1}}\left[\widehat{\mathrm{Reg}}_m^{l-1}(\widehat{\pi}_m^l \mid \Upsilon^{l-1})\right] &\le \frac{5}{4}\sum_{m\in[M]} E_m^l\mathrm{Reg}_m(\pi_m)\\ &\quad + \frac{\sum_{m\in[M]} E_m^l K_m}{4c\gamma^l} + \frac{17c^2\sum_{m\in[M]} E_m^l K_m}{2\gamma^l}\\ \Rightarrow \sum_{m\in[M]} E_m^l\mathbb{E}_{\Upsilon^{l-1}}\left[\widehat{\mathrm{Reg}}_m^{l-1}(\widehat{\pi}_m^l \mid \Upsilon^{l-1})\right] &\le 2\sum_{m\in[M]} E_m^l\mathrm{Reg}_m(\pi_m) + \eta^l.\end{aligned}$$

With these two parts, the lemma can be obtained by induction. □

Furthermore, the following lemma provides a characterization of the per-epoch loss of the federation.

Lemma 3.3.19. *For every epoch $l > 1$, conditioned on Υ^{l-1}, it holds that*

$$\mathbb{E}_{\Upsilon^{l-1}}\left[\sum_{m\in[M]} E_m^l\sum_{\pi_m\in\Psi_m} Q_m^l(\pi_m \mid \Upsilon^{l-1})\mathrm{Reg}_m(\pi_m)\right] \le \frac{11c^2}{\gamma^l}\sum_{m\in[M]} E_m^l K_m,$$

where $Q^l(\cdot|\Upsilon^{l-1})$ is a probability measure on Ψ_m defined in Lemma 3.3.21

Proof. For any probability measures $\{\tilde{Q}_m^l(\cdot) : m \in [M]\}$, where $\tilde{Q}_m^l(\cdot)$ is on Ψ_M, it holds that

$$\begin{aligned}&\sum_{m\in[M]} E_m^l\sum_{\pi_m\in\Psi_m}\tilde{Q}_m^l(\pi_m)\mathrm{Reg}_m(\pi_m)\\ &\overset{(a)}{\le} 2\mathbb{E}_{\Upsilon^{l-1}}\left[\sum_{\pi_{[M]}\in\Psi_{[M]}}\tilde{Q}^l(\pi_{[M]})\sum_{m\in[M]} E_m^l\widehat{\mathrm{Reg}}_m(\pi_m \mid \Upsilon^{l-1})\right] + \eta^l\\ &= 2\mathbb{E}_{\Upsilon^{l-1}}\left[\sum_{m\in[M]} E_m^l\sum_{\pi_m\in\Psi_m}\tilde{Q}_m^l(\pi_m)\widehat{\mathrm{Reg}}_m(\pi_m \mid \Upsilon^{l-1})\right] + \eta^l,\end{aligned}$$

where inequality (a) is from Lemma 3.3.18 and $\tilde{Q}^l(\pi_{[M]}) := \prod_{m\in[M]}\tilde{Q}_m^l(\pi_m)$. Thus, we can obtain that

$$\begin{aligned}&\mathbb{E}_{\Upsilon^{l-1}}\left[\sum_{m\in[M]} E_m^l\sum_{\pi_m\in\Psi_m} Q_m^l(\pi_m \mid \Upsilon^{l-1})\mathrm{Reg}_m(\pi_m)\right]\\ &\le 2\mathbb{E}_{\Upsilon^{l-1}}\left[\sum_{m\in[M]} E_m^l\sum_{\pi_m\in\Psi_m} Q_m^l(\pi_m \mid \Upsilon^{l-1})\widehat{\mathrm{Reg}}_m(\pi_m \mid \Upsilon^{l-1})\right] + \eta^l\end{aligned}$$

$$\overset{(b)}{\leq} \frac{2}{\gamma^l} \sum_{m\in[M]} E_m^l K_m + \frac{9c^2}{\gamma^l} \sum_{m\in[M]} E_m^l K_m$$
$$\leq \frac{11c^2}{\gamma^l} \sum_{m\in[M]} E_m^l K_m,$$

where inequality (b) is from Lemma 3.3.23. □

With the previous lemmas, we can obtain the final Theorem 3.3.3, which is restated in the following.

Theorem 3.3.20 (Restatement of Theorem 3.3.3)**.** *Using a learning rate*

$$\gamma^l = O\left(\sqrt{\sum_{m\in[M]} E_m^{l-1} K_m / \left(\sum_{m\in[M]} E_m^{l-1} \mathcal{E}(E_{[M]}^{l-1})\right)}\right)$$

in epoch l, denoting $\bar{K}^l := \sum_{m\in[M]} E_m^l K_m / \sum_{m\in[M]} E_m^l$, the regret of FedIGW can be bounded as

$$\mathrm{Reg}(T) = O\left(\sum_{m\in[M]} E_m^1 + \sum_{l\in[2,l(T)]} c^{\frac{5}{2}} \sqrt{\bar{K}^l \mathcal{E}(E_{[M]}^{l-1})} \sum_{m\in[M]} E_m^l\right).$$

Proof of Theorem 3.3.3. The expected regret can be bounded as

$$\mathrm{Reg}(T) = \mathbb{E}\left[\sum_{m\in[M]} \sum_{t_m\in[T_m]} (f^*(x_{m,t_m}, \pi_m^*(x_{m,t_m})) - f^*(x_{m,t_m}, a_{m,t_m}))\right]$$
$$\leq \mathbb{E}\left[\sum_{l\in[2,l(T)]} \sum_{m\in[M]} \sum_{t_m\in[t_m(\tau^{l-1})+1, t_m(\tau^l)]} (f^*(x_{m,t_m}, \pi_m^*(x_{m,t_m})) - f^*(x_{m,t_m}, a_{m,t_m}))\right] + \sum_{m\in[M]} E_m^1$$
$$= \sum_{l\in[2,l(T)]} \mathbb{E}_{\Upsilon^{l-1}}\left[\mathbb{E}_{x_m, a_m^l}\left[\sum_{m\in[M]} E_m^l \left(f^*(x_m, \pi_m^*(x_m)) - f^*(x_m, a_m)\right) \mid \Upsilon^{l-1}\right] \mid \Upsilon^{l-1}\right] + \sum_{m\in[M]} E_m^1$$
$$\overset{(a)}{=} \sum_{l\in[2,l(T)]} \mathbb{E}_{\Upsilon^{l-1}}\left[\sum_{m\in[M]} E_m^l \sum_{\pi_m\in\Psi^m} Q_m^l(\pi_m \mid \Upsilon^{l-1}) \mathrm{Reg}_m(\pi_m) \mid \Upsilon^{l-1}\right] + \sum_{m\in[M]} E_m^1$$
$$\overset{(b)}{\leq} \sum_{l\in[2,l(T)]} \frac{11c^2}{\gamma^l} \sum_{m\in[M]} E_m^l K_m + \sum_{m\in[M]} E_m^1$$
$$\overset{(c)}{=} \sum_{l\in[2,l(T)]} 11c^2 \sqrt{\frac{\sum_{m\in[M]} E_m^{l-1} \mathcal{E}(\mathcal{F}; E_{[M]}^{l-1})}{\sum_{m\in[M]} E_m^{l-1} K_m}} \sum_{m\in[M]} E_m^l K_m + \sum_{m\in[M]} E_m^1$$
$$\leq \sum_{l\in[2,l(T)]} 11c^2 \sqrt{\overline{K} \mathcal{E}(\mathcal{F}; E_{[M]}^{l-1})} \sum_{m\in[M]} E_m^{l-1} + \sum_{m\in[M]} E_m^1,$$

where equality (a) is from Lemma 3.3.22, inequality (b) is from Lemma 3.3.19, and inequality (c) is from the choice of γ^l. The proof is then concluded. □

The following supporting lemmas can be similarly obtained by the corresponding proofs in Simchi-Levi and Xu (2022).

Lemma 3.3.21 (Lemma 3, Simchi-Levi and Xu (2022)). *For any epoch $l \in \mathbb{N}$, conditioned on Υ^{l-1}, there exists a probability measure $Q_m^l(\cdot|\Upsilon^{l-1})$ on Ψ_m such that*

$$\forall a_m \in \mathcal{A}_m, \forall x_m \in \mathcal{X}_m, \qquad p_m^l(a_m|x_m, \Upsilon^{l-1}) = \sum_{\pi_m \in \Psi_m} \mathbb{1}\{\pi_m(x_m) = a_m\} Q_m^l(\pi_m|\Upsilon^{l-1}).$$

Lemma 3.3.22 (Lemma 4, Simchi-Levi and Xu (2022)). *Fix any epoch $l \in \mathbb{N}$, we have*

$$\begin{aligned}&\mathbb{E}_{x_m \sim \mathcal{D}_m^{\mathcal{X}_m}, a_m^l \sim p_m^l(\cdot|x_m)} \left[f^*(x_m, \pi_m^*(x_m)) - f^*(x_m, a_m^l) \mid \Upsilon^{l-1} \right] \\ &\qquad = \sum_{\pi_m \in \Psi_m} Q_m^l(\pi_m \mid \Upsilon^{l-1}) \mathrm{Reg}_m(\pi_m).\end{aligned}$$

Lemma 3.3.23 (Lemma 5, Simchi-Levi and Xu (2022)). *Fix any epoch $l \in \mathbb{N}$, conditioned on Υ^{l-1}, we have*

$$\sum_{\pi \in \Psi_m} Q_m^l(\pi_m \mid \Upsilon^{l-1}) \widehat{\mathrm{Reg}}_m^l(\pi_m \mid \Upsilon^{l-1}) \leq \frac{K_m}{\gamma^l}.$$

Lemma 3.3.24 (Lemma 6, Simchi-Levi and Xu (2022)). *Fix any epoch $l \in \mathbb{N}$, for any policy $\pi_m \in \Psi_m$, we have*

$$V_m(p_m^l, \pi_m \mid \Upsilon^{l-1}) \leq K_m + \gamma^l \widehat{\mathrm{Reg}}_m^l(\pi_m \mid \Upsilon^{l-1}).$$

Reward Function Classes with Finite Cardinalities

First, with realizability, i.e., Assumption 3.3.1, the following characterization can be obtained.

