Swap-Regret-Minimizing Bandits for Distributed Network Optimization

by

Zhiming Huang

B.Eng., Northwestern Polytechnical University, 2018
M.Sc., University of Victoria, 2020

A Dissertation Submitted in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Computer Science

We acknowledge and respect the lək̓ʷəŋən (Songhees and Xʷsepsəm/Esquimalt) Peoples on whose territory the university stands, and the lək̓ʷəŋən and W̱SÁNEĆ Peoples whose historical relationships with the land continue to this day.

Swap-Regret-Minimizing Bandits for Distributed Network Optimization

by

Zhiming Huang

B.Eng., Northwestern Polytechnical University, 2018
M.Sc., University of Victoria, 2020

Supervisory Committee

Dr. Jianping Pan, Supervisor
(Department of Computer Science)

Dr. Nishant Mehta, Departmental Member
(Department of Computer Science)

Dr. Hongchuan Yang, Outside Member
(Department of Electrical and Computer Engineering)

ABSTRACT

Modern networked systems—ranging from real-time communication platforms to distributed computing infrastructures—operate in increasingly dynamic and strategic environments, where traditional optimization methods often fall short. This dissertation develops a new algorithmic framework for distributed network optimization grounded in game-theoretic bandit learning. We model fundamental problems, such as congestion control and resource allocation, as repeated games involving strategic agents who receive only partial (bandit) feedback. Motivated by practical challenges in computer networks, we design and analyze algorithms that not only minimize regret but also steer collective behavior toward equilibrium.

The contributions of this dissertation are threefold. First, we propose a new framework based on swap-regret minimization and online mirror descent, and establish high-probability regret bounds in multi-player bandit settings. These results guarantee convergence to correlated equilibria under decentralized, partial-information feedback. Second, we introduce optimistic learning techniques to accelerate convergence by leveraging predictability in the environment. Third, we apply our algorithms to real-world networking tasks, including TCP congestion control, and demonstrate improved stability, throughput, and fairness through extensive trace-driven emulations.

Together, these contributions bridge the theoretical foundations of online learning and game theory with practical considerations in network protocol design, offering robust tools for decentralized decision-making in uncertain and adversarial environments.

Table of Contents

List of Tables

List of Figures

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my parents, who have always stood behind me with unwavering support. They have never questioned my choices, but instead offered silent encouragement and trust in everything I chose to pursue. Their love and quiet strength have been the foundation of my academic journey, and I am endlessly grateful for their presence in my life.

Next, I want to express my deepest gratitude to my PhD supervisor, Professor Jianping Pan. He was the one who first introduced me to research and patiently guided me in developing the ability to think critically, work rigorously, and pursue work that makes a real impact. His mentorship has been instrumental not only in shaping my academic growth but also in supporting my personal and professional development. Whenever I faced important life decisions, I could always count on his thoughtful advice and unwavering support. He has been a true mentor in every sense—his wisdom, generosity, and encouragement have left a lasting mark on me.

I would also like to thank Professor Nishant Mehta for his continuous support in my research on machine learning theory. His insights and feedback have been invaluable, and he played a key role in shaping my interest in theoretical research. He made me feel welcome in the department's theory community, frequently inviting me to seminars and discussions from which I benefited greatly. He also looked out for me during academic conferences—introducing me to researchers, and checking in with encouragement. His support extended beyond research, including my scholarship applications and career planning, and I'm deeply grateful for his generosity.

Finally, I would like to thank my collaborators and friends. I am especially grateful to Dr. Bingshan Hu and Dr. Yifan Xu for their patience, support, and willingness to collaborate with me despite my occasional carelessness and lack of attention to detail. Their help has been crucial throughout our work together. I also want to thank my longtime friends—Haomin Yu, Yubo Huang, Zichang He, and Yuqing Yang—for their companionship and encouragement over the years. From high school and undergraduate days until now, your friendship has made this journey far more bearable and meaningful. I am also grateful to all my labmates and friends at UVic, whose presence made daily life more enjoyable and whose support lightened the challenges of graduate school. Thank you for the countless small moments that helped carry me through.