Lemma 3.3.25 (Lemma 4.2, Agarwal et al. (2012)). *Fix a function $f \in \mathcal{F}$. Suppose we sample x_m, r_m from the data distribution $\mathcal{D}_m$, and an action a_m from an arbitrary distribution such that r_m and a_m are conditionally independent given x_m. Define the random variable*

$$\ell_m(f) := \left(f(x_m, a_m) - r_m(a_m)\right)^2 - \left(f^*(x_m, a_m) - r_m(a_m)\right)^2.$$

Then, we have

$$\mathbb{E}_{x_m, r_m, a_m} \left[\ell_m(f)\right] = \mathbb{E}_{x_m, a_m} \left[\left(f(x_m, a_m) - f^*(x_m, a_m)\right)^2 \right]$$

and

$$\mathbb{V}_{x_m,r_m,a_m}\left[\ell_m(f)\right] \le 4\mathbb{E}_{x_m,r_m,a_m}\left[\ell_m(f)\right],$$

where $\mathbb{V}[\cdot]$ *denotes the variance of a random variable.*

First, we establish the excess risk bound required in Assumption 3.3.2 via the following complete version of Lemma 3.3.4

Lemma 3.3.26 (Complete Version of Lemma 3.3.4)**.** *Under the setup of Assumption 3.3.2, if the adopted FL routine provides an exact minimizer for the optimization problem in Eqn.* (3.15) *with quadratic losses, i.e.,*

$$\widehat{f} = \arg\min_{f\in\mathcal{F}} \frac{1}{n}\sum_{m\in[M]}\sum_{i\in[n_m]}\left(f(x_m^i, a_m^i) - y_m^i\right)^2,$$

then, with probability at least $1-\delta$*, it holds that*

$$\sum_{m\in[M]}\frac{n_m}{n}\cdot\mathbb{E}_{x_m\sim\mathcal{D}_m^{\mathcal{X}_m},a_m\sim p_m(\cdot|x_m)}\left[\left(\widehat{f}(x_m,a_m) - f^*(x_m,a_m)\right)^2\right] \le \frac{25\log(|\mathcal{F}|/\delta)}{n}.$$

As a result, Assumption 3.3.2 holds with

$$\mathcal{E}(\delta, n_{[M]}) = O\left(\log(|\mathcal{F}|n)/n\right).$$

Proof. For simplicity, we abbreviate the quadratic loss associated with a fixed function $f\in\mathcal{F}$ as

$$\ell_m^i(f) = \ell_m(f(x_m^i,a_m^i); r_m^i) := \left(f(x_m^i,a_m^i) - r_m^i\right)^2, \qquad \forall m\in[M].$$

Then, with a probability at least $1-\delta$, for a fixed $f\in\mathcal{F}$, it holds that

$$\begin{aligned}
&\sum_{m\in[M]}\sum_{i_m\in[n_m]}\mathbb{E}_{x_m^i,r_m^i,a_m^i}\left[\ell_m^i(f) - \ell_m^i(f^*)\right] - \sum_{m\in[M]}\sum_{i\in[n_m]}\left[\ell_m^i(f) - \ell_m^i(f^*)\right]\\
&\overset{(a)}{\le} 2\sqrt{\sum_{m\in[M]}\sum_{i_m\in[n_m]}\mathbb{V}_{x_m^i,r_m^i,a_m^i}\left[\ell_m^i(f) - \ell_m^i(f^*)\right]\log(1/\delta)} + \frac{4}{3}\log(1/\delta)\\
&\overset{(b)}{\le} 4\sqrt{\sum_{m\in[M]}\sum_{i_m\in[n_m]}\mathbb{E}_{x_m^i,r_m^i,a_m^i}\left[\ell_m^i(f) - \ell_m^i(f^*)\right]\log(1/\delta)} + \frac{4}{3}\log(1/\delta),
\end{aligned}$$

where inequality (a) leverages Bernstein's inequality and inequality (b) is based on Lemma 3.3.25.

With

$$X(f) = \sqrt{\sum_{m\in[M]}\sum_{i_m\in[n_m]} \mathbb{E}_{x_m^i, r_m^i, a_m^i}\left[\ell_m^i(f) - \ell_{m,i}(f^*)\right]};$$
$$Z(f) = \sum_{m\in[M]}\sum_{i\in[n_m]} \left[\ell_m^i(f) - \ell_{m,i}(f^*)\right]; \qquad C = \sqrt{\log(1/\delta)}.$$

Applying a union bound to the above inequality indicates that with probability $1 - |\mathcal{F}|\delta$, for all $f \in \mathcal{F}$, it holds that

$$X(f)^2 - Z(f) \le 4CX(f) + \frac{4}{3}C^2 \quad \Rightarrow \quad (X(f) - 2C)^2 - Z(f) \le \frac{16}{3}C^2.$$

Since $\widehat{f}$ satisfies that $Z(\widehat{f}) \le 0$, we can obtain that

$$X(\widehat{f})^2 \le 25C^2,$$

In other words, with probability $1 - \delta$, it holds that

$$\sum_{m\in[M]}\sum_{i_m\in[n_m]} \mathbb{E}_{x_m^i, r_m^i, a_m^i}\left[\left(\widehat{f}(x_m^i, a_m^i) - r_m^i\right)^2 - \left(f^*(x_m^i, a_m^i) - r_m^i\right)^2\right]$$
$$= \sum_{m\in[M]} n_m \mathbb{E}_{x_m^i, a_m^i}\left[\left(\widehat{f}(x_m^i, a_m^i) - f^*(x_m^i, a_m^i)\right)^2\right] \le 25\log(|\mathcal{F}|/\delta),$$

where the equality is from the realizability in Assumption 3.3.1. The first half of the lemma is then proved.

With $\delta = 1/n$, the second half can be obtained as

$$\mathbb{E}_{S_{[M]}}\left[\sum_{m\in[M]} \frac{n_m}{n} \cdot \mathbb{E}_{x_m, a_m}\left[\left(\widehat{f}(x_m, a_m) - f^*(x_m, a_m)\right)^2\right]\right] \le \frac{25\log(|\mathcal{F}|n)}{n} + \frac{1}{n},$$

which concludes the proof. □

Based on the established excess risk bound, Corollary 3.3.5 can be obtained as follows.

Corollary 3.3.27 (Restatement of Corollary 3.3.5)**.** *If $|\mathcal{F}| < \infty$ and the adopted FL routine provides an exact minimizer for Eqn.* (3.15) *with quadratic losses, with $\tau^l = 2^l$, FedIGW incurs a regret of*

$$\mathrm{Reg}(T) = O(\sqrt{KMT\log(|\mathcal{F}|MT)})$$

and a total $O(\log(T))$ calls of the adopted FL routine.

Proof of Corollary 3.3.5. With Theorem 3.3.3 and Lemma 3.3.4, under the choice of $\tau^l = 2^l$, the regret can be bounded as

$$\begin{aligned}\mathrm{Reg}(T) &= O\left(ME^1 + \sum_{l\in[2,l(T)]}\sqrt{KME^l\log(|\mathcal{F}|ME^l)}\right)\\ &= O\left(\sum_{l\in[2,\lceil\log_2(T)\rceil]}\sqrt{KM2^l\log(|\mathcal{F}|MT)}\right)\\ &= O\left(\sqrt{KMT\log(|\mathcal{F}|MT)}\right),\end{aligned}$$

and the exponentially growing epoch length naturally leads to $O(\log(T))$ calls of the adopted FL routine, which concludes the proof. □

Reward Function Classes with Convex and Smooth Losses

In the following, we first prove Lemma 3.3.6 while also noting that this result is general and does not rely on the specific parameterization of $\mathcal{F}$, although we presented it with the d-dimensional parameterization.

Lemma 3.3.28 (Complete Version of Lemma 3.3.6)**.** *If the loss function $l_m(\cdot;\cdot)$ is μ_f-strongly convex in its first coordinate for all $m \in [M]$, i.e.,*

$$l_m(z_1';z_2) - l_m(z_1;z_2) \geq \frac{\mathrm{d}l_m(z_1;z_2)}{\mathrm{d}z_1}\cdot(z_1'-z_1) + \frac{\mu_f}{2}(z_1'-z_1)^2, \quad \textit{for any } z_1, z_1' \textit{ and } z_2,$$

and

$$\inf_{y\in\mathbb{R}}\mathbb{E}_{r_m}[l_m(y,r_m(a_m))|x_m,a_m] = \mathbb{E}_{r_m}[l(f_{\omega^*}(x_m,a_m),r_m(a_m))|x_m,a_m] \tag{3.16}$$

for all $m \in [M]$, $(x_m,a_m) \in \mathcal{X}_m \times \mathcal{A}_m$, then Assumption 3.3.2 holds with

$$\mathcal{E}(\mathcal{F};n_{[M]}) \geq 2\left(\varepsilon_{opt}(\mathcal{F};n_{[M]}) + \varepsilon_{gen}(\mathcal{F};n_{[M]})\right)/\mu_f,$$

where

$$\begin{aligned}\varepsilon_{gen}(\mathcal{F};n_{[M]}) &:= \mathbb{E}_{\mathcal{S},\xi}[\mathcal{L}(f_{\widehat{\omega}_{\mathcal{S}}}) - \widehat{\mathcal{L}}(f_{\widehat{\omega}_{\mathcal{S}}};\mathcal{S})];\\ \varepsilon_{opt}(\mathcal{F};n_{[M]}) &:= \mathbb{E}_{\mathcal{S},\xi}[\widehat{\mathcal{L}}(f_{\widehat{\omega}_{\mathcal{S}}};\mathcal{S}) - \widehat{\mathcal{L}}(f_{\omega^*_{\mathcal{S}}};\mathcal{S})].\end{aligned}$$

Proof. First, for any $\widehat{\omega}_{\mathcal{S}}$, it holds that

$$\begin{aligned}
&\mathcal{L}(f_{\widehat{\omega}_{\mathcal{S}}}) - \mathcal{L}(f_{\omega^*}) \\
&= \sum_{m\in[M]} \frac{n_m}{n} \mathbb{E}_{x_{m,i},a_{m,i},r_{m,i}} \left[\ell(f_{\widehat{\omega}_{\mathcal{S}}}(x_{m,i},a_{m,i});r_{m,i}) - \ell(f_{\omega^*}(x_{m,i},a_{m,i});r_{m,i})\right] \\
&\geq \frac{\mu_f}{2} \sum_{m\in[M]} \frac{n_m}{n} \mathbb{E}_{x_{m,i},a_{m,i}} \left[\left(f_{\widehat{\omega}_{\mathcal{S}}}(x_{m,i},a_{m,i}) - f_{\omega^*}(x_{m,i},a_{m,i})\right)^2\right]
\end{aligned}$$

where the inequality is due to the strong convexity of $\ell(\cdot;\cdot)$ w.r.t. its first coordinate and the optimality of f_{ω^*} assumed in Eqn. (3.16). Thus, we obtain that

$$\sum_{m\in[M]} \frac{n_m}{n} \mathbb{E}_{x_{m,i},a_{m,i}} \left[\left(f_{\widehat{\omega}_{\mathcal{S}}}(x_{m,i},a_{m,i}) - f_{\omega^*}(x_{m,i},a_{m,i})\right)^2\right] \leq \frac{2}{\mu_f} \left(\mathcal{L}(f_{\widehat{\omega}_{\mathcal{S}}}) - \mathcal{L}(f_{\omega^*})\right).$$

Furthermore, it holds that

$$\begin{aligned}
&\mathbb{E}_{\mathcal{S},\xi}\left[\mathcal{L}(f_{\widehat{\omega}_{\mathcal{S}}})\right] - \mathcal{L}(f_{\omega^*}) \\
&= \mathbb{E}_{\mathcal{S},\xi}\left[\mathcal{L}(f_{\widehat{\omega}_{\mathcal{S}}})\right] - \mathbb{E}_{\mathcal{S},\xi}\left[\widehat{\mathcal{L}}(f_{\widehat{\omega}_{\mathcal{S}}};\mathcal{S})\right] + \mathbb{E}_{\mathcal{S},\xi}\left[\widehat{\mathcal{L}}(f_{\widehat{\omega}_{\mathcal{S}}};\mathcal{S})\right] - \mathcal{L}(f_{\omega^*}) \\
&\leq \mathbb{E}_{\mathcal{S},\xi}\left[\mathcal{L}(f_{\widehat{\omega}_{\mathcal{S}}})\right] - \mathbb{E}_{\mathcal{S},\xi}\left[\widehat{\mathcal{L}}(f_{\widehat{\omega}_{\mathcal{S}}};\mathcal{S})\right] + \mathbb{E}_{\mathcal{S},\xi}\left[\widehat{\mathcal{L}}(f_{\widehat{\omega}_{\mathcal{S}}};\mathcal{S})\right] - \mathbb{E}_{\mathcal{S},\xi}\left[\widehat{\mathcal{L}}(f_{\omega^*_{\mathcal{S}}};\mathcal{S})\right],
\end{aligned}$$

where the last inequality is due to

$$\mathcal{L}(f_{\omega^*}) = \mathbb{E}_{\mathcal{S}}\left[\widehat{\mathcal{L}}(f_{\omega^*};\mathcal{S})\right] \geq \mathbb{E}_{\mathcal{S}}\left[\widehat{\mathcal{L}}(f_{\omega^*_{\mathcal{S}}};\mathcal{S})\right].$$

The proof is then concluded. □

Then, for the generalization error analyses, Lemma 3.3.7, restated below, follows standard proofs (e.g., Theorem 6.4 in Zhang (2023); Theorem 3.3 in Mohri et al. (2018)).

Lemma 3.3.29 (Restatement of Lemma 3.3.7)**.** *It holds that*

$$\varepsilon_{gen}(\mathcal{F};n_{[M]}) := \mathbb{E}_{\mathcal{S},\xi}[\mathcal{L}(f_{\widehat{\omega}_{\mathcal{S}}}) - \widehat{\mathcal{L}}(f_{\widehat{\omega}_{\mathcal{S}}};\mathcal{S})] \leq 2\mathfrak{R}(\mathcal{F};n_{[M]}),$$

where

$$\mathfrak{R}(\mathcal{F};n_{[M]}) = \sup\left\{\mathbb{E}_{\mathcal{S},\boldsymbol{\sigma}}\left[\sup_{\omega}\left\{\sum_{m\in[M]}\frac{1}{n}\sum_{i\in[n_m]}\sigma_{m,i}\cdot\ell_m(f_\omega(x_{m,i},a_{m,i});r_{m,i})\right\}\right]\right\},$$

where the outside supremum is over possible distributions of dataset $\mathcal{S}$ defined in Assumption 3.3.2.

The optimization error of FedAvg can be found in the Lemma 3.3.8. Combining the generalization error and optimization error via Lemma 3.3.6 into Theorem 3.3.3, Corollary 3.3.9 can be obtained, which is restated in the following.

Corollary 3.3.30 (Restatement of Corollary 3.3.9)**.** *Under the conditions of Lemmas 3.3.6 and 3.3.8, if FedAvg is used as the FL routine, the regret of FedIGW can be bounded as*

$$\mathrm{Reg}(T) = O\left(ME^1 + \sum\nolimits_{l\in[2,l(T)]} \sqrt{\frac{K}{\mu_f}\cdot\left(\mathfrak{R}^{l-1} + \frac{\sigma_b^2}{\mu_\omega \rho^{l-1}\kappa^{l-1}M} + \frac{\beta_\omega G_b^2}{\mu_\omega^2(\rho^{l-1})^2}\right)} ME^l\right),$$

where $\mathfrak{R}^l := \mathfrak{R}(\mathcal{F}; \{E^l : m \in [M]\})$ while ρ^l and κ^l the round of agents-server communications and local updates between in epoch l, respectively.

Proof. We can specify

$$\begin{aligned}\mathcal{E}(\mathcal{F}; \{E^l : m \in [M]\}) &= \frac{2}{\mu_f}\left(2\mathfrak{R}(\mathcal{F}; \{E^l : m \in [M]\}) + \tilde{O}\left(\frac{\sigma_b^2}{\mu_\omega \rho^l \kappa^l M} + \frac{\beta_\omega G_b^2}{\mu_\omega^2(\rho^l)^2}\right)\right)\\ &\geq \frac{2}{\mu_f}\left(\varepsilon_{\mathrm{gen}}(\mathcal{F}; \{E^l : m \in [M]\}) + \varepsilon_{\mathrm{opt}}(\mathcal{F}; \{E^l : m \in [M]\})\right),\end{aligned}$$

where the inequality is from Lemmas 3.3.7 and 3.3.8. This is a valid excess risk bound due to Lemma 3.3.6. Then, by plugging this excess risk bound into Theorem 3.3.3, the corollary is proved. □

Corollary 3.3.10 can be obtained by setting a suitable number of global aggregations for each epoch such that the optimization error is on the same order as the generalization error.