Chapter 1

Introduction

Many natural and engineered systems can be understood as dynamic networks, where each node represents an autonomous, decision-making entity, such as a user, device, or institution. These entities interact over time in ways shaped by uncertainty, partial information, and evolving local contexts. Rather than adhering to centralized control, each agent adapts its behavior in response to its environment and to the observed or anticipated actions of others. The global behavior of the system emerges from these decentralized interactions, often stabilizing toward forms of equilibrium that reflect a balance of incentives, constraints, and learning dynamics among agents. Understanding when and how such equilibria arise—whether cooperative, competitive, or mixed—is central to designing systems that are both robust and efficient. This perspective unifies ideas from computation, game theory, and complex systems, offering a principled approach to reasoning about coordination, conflict, and adaptation in strategic environments. These ideas motivate us to study how local learning dynamics can drive global system behavior in real-world networks.

Our research begins with congestion control, a classical yet persistently challenging problem at the heart of networked systems. Despite decades of study and its seemingly intuitive structure, an ideal solution that balances efficiency, fairness, and robustness remains elusive. Congestion control has immense practical importance. It directly affects how billions of users experience Internet-based services such as web browsing, video streaming, and real-time communication. As modern networked systems become increasingly complex—spanning data centers, mobile devices, and edge infrastructure—the demand for congestion control mechanisms that are efficient, fair, and adaptable is more urgent than ever. Traditional

heuristic-based protocols, while scalable, often lack robustness under dynamic and heterogeneous conditions. This underscores the need for principled methods grounded in learning theory that can adapt to local feedback, operate without centralized control, and provide performance guarantees.

As we delved deeper into this domain, it became increasingly apparent that many foundational questions, particularly those concerning decentralized decision-making and feedback, remain unresolved. This realization led to a broader line of inquiry throughout my doctoral research: exploring the theoretical underpinnings of distributed learning and game-theoretic dynamics, and the development of principled algorithms that bridge theory and practical deployment.

Although congestion control serves as the primary motivation, the techniques and insights developed in this dissertation extend well beyond networking, with potential applications in economics, multi-agent systems, and large-scale decision-making under uncertainty. In what follows, I outline the motivation behind this dissertation, present the research problems, summarize our key contributions, and provide an overview of the structure.

1.1 A Motivation Example – TCP Congestion Control

Congestion arises in all networked systems when competing traffic exceeds the limited communication capacity. Among these systems, one of the most critical to the functioning of modern society is computer networking, ranging from local area networks to the global Internet. At the heart of this infrastructure lies *Transmission Control Protocol (TCP)*.

TCP is a fundamental protocol used for reliable data transmission on the Internet. Beyond ensuring that data is delivered correctly and in order, TCP plays a crucial role in congestion control, regulating how much data are injected into the network at any time. Each TCP connection, or TCP flow, independently adjusts its sending rate based on local feedback such as packet loss or throughput, without direct knowledge of the overall network state. This decentralized adaptation enables TCP to respond to congestion dynamically; however, it also means that the collective behavior of multiple flows can lead to complex interactions and inefficiencies, particularly when resources are shared or limited.

Modern TCP congestion control often relies on heuristic-based mechanisms that

have enabled the Internet to scale effectively [67]. However, these methods frequently fall short of achieving optimal efficiency or fairness in increasingly complex and heterogeneous network environments. Consider a scenario where multiple TCP flows share a common bottleneck link. Each flow independently adjusts its transmission rate based on local feedback, such as packet loss or delay, yet lacks visibility into the global network state or the behavior of other flows.

This decentralized setting naturally invites a game-theoretic interpretation: each flow can be viewed as a self-interested player aiming to optimize its own performance (e.g., throughput or latency) without centralized coordination. Game theory offers a principled framework for analyzing such strategic interactions among multiple players. In the context of networked systems, users or protocols can be modeled as players in a game, where each player selects a strategy, such as a transmission rate, and receives a payoff determined by the collective behavior of all players.

This perspective motivates a central question: **Can game-theoretic tools provide stronger theoretical guarantees than traditional heuristic-based methods for guiding individual TCP flows to adapt their behavior in a decentralized manner, such that the system converges toward an equilibrium?** Ideally, such equilibria would promote fair bandwidth sharing among flows and lead to globally efficient network performance. Addressing this question calls for bridging the gap between the theoretical foundations of game theory and the practical constraints of real-world network environments.