Corollary 3.3.31 (Restatement of Corollary 3.3.10)**.** *Under the conditions of Lemmas 3.3.6 and 3.3.8, with FedAvg as the adopted FL routine, FedIGW incurs a regret of*

$$\mathrm{Reg}(T) = O\left(ME^1 + \sum\nolimits_{l\in[2,l(T)]} \sqrt{K\mathfrak{R}^{l-1}/\mu_f} ME^l\right)$$

with

$$\tilde{O}\left(\sum_{l\in[l(T)]} \frac{\beta_\omega}{\mu_\omega} + \frac{\sigma_b^2}{\mu_\omega \mathfrak{R}^l \kappa^l M} + + \sqrt{\frac{\beta_\omega G_b^2}{\mu_\omega^2 \mathfrak{R}^l}}\right)$$

rounds of communications.

Proof. From Lemma 3.3.8, the optimization error in epoch l of form

$$\tilde{O}\left(\frac{\sigma_b^2}{\mu_\omega \rho^l \kappa^l M} + \frac{\beta_\omega G_b^2}{\mu_\omega^2 (\rho^l)^2}\right),$$

when $\rho^l = \Omega(\beta_\omega/\mu_\omega)$. Thus, if the communication rounds

$$\rho^l = \tilde{\Theta}\left(\frac{\beta_\omega}{\mu_\omega} + \frac{\sigma_b^2}{\mu_\omega \mathfrak{R}^l \kappa^l M} + \sqrt{\frac{\beta_\omega G_b^2}{\mu_\omega^2 \mathfrak{R}^l}}\right).$$

we are guaranteed to have the optimization error on the order of $O(\mathfrak{R}^l)$.

Then, the regret in Corollary 3.3.9 is of order

$$\mathrm{Reg}(T) = O\left(ME^1 + \sum\nolimits_{l\in[2,l(T)]} \sqrt{K\mathfrak{R}^{l-1}/\mu_f} ME^l\right)$$

while the overall communication rounds can be bounded as

$$\sum_{l\in[l(T)]} \rho^l = \tilde{O}\left(\sum_{l\in[l(T)]} \frac{\beta_\omega}{\mu_\omega} + \frac{\sigma_b^2}{\mu_\omega \mathfrak{R}^l \kappa^l M} + \sqrt{\frac{\beta_\omega G_b^2}{\mu_\omega^2 \mathfrak{R}^l}}\right),$$

which concludes the proof. □

A Linear Reward Function Class

We here provide a detailed discussion on the linear reward function class. Especially, following standard assumptions in linear bandits (Abbasi-Yadkori et al., 2011) and federated linear bandits (Li and Wang, 2022a; He et al., 2022; Amani et al., 2022), we consider $\mu_m(x_m, a_m) = \langle \phi(x_m, a_m), \omega^* \rangle$, where $\phi(\cdot)$ is a known d-dimensional mapping and ω^* is an unknown d-dimensional system parameter. Then, it is sufficient to consider a linear function class $\mathcal{F}$, where $f_\omega(\cdot) = \langle \omega, \phi(\cdot) \rangle$ and $f^*(\cdot) = \langle \omega^*, \phi(\cdot) \rangle$. Moreover, for convenience, we assume that $\|\phi(x_m, a_m)\|_2 \leq 1$ and $\|\omega^*\|_2 \leq 1$.

The FL problem can be formulated as a standard ridge regression with

$$\ell_m(f_\omega(x_m, a_m); r_m) := (\langle \omega, \phi(x_m, a_m) \rangle - r_m)^2 + \lambda \|\omega\|_2^2.$$

In other words, Eqn. (3.15) can be restated as

$$\min_{\omega\in\mathbb{R}^d} \widehat{\mathcal{L}}(f_\omega; \mathcal{S}) := \sum_{m\in[M]} \frac{1}{n} \sum_{i\in[n_m]} \left(\langle \omega, \phi(x_m^i, a_m^i) \rangle - r_m^i\right)^2 + \lambda \|\omega\|_2^2, \tag{3.17}$$

which has an exact minimizer as

$$\omega_{\mathcal{S}}^* = \left(\frac{1}{n} \sum_{m\in[M]} \sum_{i\in[n_m]} \phi(x_m^i, a_m^i)\phi(x_m^i, a_m^i)^\top + \lambda I \right)^{-1} \left(\frac{1}{n} \sum_{m\in[M]} \sum_{i\in[n_m]} \phi(x_m^i, a_m^i) r_m^i \right).$$

We provide an excess risk bound required in Assumption 3.3.2 through the following decomposition:

$$\begin{aligned}
&\mathbb{E}_{\mathcal{S},\xi} \left[\sum_{m\in[M]} \frac{n_m}{n} \mathbb{E}_{x_m,a_m} \left(\langle \widehat{\omega}_{\mathcal{S}}, \phi(x_m, a_m) \rangle - \langle \omega^*, \phi(x_m, a_m) \rangle \right)^2 \right] \\
&\leq 2\mathbb{E}_{\mathcal{S},\xi} \left[\sum_{m\in[M]} \frac{n_m}{n} \mathbb{E}_{x_m,a_m} \left(\langle \widehat{\omega}_{\mathcal{S}}, \phi(x_m, a_m) \rangle - \langle \omega_{\mathcal{S}}^*, \phi(x_m, a_m) \rangle \right)^2 \right] \\
&\quad + 2\mathbb{E}_{\mathcal{S},\xi} \left[\sum_{m\in[M]} \frac{n_m}{n} \mathbb{E}_{x_m,a_m} \left(\langle \omega_{\mathcal{S}}^*, \phi(x_m, a_m) \rangle - \langle \omega^*, \phi(x_m, a_m) \rangle \right)^2 \right] \\
&= 2\mathbb{E}_{\mathcal{S},\xi} \left[\|\widehat{\omega}_{\mathcal{S}} - \omega_{\mathcal{S}}^*\|_\Sigma^2 \right] \\
&\quad + 2\mathbb{E}_{\mathcal{S}} \left[\sum_{m\in[M]} \frac{n_m}{n} \mathbb{E}_{x_m,a_m} \left(\langle \omega_{\mathcal{S}}^*, \phi(x_m, a_m) \rangle - \langle \omega^*, \phi(x_m, a_m) \rangle \right)^2 \right] \\
&\leq 2\mathbb{E}_{\mathcal{S},\xi} \left[\lambda_{\max}(\Sigma) \|\widehat{\omega}_{\mathcal{S}} - \omega_{\mathcal{S}}^*\|_2^2 \right] && =: \text{term (A)} \\
&\quad + 2\mathbb{E}_{\mathcal{S}} \left[\sum_{m\in[M]} \frac{n_m}{n} \mathbb{E}_{x_m,a_m} \left(\langle \omega_{\mathcal{S}}^*, \phi(x_m, a_m) \rangle - \langle \omega^*, \phi(x_m, a_m) \rangle \right)^2 \right] && =: \text{term (B)}
\end{aligned}$$

where

$$\Sigma := \sum_{m\in[M]} \frac{n_m}{n} \mathbb{E}_{x_m,a_m} \left[\phi(x_m, a_m)\phi(x_m, a_m)^\top \right]$$

and $\lambda_{\max}(\Sigma)$ denotes the maximum eigenvalue of Σ. With $\|\phi(x, a)\|_2 \leq 1$, it can be verified that $\lambda_{\max}(\Sigma) \leq 1$. In the above decomposition, term (A) can be interpreted as the optimization error, while term (B) is the generalization error.

We can then plug in the aforementioned explicit formula of $\omega_{\mathcal{S}}^*$ into term (B) and demonstrate that term (B) $= \tilde{O}(d/n)$ with $\lambda = 1/n$ under the assumption that $\|\omega^*\|_2 \leq 1$ and $r_m \in [0, 1]$ (e.g., following Theorem 9.35 in Zhang (2023)). Then, with many efficient optimization algorithms (e.g., a distributed version of accelerated gradient descent (AGD)) (Nesterov, 2003)), it takes only $O(\sqrt{\kappa}\log(1/\varepsilon'))$ rounds of iterations (i.e., communications) to have an optimization error of ε', where κ is the condition number (i.e., the ratio between the smooth and strongly convex parameter in the considered problem). With $\lambda = 1/n$, it holds that $\kappa = O(n)$ and thus takes $O(\sqrt{n}\log(d/n))$ rounds of communications to obtain an optimization error of

order $\tilde{O}(d/n)$. Moreover, the adopted optimization algorithms (e.g., distributed AGD) typically only need to aggregate processed model parameters (e.g., gradients) with the server, which avoids communicating raw or compressed data (e.g., local variance matrices) as in previous federated linear bandit designs (Wang et al., 2020b; Dubey and Pentland, 2020; Li and Wang, 2022a; He et al., 2022; Amani et al., 2022).

With the above illustration, the following corollary is then a straightforward extension from Theorem 3.3.3.

Corollary 3.3.32. *In the considered linear reward function class with shared true parameters, using distributed AGD as the adopted FL routine to solve the FL problem in Eqn.* (3.17) *and* $\tau^l = 2^l$*, FedIGW obtains a regret of*

$$\mathrm{Reg}(T) = \tilde{O}\left(\sum_{l\in[\log_2(T)]} \sqrt{\frac{Kd}{M2^{l-1}}} M2^l\right) = \tilde{O}\left(\sqrt{MKdT}\right)$$

with

$$O\left(\sum_{l\in[\log_2(T)]} \sqrt{M2^l}\log(d/(M2^l))\right) = \tilde{O}(\sqrt{MT})$$

rounds of communications.