1.2 Equilibrium Learning in Games

A central goal in game theory is to characterize equilibria, i.e., stable outcomes in which no player has an incentive to deviate. The most well-known concept is the Nash equilibrium (NE), where each player's strategy is a best response to others, i.e., no players would deviate unilaterally. However, while Nash equilibria capture rational behavior in principle, they often suffer from inefficiencies in practice and can be computationally intractable in large-scale systems.

In contrast, the correlated equilibrium (CE) is a broader solution concept that often yields better social performance. A CE is defined as a joint distribution over actions from which a central coordinator privately recommends an action to each

player. The key requirement is that no player has an incentive to unilaterally deviate from the recommendation, assuming that others follow theirs. This equilibrium notion relaxes the independence assumption of NE and allows for implicit coordination through correlation.

Traditional methods for computing equilibria assume full knowledge of the game and centralized computation—assumptions that break down in dynamic and large-scale environments [62]. In practice, especially in networking scenarios such as TCP congestion control, players only have access to local feedback and act independently. This has motivated a shift toward learning-based approaches that approximate equilibria over time through repeated interactions.

Recent developments in online learning have uncovered a deep connection between regret minimization and equilibrium computation. At a high level, regret quantifies the performance gap between a learner's decisions and a competitor's decisions. The most fundamental notion is external regret (see Definition 2.5), which measures how much worse the learner performs compared to the competitor who always plays the best fixed action in retrospect. In repeated games, if each player minimizes external regret, the empirical strategy profile converges to a Nash equilibrium in two-player zero-sum games (see Theorem 2.1).

A stronger notion, known as swap regret (see Definition 2.6), compares the learner's performance against competitors with all alternative strategies defined by swap functions, which remap actions to potentially better alternatives. If every player minimizes swap regret, the time-averaged joint distribution of play converges to a correlated equilibrium in an ergodic sense, as established in Theorem 2.2.

This learning-based viewpoint enables equilibrium computation through fully decentralized, feedback-driven dynamics, making it both scalable and applicable to real-world multi-agent systems where centralized control is infeasible.

However, while most existing results are established under the full-information feedback, where the losses or rewards of all actions, including unchosen ones, are observable, the corresponding guarantees under bandit feedback remain less understood. In many practical systems, such as TCP congestion control, players can only observe the outcome of their own actions (e.g., the experienced delay or packet loss from a selected transmission rate), making bandit feedback a more realistic model. This partial observability poses significant challenges for equilibrium learning, particularly for minimizing swap regret, which plays a central

role in ensuring convergence to correlated equilibria. In this work, we focus on bridging this gap by studying swap regret minimization in the bandit setting, with the goal of developing learning algorithms that are both theoretically sound and practically applicable to distributed network protocols.

1.3 Contributions

The research presented in this dissertation is informed by a sequence of earlier studies during my doctoral training that deepened my understanding of online learning and networked systems. My initial work focused on regret minimization in bandit problems with combinatorial actions and stochastic rewards [50, 51], and later expanded to more complex adversarial settings [54, 56]. While these prior projects addressed distinct subproblems, they played a formative role in developing the theoretical perspective and algorithmic tools that laid the groundwork for this dissertation.

Building on this foundation, the present work takes a unified view of swap-regret minimization under bandit feedback and its implications for decentralized learning and control. We develop a series of algorithms and analyses that not only advance the theoretical understanding of high-probability regret bounds, but also demonstrate how game-theoretic learning dynamics can be applied to the design of distributed protocols in networked systems. The key contributions are:

1. We establish a high-probability upper bound on the instantaneous swap regret under bandit feedback, showing that it scales as $O\left(A_n\sqrt{T\ln\left(\frac{A_n}{\delta}\right)}\right)$ for A_n actions over T rounds, with probability at least $1-\delta$.

   To the best of our knowledge, this is the tightest known swap-regret bound available in the bandit setting. Building on this result, we further show that if each player adopts the proposed swap-regret-minimizing algorithm, the empirical joint distribution of play converges to a correlated equilibrium with high probability.

   This contribution is notable because existing bounds—typically stated in expectation—do not guarantee convergence in a single realization. In contrast, our high-probability framework offers stronger and more practical guarantees for decentralized learning, bridging a critical gap between regret min-

imization and equilibrium computation in bandit settings. These results were published in UAI 2023 [52] and IEEE/ACM Transactions on Networking [55].