3.3.8 Full Proofs of the Flexible Appendages

In this section, additional details for the personalized learning setting are discussed. The overall algorithm structure still follows Algorithm 9. The major difference is that a personalized FL problem is considered:

$$\min_{\omega^\alpha, \omega^\beta_{[M]}} \widehat{\mathcal{L}}(f_{\omega^\alpha, \omega^\beta_{[M]}}; \mathcal{S}_{[M]}) := \sum_{m\in[M]} \frac{n_m}{n} \widehat{\mathcal{L}}_m(f_{\omega^\alpha, \omega^\beta_m}; \mathcal{S}_m),$$

where

$$\widehat{\mathcal{L}}_m(f_{\omega^\alpha, \omega^\beta_m}; \mathcal{S}_m) := \frac{1}{n_m} \sum_{i\in[n_m]} \ell_m(f_{\omega^\alpha, \omega^\beta_m}(x^i_m, a^i_m); r^i_m).$$

Proof of Corollary 3.3.15

The proof of Corollary 3.3.15 largely follows those of Corollary 3.3.10: decomposing excess risk to generalization and optimization errors; using Rademacher complexity to characterize the generalization error; using FL convergence analyses to characterize the optimization error; and combining them together such that the optimization error does not dominate the generalization error.

The first major difference is that a slightly different Rademacher complexity is introduced:

$$\mathfrak{P}(\mathcal{F}_{[M]}; n_{[M]}) = \sup \left\{ \mathbb{E}_{\mathcal{S},\boldsymbol{\sigma}} \left[\sup_{\omega^\alpha, \omega^\beta_{[M]}} \left\{ \sum_{m\in[M]} \frac{1}{n} \sum_{i\in[n_m]} \sigma_{m,i} \cdot \ell_m(f_{\omega_m}(x_m^i, a_m^i); r_m^i) \right\} \right] \right\},$$

which is suitable for the considered personalized setting with parameters $[\omega^\alpha, \omega^\beta_{[M]}]$ involved. A similar notation is also adopted in Mohri et al. (2019). Moreover, as the LSGD-PFL algorithm (Hanzely et al., 2021) is adopted to solve the personalized FL task as an illustration, its corresponding convergence analyses should be incorporated, which is presented in Lemma 3.3.14 and restated as Lemma 3.3.40. With these two parts ready, Corollary 3.3.15, restated in the following, can be obtained similarly to Corollary 3.3.10.

Corollary 3.3.33 (Restatement of Corollary 3.3.15)**.** *Under the conditions of Lemmas 3.3.6 and 3.3.14, with LSGD-PFL as the adopted personalized FL routine, FedIGW incurs a regret of*

$$\mathrm{Reg}(T) = O\left(ME^1 + \sum\nolimits_{l\in[2,l(T)]} \sqrt{K\mathfrak{P}^{l-1}/\mu_f} ME^l \right)$$

with

$$\tilde{O}\left(\sum_{l\in[l(T)]} \frac{\max\{\beta_{\omega^\beta}(\kappa^l))^{-1}, \beta_{\omega^\alpha}\}}{\mu_\omega} + \frac{\sigma_b^2}{\mu_\omega \kappa^l M \mathfrak{P}^l} + \sqrt{\frac{\beta_{\omega^\alpha}(G^2+\sigma^2)}{\mu_\omega^2 \mathfrak{P}^l}} \right)$$

rounds of communications, where $\mathfrak{P}^l := \mathfrak{P}(\mathcal{F}_{[M]}, \{E^l : m \in [M]\})$ *and* κ^l *is the number of local updates in epoch* l*.*

A Linear Reward Function Class

As an extension of the linear reward function in Appendix 3.3.7, we consider that

$$\mu_m(x_m, a_m) = \langle \phi(x_m, a_m), \omega_m^* \rangle, \qquad \forall m \in [M], (x_m, a_m) \in \mathcal{X}_m \times \mathcal{A}_m,$$

and the true model parameters $\{\omega_m^* : m \in [M]\}$ follow Assumption 3.3.13, i.e., $\omega_m^* = [\omega^{\alpha,*}, \omega_m^{*,\beta}]$ with $\omega^{\alpha,*}$ shared among all agents.

It can be further realized that the above problem setting is identical to a $\tilde{d}$-dimensional linear system, where $\tilde{d} := d^\alpha + \sum_{m\in[M]} d_m^\beta$: the overall true model parameter is

$$\tilde{\omega}^* = \left[\omega^{*,\alpha}, \omega_1^{*,\beta}, \cdots, \omega_M^{*,\beta} \right] \in \mathbb{R}^{\tilde{d}}.$$

and a correspondingly feature mapping $\tilde{\phi}(\cdot)$ is

$$\tilde{\phi}(x_m, a_m) = \left[\phi(x_m, a_m)_{[1:d^\alpha]}, \boldsymbol{O}_{d_1^\beta}, \cdots, \boldsymbol{O}_{d_{m-1}^\beta}, \phi(x_m, a_m)_{[d^\alpha+1:d_m]}, \boldsymbol{O}_{d_{m+1}^\beta}, \cdots, \boldsymbol{O}_{d_M^\beta}\right],$$

i.e., an expanded version of the original feature, where $\phi(x_m, a_m)_{[i:j]} \in \mathbb{R}^{j-i+1}$ denotes the sub-vector containing $[i:j]$-th elements in $\phi(x_m, a_m)$ and $\boldsymbol{O}_i \in \mathbb{R}^i$ an i-dimensional null vector.

With this reformulated problem, discussions from Appendix 3.3.7 can be directly leveraged. Especially, Corollary 3.3.32 indicates the following result.

Corollary 3.3.34. *In the considered linear reward function class with partially true parameters, using distributed AGD as the adopted FL routine to solve the FL problem in Eqn.* (3.17) *with reformulated feature mapping $\tilde{\phi}(\cdot)$ and $\tau^l = 2^l$, FedIGW incurs a regret of*

$$\mathrm{Reg}(T) = \tilde{O}\left(\sqrt{MK\tilde{d}T}\right)$$

with $\tilde{O}(\sqrt{MT})$ rounds of communications.

Robustness, Privacy, and Beyond

We here provide some additional discussions on incorporating appendages in FL studies to provide robustness and privacy guarantees for FedIGW among some other directions (e.g., fairness guarantees (Mohri et al., 2019; Du et al., 2021), client selections (Balakrishnan et al., 2022; Fraboni et al., 2021), and practical communication designs (Chen et al., 2021; Wei and Shen, 2022; Zheng et al., 2020)). Following the unified principle that "**FCB = FL + CB**", we can develop the corresponding versions of FedIGW and the associated theoretical analyses following the comprehensive example involving personalized learning.

The key is that as long as one FL routine can provide an estimated function $\widehat{f}$ (which is used in IGW interactions), it can be adopted in FedIGW; thus the desirable properties of the selected FL routine are naturally inherited to FedIGW. For example, Yin et al. (2018); Pillutla et al. (2022); Fu et al. (2019); Li et al. (2021); Zhu et al. (2023) studied how to handle malicious agents, who can deviate arbitrarily from the FL protocol and tamper with their own updates, during learning. The commonly adopted scheme is to invoke certain robust estimators (e.g., median and trimmed mean). Under suitable assumptions, existing approaches have shown that as long as the proportion of malicious agents does not exceed a threshold (typically, $1/2$), the estimators calculated by federation can still converge within certain amounts of error due to the malicious agents. A recent work (Zhu et al., 2023) provides a summary of convergence rates with different robust estimators, which can be leveraged to establish theoretical understandings of FedIGW with robustness.

On the privacy side, many mechanisms have also been studied in FL (Wei et al., 2020; Yin et al., 2021; Liu et al., 2022), to guarantee differential privacy (DP), where the most common approach is to insert noises of suitable scales. Convergence rates have also been established under suitable assumptions, e.g., in Wei et al. (2020); Girgis et al. (2021); Wei et al. (2021). With those analyses, the theoretical behavior of FedIGW with DP can also be similarly established as Corollaries 3.3.10 and 3.3.15.

3.3.9 Omitted Details of FL Designs

FedAvg

The FedAvg algorithm (McMahan et al., 2017) is one of the most standard and well-adopted FL designs. Following it, agents perform local stochastic gradient descents (SGD) with their local objective functions for certain steps and then communicate the updated local models to the server; the server aggregates local models to a global one via a weighted average, which is then communicated to the agents to perform further local SGDs.

Many theoretical analyses have been provided for FedAvg (e.g., Li et al. (2020b)). We adopt the one from Karimireddy et al. (2020) as Lemma 3.3.8, whose complete version is provided in the following.

Lemma 3.3.35 (Complete Version of Lemma 3.3.8; Theorem V in Karimireddy et al. (2020) without client sampling)**.** *For any dataset $\mathcal{S}$, if*

- *$\widehat{\mathcal{L}}_m(f_\omega; \mathcal{S}_m)$ is μ_ω-strongly convex w.r.t. ω (see Definition 3.3.36) for all $m \in [M]$;*
- *$\widehat{\mathcal{L}}_m(f_\omega; \mathcal{S}_m)$ is β_ω-smooth w.r.t. ω (see Definition 3.3.37) for all $m \in [M]$;*
- *the stochastic gradients are unbiased and have a σ_b^2-bounded variance (see Definition 3.3.38);*
- *the gradients have G_b-bounded dissimilarity (see Definition 3.3.39),*

with FedAvg as the adopted FL routine, the output $\widehat{\omega}$ satisfies that

$$\mathbb{E}_\xi[\widehat{\mathcal{L}}(f_{\widehat{\omega}_\mathcal{S}}; \mathcal{S}) - \widehat{\mathcal{L}}(f_{\omega^*_\mathcal{S}}; \mathcal{S}) \mid \mathcal{S}] \leq \tilde{O}\left(\frac{\sigma_b^2}{\mu_\omega \rho \kappa M} + \frac{\beta_\omega G_b^2}{\mu_\omega^2 \rho^2} + \mu_\omega \|\omega^0 - \omega^*_\mathcal{S}\|_2^2 \exp\left(-\frac{\mu_\omega \rho}{16\beta_\omega}\right)\right)$$

when $\rho \geq \frac{8\beta_\omega}{\mu_\omega}$, where ρ denotes the round of communications (i.e., number of global aggregations), κ is the number of local updates (i.e., SGD) between each communication, and ω^0 is the initialization. Note that the last term which decays exponentially w.r.t. ρ is omitted in Lemma 3.3.8 and the following derivations for simplicity.

A few definitions used above are made precise in the following, which are inherited from Karimireddy et al. (2020) and presented here for completeness:

Definition 3.3.36 (Strongly Convex). *$\widehat{\mathcal{L}}_m(f_\omega;\mathcal{S})$ is μ_ω-strongly convex w.r.t. ω for $\mu_\omega > 0$ if*

$$\widehat{\mathcal{L}}_m(f_{\omega'};\mathcal{S}) - \widehat{\mathcal{L}}_m(f_\omega;\mathcal{S}) \geq \left\langle \nabla_\omega \widehat{\mathcal{L}}_m(f_\omega;\mathcal{S}), \omega' - \omega \right\rangle + \frac{\mu_\omega}{2} \|\omega' - \omega\|_2^2, \quad \text{for any } \omega \text{ and } \omega'.$$

Definition 3.3.37 (Smooth). *$\widehat{\mathcal{L}}_m(f_\omega;\mathcal{S})$ is β_ω-smooth w.r.t. ω for $\beta_\omega > 0$ if*

$$\widehat{\mathcal{L}}_m(f_{\omega'};\mathcal{S}) - \widehat{\mathcal{L}}_m(f_\omega;\mathcal{S}) \leq \left\langle \nabla_\omega \widehat{\mathcal{L}}_m(f_\omega;\mathcal{S}), \omega' - \omega \right\rangle + \frac{\beta_\omega}{2} \|\omega' - \omega\|_2^2, \quad \text{for any } \omega \text{ and } \omega'.$$

Definition 3.3.38 (Stochastic Gradients with Bounded Variances). *The stochastic gradients have a σ_b^2-bounded variance if*

$$\frac{1}{n_m} \sum_{i\in[n_m]} \left\| \nabla_\omega \ell_m(f_\omega(x_m^i, a_m^i); r_m^i) - \nabla_\omega \widehat{\mathcal{L}}_m(f_\omega;\mathcal{S}_m) \right\|_2^2 \leq \sigma_b^2, \quad \text{for any } \omega \text{ and } m.$$

Definition 3.3.39 (Gradients with Bounded Dissimilarity). *The gradients have a G_b-bounded dissimilarity if*

$$\frac{1}{M} \sum_{m\in[M]} \left\| \nabla_\omega \widehat{\mathcal{L}}_m(f_\omega;\mathcal{S}_m) \right\|_2^2 \leq G_b^2, \quad \text{for any } \omega.$$

LSGD-PFL

The LSGD-PFL algorithm is summarized in Hanzely et al. (2021), which is a general design for personalized federated learning problems. It largely follows FedAvg (McMahan et al., 2017), while only the globally shared parameters are communicated and aggregated. The following lemma, a complete version of Lemma 3.3.14, is provided in Hanzely et al. (2021) to characterize the convergence of LSGD-PFL.

Lemma 3.3.40 (Complete Version of Lemma 3.3.14; Theorem 1 Hanzely et al. (2021)). *For any dataset $\mathcal{S}$, if*

- *$\widehat{\mathcal{L}}_m(f_{\omega_m};\mathcal{S}_m)$ is μ_ω-strongly convex w.r.t. ω_m (see Definition 3.3.36) for all $m \in [M]$;*
- *$\widehat{\mathcal{L}}_m(f_{\omega^\alpha,\omega_m^\beta};\mathcal{S}_m)$ is β_{ω^α}-smooth w.r.t. ω^α and $M\beta_{\omega^\beta}$-smooth w.r.t. ω_m^β (see Definition 3.3.37) for all $m \in [M]$;*
- *the stochastic gradients w.r.t. ω^α is unbiased and have a σ_b^2-bounded variance (see Definition 3.3.38);*
- *the stochastic gradients w.r.t. $\{\omega_m^\beta : m \in [M]\}$ is unbiased and have a σ_b^2-bounded variance (see Definition 3.3.38);*
- *the gradients w.r.t. ω have G_b bounded dissimilarity (see Definition 3.3.39),*

with LSGD-PFL as the adopted FL routine, the output $\widehat{\omega}$ *has* $\varepsilon_{opt}(\mathcal{F}_{[M]}; n_{[M]}) \leq \varepsilon'$ *after*

$$\tilde{O}\left(\frac{\max\{\beta_{\omega^\beta}\kappa^{-1}, \beta_{\omega^\alpha}\}}{\mu_\omega} + \frac{\sigma_b^2}{\mu_\omega \kappa M \varepsilon'} + \frac{1}{\mu_\omega}\sqrt{\frac{\beta_{\omega^\alpha}(G^2+\sigma^2)}{\varepsilon'}}\right)$$

rounds of communications, where κ *is the number of local updates.*

Chapter 4

Conclusions

In this dissertation, we explored decision-making in multi-agent systems under various scenarios and environments. This research aimed to provide fundamental insights into how to design communication and collaboration strategies in different agent settings.

For communication designs in Chapter 2, we focused on two main aspects: effectiveness (Section 2.2), and robustness (Section 2.3), under the problem of multi-player multi-armed bandits (MPMAB). By studying these contexts, we gained valuable insights into the challenges and opportunities associated with communication designs in multi-agent decision-making problems. Our findings highlighted the importance of leveraging tools established in broader communication communities, in particular, information-theoretic studies.

For collaboration designs in Chapter 3, focusing on the federated multi-armed bandits problem, our investigation encompassed decision-making studies when handling different global-local relationships (Section 3.1) and varying generalization-personalization balances (Section 3.2), and also provided a modulized approach to flexible involve established FL schemes. These scenarios posed unique challenges that required us to explore novel decision-making algorithms. By delving into these areas, we shed light on the general collaboration principles in multi-agent decision-making applications.

Overall, this dissertation contributes to the growing field of decision-making designs in multi-agent systems. By addressing the communication and collaboration problems, we have expanded the understanding of decision-making dynamics in complex agent interactions, which may further bring fundamental insights for real-world applications, such as autonomous systems, distributed networks, and economic markets.

Bibliography

Abbasi-Yadkori, Y., Pál, D., and Szepesvári, C. (2011). Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24.

Abe, N. and Long, P. M. (1999). Associative reinforcement learning using linear probabilistic concepts. In *ICML*, pages 3–11. Citeseer.

Agarwal, A., Dudík, M., Kale, S., Langford, J., and Schapire, R. (2012). Contextual bandit learning with predictable rewards. In *Artificial Intelligence and Statistics*, pages 19–26. PMLR.

Agarwal, A., Langford, J., and Wei, C.-Y. (2020). Federated residual learning. *arXiv preprint arXiv:2003.12880*.

Alatur, P., Levy, K. Y., and Krause, A. (2020). Multi-player bandits: The adversarial case. *Journal of Machine Learning Research*, 21.

Amani, S., Lattimore, T., György, A., and Yang, L. F. (2022). Distributed contextual linear bandits with minimax optimal communication cost. *arXiv preprint arXiv:2205.13170*.

Anandkumar, A., Michael, N., and Tang, A. (2010). Opportunistic spectrum access with multiple users: Learning under competition. In *2010 Proceedings IEEE INFOCOM*, pages 1–9. IEEE.

Anandkumar, A., Michael, N., Tang, A. K., and Swami, A. (2011). Distributed algorithms for learning and cognitive medium access with logarithmic regret. *IEEE Journal on Selected Areas in Communications*, 29(4):731–745.

Anantharam, V., Varaiya, P., and Walrand, J. (1987). Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays-part i: Iid rewards. *IEEE Transactions on Automatic Control*, 32(11):968–976.

Auer, P., Cesa-Bianchi, N., and Fischer, P. (2002). Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256.