2. We design new bandit algorithms that achieve improved swap-regret bounds with reduced dependence on the time horizon T. By leveraging the fact that each player runs a swap-regret-minimizing algorithm—making their behavior predictable—we reduce the time dependence of the regret from the standard $O(\sqrt{T})$ to $\tilde{O}(T^{1/4})$, significantly accelerating convergence to correlated equilibrium.

   This result highlights the potential of adaptivity in bandit learning, offering both stronger theoretical guarantees and enhanced empirical performance in decentralized settings. We note that our approach does not use fully bandit feedback, because we take expectations on the observed rewards over other players' actions, which also leads to an open question of whether similar improvements can be achieved in a fully bandit setting. The work appears in IEEE INFOCOM 2025 [57].

3. We bridge theory and systems by demonstrating that the game-theoretic models can inform practical protocol design in real-world systems. In particular, we apply our theoretical insights to TCP congestion control by implementing the proposed swap-regret-minimizing algorithm in the Linux Kernel 5.13.12, using the congestion control plane architecture [80].

   Through trace-driven emulation across diverse network topologies, we show that the resulting protocol improves both throughput and fairness, underscoring the practical impact of learning-based approaches in real-world networking environments. These results were published in IEEE ICDCS 2023 [53] and IEEE/ACM Transactions on Networking [55].

While the core contributions of this dissertation are theoretical, they carry meaningful implications for real-world system design. They reflect a broader principle explored throughout this work: in complex systems, global order need not be dictated—it can emerge through learning.

Thus, embedding learning-based decision-making into network protocols offers a promising path toward self-optimizing infrastructure systems that adapt and improve over time through continual interaction, rather than relying on manually tuned heuristics. The algorithms and insights developed in this work lay the

foundation for scalable, adaptive, and principled solutions across a range of settings, including computer networks, cloud resource management, and multi-agent coordination.

Beyond networking, these methods extend to broader domains such as economics and societal systems, where autonomous agents interact strategically under uncertainty. As modern infrastructure becomes increasingly autonomous and data-driven, learning-enabled protocols are poised to play a central role in shaping the next generation of distributed systems.

1.4 Dissertation Overview

We end the introduction with an overview of the remaining dissertation.

- Chapter 2 provides background on normal-form games, equilibrium concepts, online learning, and classical algorithms. This foundational material supports the theoretical developments in the subsequent chapters.
- Chapter 3 introduces algorithms that minimize swap regret under bandit feedback and establish high-probability bounds. We further show that such guarantees lead to convergence toward the set of correlated equilibria.
- Chapter 4 presents an adaptive swap-regret-minimizing algorithm achieving a faster convergence rate by incorporating optimistic learning techniques.
- Chapter 5 applies one of the proposed algorithms to the practical setting of TCP congestion control. We implement the algorithm through the Linux Kernel 5.13.12 based on the congestion control plane [80] and evaluate its performance through extensive trace-driven emulation experiments.
- Chapter 6 concludes the dissertation and outlines possible directions for future research.
- Detailed proofs and technical lemmas for the theoretical results are provided in Appendix A.

Chapter 2

Background Materials

In this chapter, we explore the connections between game theory and online learning, and present key theorems that support our theoretical contributions. Unless otherwise stated, the full proofs of the theorems and lemmas discussed here are provided in Appendix A.1.

2.1 Game Theory

2.1.1 Normal-Form Games

A normal-form game is defined by a tuple $(N, \{\mathcal{A}_n\}_{n\in[N]}, \{u_n\}_{n\in[N]})$, where N is the number of players. Each player $n \in [N] := \{1, \dots, N\}$ has an action set $\mathcal{A}_n$ of finite size $A_n := |\mathcal{A}_n|$, and a reward function $u_n : \mathcal{A} \to \mathbb{R}$, where $\mathcal{A} = \bigotimes_{n=1}^{N} \mathcal{A}_n$ is the set of joint actions of all players. The reward function means that, after each player n chooses an action $a_n \in \mathcal{A}_n$, the resulting joint action determines payoffs for each player.

A player may also sample actions according to a probability distribution $p_n \in \Delta(\mathcal{A}_n)$ over their action set $\mathcal{A}_n$; we refer to this approach as a *mixed strategy*. Under this strategy, the player samples an action $a \sim p_n$ and plays the selected action.