Auer, P. and Ortner, R. (2010). Ucb revisited: Improved regret bounds for the stochastic multi-armed bandit problem. *Periodica Mathematica Hungarica*, 61(1-2):55–65.

Avner, O. and Mannor, S. (2014). Concurrent bandits and cognitive radio networks. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 66–81. Springer.

Avner, O. and Mannor, S. (2016). Multi-user lax communications: a multi-armed bandit approach. In *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9. IEEE.

Balakrishnan, R., Li, T., Zhou, T., Himayat, N., Smith, V., and Bilmes, J. (2022). Diverse client selection for federated learning via submodular maximization. In *International Conference on Learning Representations*.

Bande, M. and Veeravalli, V. V. (2019). Multi-user multi-armed bandits for uncoordinated spectrum access. In *2019 International Conference on Computing, Networking and Communications (ICNC)*, pages 653–657. IEEE.

Bartlett, P., Bousquet, O., and Mendelson, S. (2005). Local rademacher complexities. *Annals of Statistics*, 33(4):1497–1537.

Besson, L. and Kaufmann, E. (2018). Multi-player bandits revisited. In *Algorithmic Learning Theory*, pages 56–92.

Bistritz, I., Baharav, T., Leshem, A., and Bambos, N. (2020). My fair bandit: Distributed learning of max-min fairness with multi-player bandits. In *International Conference on Machine Learning*, pages 930–940. PMLR.

Bistritz, I. and Leshem, A. (2020). Game of thrones: Fully distributed learning for multiplayer bandits. *Mathematics of Operations Research*.

Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., Kiddon, C., Konecny, J., Mazzocchi, S., McMahan, H. B., Overveldt, T. V., Petrou, D., Ramage, D., and Roselander, J. (2019). Towards federated learning at scale: System design. In *Proceedings of the 2nd SysML Conference*, pages 1–15.

Boursier, E., Kaufmann, E., Mehrabian, A., and Perchet, V. (2020). A practical algorithm for multiplayer bandits when arm means vary among players. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, Palermo, Sicily, Italy.

Boursier, E. and Perchet, V. (2019). Sic-mmab: synchronisation involves communication in multiplayer multi-armed bandits. In *Advances in Neural Information Processing Systems*, pages 12071–12080.

Brânzei, S. and Peres, Y. (2019). Multiplayer bandit learning, from competition to cooperation. *arXiv preprint arXiv:1908.01135*.

Bubeck, S., Li, Y., Peres, Y., and Sellke, M. (2020). Non-stochastic multi-player multi-armed bandits: Optimal rate with collision information, sublinear without. In *Conference on Learning Theory*, pages 961–987.

Cantador, I., Brusilovsky, P., and Kuflik, T. (2011). 2nd Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec 2011). In *Proceedings of the 5th ACM Conference on Recommender Systems*, RecSys 2011, New York, NY, USA. ACM.

Chen, M., Gündüz, D., Huang, K., Saad, W., Bennis, M., Feljan, A. V., and Poor, H. V. (2021). Distributed learning in wireless networks: Recent progress and future challenges. *IEEE Journal on Selected Areas in Communications*, 39(12):3579–3605.

Chen, W., Hu, W., Li, F., Li, J., Liu, Y., and Lu, P. (2016a). Combinatorial multi-armed bandit with general reward functions. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pages 1659–1667.

Chen, W., Wang, Y., and Yuan, Y. (2013). Combinatorial multi-armed bandit: General framework and applications. In *International Conference on Machine Learning*, pages 151–159.

Chen, W., Wang, Y., Yuan, Y., and Wang, Q. (2016b). Combinatorial multi-armed bandit and its extension to probabilistically triggered arms. *The Journal of Machine Learning Research*, 17(1):1746–1778.

Combes, R., Talebi Mazraeh Shahi, M. S., Proutiere, A., et al. (2015). Combinatorial bandits revisited. *Advances in neural information processing systems*, 28:2116–2124.

Cortes, D. (2018). Adapting multi-armed bandits policies to contextual bandits scenarios. *arXiv preprint arXiv:1811.04383*.

Dai, Z., Shu, Y., Verma, A., Fan, F. X., Low, B. K. H., and Jaillet, P. (2023). Federated neural bandit. *The Eleventh International Conference on Learning Representations*.

Darak, S. J. and Hanawal, M. K. (2019). Multi-player multi-armed bandits for stable allocation in heterogeneous ad-hoc networks. *IEEE Journal on Selected Areas in Communications*, 37(10):2350–2363.

Demirel, I., Yildirim, Y., and Tekin, C. (2022). Federated multi-armed bandits under byzantine attacks. *arXiv preprint arXiv:2205.04134*.

Deng, Y., Kamani, M. M., and Mahdavi, M. (2020). Adaptive personalized federated learning. *arXiv preprint arXiv:2003.13461*.

Du, W., Xu, D., Wu, X., and Tong, H. (2021). Fairness-aware agnostic federated learning. In *Proceedings of the 2021 SIAM International Conference on Data Mining (SDM)*, pages 181–189. SIAM.

Dubey, A. and Pentland, A. (2020). Differentially-private federated linear bandits. *Advances in Neural Information Processing Systems*, 33:6003–6014.

Foster, D. and Rakhlin, A. (2020). Beyond ucb: Optimal and efficient contextual bandits with regression oracles. In *International Conference on Machine Learning*, pages 3199–3210. PMLR.

Fraboni, Y., Vidal, R., Kameni, L., and Lorenzi, M. (2021). Clustered sampling: Low-variance and improved representativity for clients selection in federated learning. In *International Conference on Machine Learning*, pages 3407–3416. PMLR.

Fu, S., Xie, C., Li, B., and Chen, Q. (2019). Attack-resistant federated learning with residual-based reweighting. *arXiv preprint arXiv:1912.11464*.

Gai, Y., Krishnamachari, B., and Jain, R. (2010). Learning multiuser channel allocations in cognitive radio networks: A combinatorial multi-armed bandit formulation. In *2010 IEEE Symposium on New Frontiers in Dynamic Spectrum (DySPAN)*, pages 1–9. IEEE.

Gallager, R. G. (1968). *Information theory and reliable communication*, volume 2. Springer.

Ghosh, A., Sankararaman, A., and Ramchandran, K. (2021). Model selection for generic contextual bandits. *arXiv preprint arXiv:2107.03455*.

Girgis, A., Data, D., Diggavi, S., Kairouz, P., and Suresh, A. T. (2021). Shuffled model of differential privacy in federated learning. In *International Conference on Artificial Intelligence and Statistics*, pages 2521–2529. PMLR.

Goldsmith, A. (2005). *Wireless communications*. Cambridge university press.

Goldsmith, A. J. and Chua, S.-G. (1998). Adaptive coded modulation for fading channels. *IEEE Transactions on communications*, 46(5):595–602.

Hanzely, F. and Richtárik, P. (2020). Federated learning of a mixture of global and local models. *arXiv preprint arXiv:2002.05516*.

Hanzely, F., Zhao, B., and Kolar, M. (2021). Personalized federated learning: A unified framework and universal optimization techniques. *arXiv preprint arXiv:2102.09743*.

Harper, F. M. and Konstan, J. A. (2015). The MovieLens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems*, 5(4).

He, J., Wang, T., Min, Y., and Gu, Q. (2022). A simple and provably efficient algorithm for asynchronous federated contextual linear bandits. *Advances in neural information processing systems*.

Hillel, E., Karnin, Z., Koren, T., Lempel, R., and Somekh, O. (2013). Distributed exploration in multi-armed bandits. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*, pages 854–862.

Hsu, D., Kakade, S. M., and Zhang, T. (2012). Random design analysis of ridge regression. In *Conference on learning theory*, pages 9–1. JMLR Workshop and Conference Proceedings.

Huang, B., Li, X., Song, Z., and Yang, X. (2021a). Fl-ntk: A neural tangent kernel-based framework for federated learning analysis. In *International Conference on Machine Learning*, pages 4423–4434. PMLR.

Huang, R., Wu, W., Yang, J., and Shen, C. (2021b). Federated linear contextual bandits. *Advances in neural information processing systems*, 34:27057–27068.

Jadbabaie, A., Li, H., Qian, J., and Tian, Y. (2022). Byzantine-robust federated linear bandits. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, pages 5206–5213. IEEE.

Kalathil, D., Nayyar, N., and Jain, R. (2014). Decentralized learning for multiplayer multiarmed bandits. *IEEE Transactions on Information Theory*, 60(4):2331–2345.

Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S., Stich, S., and Suresh, A. T. (2020). Scaffold: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*, pages 5132–5143. PMLR.

Katakis, I., Tsoumakas, G., and Vlahavas, I. (2008). Multilabel text classification for automated tag suggestion. *ECML PKDD discovery challenge*, 75:2008.

Konečnỳ, J., McMahan, H. B., Ramage, D., and Richtárik, P. (2016). Federated optimization: Distributed machine learning for on-device intelligence. *arXiv preprint arXiv:1610.02527*.

Krishnamurthy, S. K., Hadad, V., and Athey, S. (2021). Adapting to misspecification in contextual bandits with offline regression oracles. In *International Conference on Machine Learning*, pages 5805–5814. PMLR.