When the normal-form game is repeated over T rounds (i.e., $t = 1, \dots, T$), we use superscripts to denote time-dependent quantities, e.g., $a_n^t \in \mathcal{A}_n$ is the action chosen by player n in round t, and $u_n^t : \mathcal{A} \to \mathbb{R}$ is the corresponding reward function for that round.

2.1.2 Equilibrium Definition

A Nash equilibrium (NE) is a joint strategy profile in which no player can benefit by unilaterally deviating. A formal definition is as follows.

Definition 2.1 (Nash Equilibrium)**.** Let $p_n \in \Delta(\mathcal{A}_n)$ be a mixed strategy for player n, and let $\boldsymbol{p} = (p_1, \ldots, p_n)$ be a joint mixed strategy profile. The expected rewards of player n are given by:

$$u_n(\boldsymbol{p}) = \mathbf{E}_{a \sim \boldsymbol{p}}[u_n(a)] = \sum_{a \in \mathcal{A}} \left(\prod_{j=1}^{n} p_j(a_j) \right) u_n(a).$$

A strategy profile $\boldsymbol{p}^\star = (p_1^\star, \ldots, p_N^\star)$ is called a *Nash equilibrium (NE)* if no player can improve their expected rewards by unilaterally deviating. That is, for all $n \in N$ and all $p_n \in \Delta(\mathcal{A}_n)$,

$$u_n(p_n^\star; p_{-n}^\star) \geq u_n(p_n; p_{-n}^\star),$$

where $(p_n^\star; p_{-n}^\star)$ is an abbreviation of $(p_1^\star, \ldots p_n^\star, \ldots, p_N^\star)$ to highlight the mixed strategy of player n against other players.

A more general concept is *correlated equilibrium (CE)*, which describes a joint distribution of action profiles, such that if there is a mediator (or a central controller) that samples an action profile from this joint distribution and secretly signals to the corresponding players, no players will deviate from the recommendation unilaterally. We adopt the definition of correlated equilibrium from Definition 7.3.1 of [32] as follows:

Definition 2.2 (Correlated Equilibrium)**.** A joint distribution $\mathbf{P} \in \Delta(\mathcal{A})$ is a *correlated equilibrium* if, for all $n \in [N]$ and $a_n, a_n' \in \mathcal{A}_n$, we have

$$\mathbf{E}_{A \sim \mathbf{P}} \left[u_n(a_n'; a_{-n}) \mid a_n \right] \leq \mathbf{E}_{A \sim \mathbf{P}} \left[u_n(A) \mid a_n \right],$$

where $(a_n'; a_{-n})$ is an abbreviation of $(a_1, \ldots, a_n', \ldots, a_N)$ to highlight the action of player n is a_n'.

Intuitively, if a mediator privately recommends action a_n to player n, and the other players follow their recommendations, then player n has no incentive to unilaterally deviate from a_n to any alternative action a_n'.

Next, we introduce the ϵ-correlated equilibrium, a generalization of the standard correlated equilibrium that allows for an additional cost. We adopt the alternative definition from Theorem 7.3.2 of [32], which removes the conditioning in the original definition at the cost of quantifying over all swapping functions. This formulation is particularly useful, as it naturally connects to the notion of swap regret introduced in Sec. 2.2.3.

Definition 2.3 (ϵ-Correlated Equilibrium)**.** A joint distribution $\mathbf{P} \in \Delta(\mathcal{A})$ is an ϵ-*correlated equilibrium* if, for every player $n \in [N]$, and any function $F_n : \mathcal{A}_n \to \mathcal{A}_n$, such that

$$\begin{aligned} &\mathbf{E}_{A\sim\mathbf{P}}\left[u_n(F_n(a_n); a_{-n}) - u_n(a_n; a_{-n})\right] \\ &= \sum_{A\in\mathcal{A}} \mathbf{P}((a_n; a_{-n}))\left(u_n(F_n(a_n); a_{-n}) - u_n(a_n; a_{-n})\right) \leq \epsilon. \end{aligned} \tag{2.1}$$

where $(a_n; a_{-n})$ is an abbreviation of $A := (a_1, \ldots, a_n, \ldots, a_N)$ to highlight the action of player n is a_n.

When $\epsilon = 0$, we obtain the definition of CE equivalent to Definition 2.2; a proof of their equivalence can be found in the proof of Theorem 7.3.2 in [32].

The main difference between a NE and a CE lies in the structure of the underlying probability distribution: a CE is defined over a joint distribution of players' actions, while a NE requires the distribution to factor as a product of independent strategies. As a result, every NE is a CE, but not every CE is an NE, making CE a strict generalization of NE by allowing strategic correlation among players.

When all the functions F_n considered are only constant functions, we have a more general equilibrium that is the coarse correlated equilibrium (CCE).

Definition 2.4 (ϵ-Coarse Correlated Equilibrium)**.** A joint distribution $\mathbf{P} \in \Delta(\mathcal{A})$ is called an ϵ-*coarse correlated equilibrium* (CCE) if, for every player $n \in [N]$, every action $a'_n \in A_n$, and $\epsilon > 0$, it holds that

$$\mathbf{E}_{a\sim\mathbf{P}}[u_n(a'_n, a_{-n}) - u_n(a)] \leq \epsilon. \tag{2.2}$$

When $\epsilon = 0$, we obtain the definition of CCE. Intuitively, the key difference between a CE and a CCE lies in what information players receive and when they commit to their actions. In a CE, a mediator samples an action profile from a joint distribution and privately recommends each player their part; players then decide whether to follow their recommendation, and the equilibrium condition ensures

that no player has an incentive to deviate given their private signal. In contrast, in a CCE, the mediator publicly samples and announces the entire action profile before players commit to any action; each player must decide whether to follow the suggestion or unilaterally switch to a fixed strategy, without conditioning on any private recommendation. As a result, CE imposes stricter incentive constraints than CCE, making the set of CEs a subset of the set of CCEs.

There is a deep connection between equilibrium computation and online learning. Broadly speaking, if each agent in a repeated game independently follows an online learning algorithm, the joint learning dynamics can converge to an equilibrium. Motivated by this relationship, we first review key concepts in online learning and then highlight its intrinsic connections to equilibrium computation in game theory.

2.2 Online Learning

Online learning provides a powerful abstraction for sequential decision-making under uncertainty. Consider the following model [108]: A learner has an action space $\Omega \subseteq \mathbb{R}^d$ for some $d > 0$, and in each round $t = 1, \ldots, T$,

1. The learner decides an action $p^t \in \Omega$,
2. The environment determines a loss/reward function $f^t : \Omega \to \mathbb{R}$,
3. The learner suffers loss $f^t(p^t)$ and observe some information about f^t.

The loss functions f^t may be chosen adversarially, either fixed in advance or adapted based on the learner's past actions.

Such a model captures various online learning problems, including the expert problem and the multi-armed bandit problem, which represent two feedback models, i.e., full-information feedback and bandit feedback.

2.2.1 The Expert Problem

The expert problem is a fundamental setting in online learning that models decision-making under uncertainty with access to a set of advisors or "experts". For example, in portfolio management, each expert can represent a distinct investment strategy, and the learner allocates capital across strategies based on past performance.

In news recommendation systems, experts correspond to different recommendation algorithms, and the learner adaptively blends their outputs to optimize user engagement. More broadly, the expert framework applies to any setting where multiple heuristics, models, or strategies are available, and the goal is to combine them adaptively in the presence of feedback.

Therefore, for an expert problem, given a set of experts $[d] := \{1, \ldots, d\}$, the learner needs to decide how to aggregate the advice of the d experts. The action space is the probability simplex over d experts, i.e., $\Omega = \Delta^{d-1} := \{p \in \mathbb{R}^d_{\geq 0} \mid \sum_{a \in [d]} p_a = 1\}$. Thus, in each round $t = 1, \ldots, T$:

- The learner needs to decide a vector $p^t \in \Omega$.
- The environment determines a loss function $f^t(p) = \langle p, y^t \rangle$, where $y^t \in [0,1]^d$ is the loss vector for experts.
- The learner receives a loss $f^t(p^t)$ and feedback y^t.

Here we note that a reward-based formulation can be equivalently converted to this loss-based setting, e.g., $y^t = \mathbf{1} - u^t$ for any reward $u^t \in [0,1]^d$, and vice versa. Therefore, for ease of presentation, we will refer to "loss" and "reward" using the same framework, converting between them as appropriate depending on context.

The expert problem is under the *full-information* feedback, meaning the learner observes the entire loss vector y^t after each round, even for experts whose advice was not followed.