Kveton, B., Wen, Z., Ashkan, A., and Szepesvari, C. (2015). Tight regret bounds for stochastic combinatorial semi-bandits. In *Artificial Intelligence and Statistics*, pages 535–543.

Lai, T. L. and Robbins, H. (1985). Asymptotically efficient adaptive allocation rules. *Adv. Appl. Math.*, 6(1):4–22.

Lattimore, T. and Szepesvári, C. (2020). *Bandit algorithms*. Cambridge University Press.

Li, C. and Wang, H. (2022a). Asynchronous upper confidence bound algorithms for federated linear bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 6529–6553. PMLR.

Li, C. and Wang, H. (2022b). Communication efficient federated learning for generalized linear bandits. *Advances in Neural Information Processing Systems*.

Li, C., Wang, H., Wang, M., and Wang, H. (2022). Communication efficient distributed learning for kernelized contextual bandits. *Advances in Neural Information Processing Systems*.

Li, C., Wang, H., Wang, M., and Wang, H. (2023). Learning kernelized contextual bandits in a distributed and asynchronous environment. *The Eleventh International Conference on Learning Representations*.

Li, T., Hu, S., Beirami, A., and Smith, V. (2021). Ditto: Fair and robust federated learning through personalization. In *International Conference on Machine Learning*, pages 6357–6368. PMLR.

Li, T., Sahu, A. K., Talwalkar, A., and Smith, V. (2020a). Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60.

Li, T. and Song, L. (2022). Privacy-preserving communication-efficient federated multi-armed bandits. *IEEE Journal on Selected Areas in Communications*, 40(3):773–787.

Li, X., Huang, K., Yang, W., Wang, S., and Zhang, Z. (2020b). On the convergence of fedavg on non-iid data. In *International Conference on Learning Representations*.

Liu, K. and Zhao, Q. (2010). Distributed learning in multi-armed bandit with multiple players. *IEEE Transactions on Signal Processing*, 58(11):5667–5681.

Liu, Z., Guo, J., Yang, W., Fan, J., Lam, K.-Y., and Zhao, J. (2022). Privacy-preserving aggregation in federated learning: A survey. *IEEE Transactions on Big Data*.

Lugosi, G. and Mehrabian, A. (2018). Multiplayer bandits without observing collision information. *arXiv preprint arXiv:1808.08416*.

Magesh, A. and Veeravalli, V. V. (2019). Multi-user mabs with user dependent rewards for uncoordinated spectrum access. In *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, pages 969–972. IEEE.

Marfoq, O., Neglia, G., Kameni, L., and Vidal, R. (2023). Federated learning for data streams. *arXiv preprint arXiv:2301.01542*.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR.

Merlis, N. and Mannor, S. (2020). Tight lower bounds for combinatorial multi-armed bandits. In *Conference on Learning Theory*, pages 2830–2857. PMLR.

Mitra, A., Adibi, A., Pappas, G. J., and Hassani, H. (2022). Collaborative linear bandits with adversarial agents: Near-optimal regret bounds. *Advances in neural information processing systems*.

Mo, J. and Walrand, J. (2000). Fair end-to-end window-based congestion control. *IEEE/ACM Transactions on networking*, 8(5):556–567.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. (2018). *Foundations of machine learning*. MIT press.

Mohri, M., Sivek, G., and Suresh, A. T. (2019). Agnostic federated learning. In *International Conference on Machine Learning*, pages 4615–4625. PMLR.

Munkres, J. (1957). Algorithms for the assignment and transportation problems. *Journal of the society for industrial and applied mathematics*, 5(1):32–38.

Nayyar, N., Kalathil, D., and Jain, R. (2016). On regret-optimal learning in decentralized multiplayer multiarmed bandits. *IEEE Transactions on Control of Network Systems*, 5(1):597–606.

Nesterov, Y. (2003). *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media.

Pillutla, K., Kakade, S. M., and Harchaoui, Z. (2022). Robust aggregation for federated learning. *IEEE Transactions on Signal Processing*, 70:1142–1154.

Rosenski, J., Shamir, O., and Szlak, L. (2016). Multi-player bandits–a musical chairs approach. In *International Conference on Machine Learning*, pages 155–163.

Sen, R., Rakhlin, A., Ying, L., Kidambi, R., Foster, D., Hill, D. N., and Dhillon, I. S. (2021). Top-k extreme contextual bandits with arm hierarchy. In *International Conference on Machine Learning*, pages 9422–9433. PMLR.

Shi, C. and Shen, C. (2021a). Federated multi-armed bandits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 9603–9611.

Shi, C. and Shen, C. (2021b). On no-sensing adversarial multi-player multi-armed bandits with collision communications. *IEEE Journal on Selected Areas in Information Theory*, 2(2):515–533.

Shi, C., Xiong, W., Shen, C., and Yang, J. (2020). Decentralized multi-player multi-armed bandits with no collision information. In *International Conference on Artificial Intelligence and Statistics*, pages 1519–1528. PMLR.

Shi, C., Xiong, W., Shen, C., and Yang, J. (2021). Heterogeneous multi-player multi-armed bandits: Closing the gap and generalization. *Advances in neural information processing systems*, 34:22392–22404.

Simchi-Levi, D. and Xu, Y. (2022). Bypassing the monster: A faster and simpler optimal algorithm for contextual bandits under realizability. *Mathematics of Operations Research*, 47(3):1904–1931.

Smith, V., Chiang, C.-K., Sanjabi, M., and Talwalkar, A. S. (2017). Federated multi-task learning. In *Advances in Neural Information Processing Systems*, pages 4424–4434.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Tallini, L. G., Al-Bassam, S., and Bose, B. (2002). On the capacity and codes for the Z-channel. In *Proceedings of the IEEE International Symposium on Information Theory*, page 422.

Tibrewal, H., Patchala, S., Hanawal, M. K., and Darak, S. J. (2019). Multiplayer multi-armed bandits for optimal assignment in heterogeneous networks. *arXiv preprint arXiv:1901.03868*.

Tsoumakas, G., Katakis, I., and Vlahavas, I. (2008). Effective and efficient multilabel classification in domains with large number of labels. In *Proc. ECML/PKDD 2008 Workshop on Mining Multidimensional Data (MMD'08)*, volume 21, pages 53–59.

Vazirani, V. V. (2013). *Approximation algorithms*. Springer Science & Business Media.

Wang, P.-A., Proutiere, A., Ariu, K., Jedra, Y., and Russo, A. (2020a). Optimal algorithms for multiplayer multi-armed bandits. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, Palermo, Sicily, Italy.

Wang, Q. and Chen, W. (2017). Improving regret bounds for combinatorial semi-bandits with probabilistically triggered arms and its applications. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 1161–1171.

Wang, S. and Chen, W. (2018). Thompson sampling for combinatorial semi-bandits. In *International Conference on Machine Learning*, pages 5114–5122.

Wang, Y., Hu, J., Chen, X., and Wang, L. (2020b). Distributed bandit learning: Near-optimal regret with efficient communication. In *2020 International Conference on Learning Representations*.

Wei, K., Li, J., Ding, M., Ma, C., Su, H., Zhang, B., and Poor, H. V. (2021). User-level privacy-preserving federated learning: Analysis and performance optimization. *IEEE Transactions on Mobile Computing*, 21(9):3388–3401.

Wei, K., Li, J., Ding, M., Ma, C., Yang, H. H., Farokhi, F., Jin, S., Quek, T. Q., and Poor, H. V. (2020). Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Transactions on Information Forensics and Security*, 15:3454–3469.

Wei, X. and Shen, C. (2022). Federated learning over noisy channels: Convergence analysis and design examples. *IEEE Transactions on Cognitive Communications and Networking*, 8(2):1253–1268.

Xu, Y. and Zeevi, A. (2020). Upper counterfactual confidence bounds: a new optimism principle for contextual bandits. *arXiv preprint arXiv:2007.07876*.

Yin, D., Chen, Y., Kannan, R., and Bartlett, P. (2018). Byzantine-robust distributed learning: Towards optimal statistical rates. In *International Conference on Machine Learning*, pages 5650–5659. PMLR.

Yin, X., Zhu, Y., and Hu, J. (2021). A comprehensive survey of privacy-preserving federated learning: A taxonomy, review, and future directions. *ACM Computing Surveys (CSUR)*, 54(6):1–36.

Zhang, T. (2023). *Mathematical Analysis of Machine Learning Algorithms*. Cambridge University Press.

Zheng, S., Shen, C., and Chen, X. (2020). Design and analysis of uplink and downlink communications for federated learning. *IEEE Journal on Selected Areas in Communications*, 39(7):2150–2167.

Zhou, X. and Chowdhury, S. R. (2023). On differentially private federated linear contextual bandits. *arXiv preprint arXiv:2302.13945*.

Zhu, B., Wang, L., Pang, Q., Wang, S., Jiao, J., Song, D., and Jordan, M. I. (2023). Byzantine-robust federated learning with optimal statistical rates. In *International Conference on Artificial Intelligence and Statistics*, pages 3151–3178. PMLR.

Zhu, Z., Zhu, J., Liu, J., and Liu, Y. (2020). Federated bandit: A gossiping approach. *arXiv preprint arXiv:2010.12763*.