2.2.2 The Multi-Armed Bandit Problem

The multi-armed bandit problem is a fundamental model in online learning that captures the trade-off between exploration and exploitation under limited feedback. The name "multi-armed bandit" comes from a metaphor involving a gambler at a casino facing multiple slot machines, each with different random payouts. Slot machines are colloquially called "one-armed bandits" because they have a single lever (the "arm") and can "steal" your money (like a bandit) if you keep playing unluckily.

Multi-armed bandits arise naturally in many real-world scenarios. For example, in online advertising, each arm represents a different ad, and the goal is to display ads that maximize click-through rates based on partial feedback. In clinical trials,

different treatments correspond to arms, and the challenge is to assign patients to effective treatments while learning their efficacy. Other applications include A/B testing and network routing under uncertainty. The simplicity and generality of the MAB model make it a foundational tool for sequential decision-making under uncertainty.

Formally, with notations similar to the expert problem, we can describe a multi-armed bandit problem with d arms as follows. In each round $t = 1, \ldots, T$:

1. The learner decides $p^t \in \Delta^{d-1}$ and selects one of the arms $a^t \sim p^t$.
2. The environment determines the loss function $f^t(a') := \sum_{a \in [d]} \mathbf{1}[a' = a] y_a^t$, where $y_a^t \in [0, 1]$.
3. The learner suffers the loss $f^t(a^t)$ but only observes the loss for the played arm $y_{a^t}^t$.

Unlike in the expert setting, the learner only observes the loss/reward associated with the chosen arm, i.e., only one entry of y^t, not the losses or rewards of the others, which defines the bandit-feedback model.

These two feedback models, *full-information* and *bandit* feedback, are the most fundamental and classical paradigms in online learning. The full-information model allows the learner to make more informed updates by accessing the complete loss vector, which typically leads to faster convergence and tighter regret bounds. In contrast, the bandit feedback model presents a more challenging scenario due to its limited observability, requiring the learner to balance exploration and exploitation to estimate losses and minimize regret. Many extensions and variations of online learning, such as contextual bandits, partial monitoring, and feedback graphs, can be viewed as generalizations or interpolations between these two extremes.

2.2.3 Regret

In online learning, we often use *regret* to measure the performance of a learning algorithm. The most basic regret notion is *external regret*, which compares the cumulative performance of a set of competitors that always play a fixed action over time.

Definition 2.5 (External Regret)**.** Given a sequence of loss vectors $y^1, \ldots, y^T \geq 0$, the regret between the actions $a^1, \ldots, a^T$ output by the learning algorithm and a

competitor always playing a fixed action $a' \in [d]$ is defined as follows:

$$\begin{aligned} R(T, a') &:= \sum_{t=1}^{T} \left(y_{a^t}^t - y_{a'}^t \right) \\ &= \sum_{t=1}^{T} \sum_{a \in [d]} \mathbf{1}[a^t = a] y_a^t - \sum_{t=1}^{T} y_{a'}^t. \end{aligned}$$

Furthermore, *external regret* is defined by

$$R^{\text{ext}}(T) := \max_{a' \in [d]} R(T, a').$$

External regret is the most basic regret notion, which only compares the learning algorithm to d competitors when there are d actions. Analyzing the external regret is not only practically meaningful when there is always one best arm, but also paves the way for more complicated regret notions, e.g., *swap regret*.

Swap regret compares the learning algorithm with competitors whose actions are defined by a set of swap functions $\mathcal{F} := \{F : [d] \to [d]\}$ that map each action to another (possibly the same) in $[d]$. The formal definition of swap regret is as follows.

Definition 2.6 (Swap Regret)**.** Given a sequence of loss vectors $y^1, \ldots, y^T \geq 0$, the regret between the actions $a^1, \ldots, a^T$ output by the learning algorithm and a competitor playing actions $F(a^1), \ldots, F(a^T) \in [d]$, where $F \in \mathcal{F}$, is defined as follows:

$$\begin{aligned} R(T, F) &:= \sum_{t=1}^{T} r_{a^t, F(a^t)} = \sum_{t=1}^{T} \left(y_{a^t}^t - y_{F(a^t)}^t \right) \\ &= \sum_{t=1}^{T} \sum_{a \in [d]} \mathbf{1}[a^t = a] y_a^t - \sum_{t=1}^{T} \sum_{a \in [d]} \mathbf{1}[a^t = a] y_{F(a)}^t. \end{aligned}$$

Furthermore, *swap regret* is defined by

$$R^{\text{swa}}(T) := \max_{F \in \mathcal{F}} R(T, F).$$

Since there are d^d possible swap functions over d actions, i.e., $|\mathcal{F}| = d^d$, swap regret compares the learning algorithm against a much larger set of competitors. As a result, a swap-regret-minimizing algorithm (i.e., one whose swap regret grows sublinearly with T) is robust to a broader class of loss sequences than algorithms

that only minimize external regret.

2.3 Connections Between Online Learning and Game Theory

For simplicity, we only consider normal-form games. There are many convergence types, including ergodic convergence and last-iterate convergence. In this dissertation, we focus on ergodic convergence, a fundamental concept that reveals interesting and important connections between online learning and game theory.

2.3.1 Ergodic Convergence to Nash Equilibrium

In a repeated two-player zero-sum game, the following theorem guarantees that if both players self-play an *external-regret-minimizing* algorithm, then the learning dynamics, i.e., the sequence of actions played over time, converge ergodically to a set of Nash equilibria as defined in Definition 2.1.

Theorem 2.1. Consider a two-player zero-sum game with payoff matrix $u \in \mathbb{R}^{A_1 \times A_2}$. Suppose that each player plays for T rounds and uses an algorithm that minimizes external regret, i.e., for player 1:

$$\max_{a \in \mathcal{A}_1} \frac{1}{T} \sum_{t=1}^{T} \left[u(a, a_2^t) - u(a_1^t, a_2^t) \right] \to 0 \quad \text{as } T \to \infty,$$

and similarly for player 2:

$$\max_{b \in \mathcal{A}_2} \frac{1}{T} \sum_{t=1}^{T} \left[-u(a_1^t, b) + u(a_1^t, a_2^t) \right] \to 0 \quad \text{as } T \to \infty.$$

Then, the empirical mixed strategies,

$$\hat{p}^T(a) := \frac{1}{T} \sum_{t=1}^{T} \mathbf{1}[a_1^t = a], \quad \hat{q}^T(a) := \frac{1}{T} \sum_{t=1}^{T} \mathbf{1}[a_2^t = a],$$

converge to the set of Nash equilibria; that is, every limit point $(\bar{p}, \bar{q})$ of the sequence $(\hat{p}^T, \hat{q}^T)$ as $T \to \infty$ is a Nash equilibrium.

2.3.2 Ergodic Convergence to Correlated Equilibrium

With the notations we introduced in Sec. 2.1, we further introduce the empirical joint distribution of actions as follows:

Definition 2.7 (Empirical Joint Distribution of Actions)**.** We define $\hat{P}^T$ as the empirical joint distribution over actions, where for any joint action $A \in \mathcal{A} := \bigotimes_{n=1}^{N} \mathcal{A}_n$, the empirical probability of A being played is

$$\hat{P}^T(A) := \frac{1}{T} \sum_{t=1}^{T} \mathbf{1}[A^t = A].$$

In a repeated multi-player normal-form game, the following theorem guarantees that, if every player self-plays a *swap-regret-minimizing* algorithm, then the empirical joint distribution $\hat{P}^T$ converges to a set of correlated equilibria defined in Definition 2.3.

Theorem 2.2. Let $\{A^t\}_{t=1}^{T}$ be the sequence of joint actions in an N-player game, where $A^t = (a_1^t, \ldots, a_N^t) \in \mathcal{A} := \bigotimes_{n=1}^{N} \mathcal{A}_n$. Suppose each player n follows a learning algorithm that guarantees

$$\max_{F_n : \mathcal{A}_n \to \mathcal{A}_n} \frac{1}{T} \sum_{t=1}^{T} \left[u_n(F_n(a_n^t), a_{-n}^t) - u_n(a_n^t, a_{-n}^t) \right] \leq \epsilon_T.$$

Then, $\hat{P}^T$ converges to the set of ϵ_T-correlated equilibria. If $\epsilon_T \to 0$ as $T \to \infty$, then $\hat{P}^T$ converges to the set of correlated equilibria as $T \to \infty$.

2.4 Classic Families of Learning Algorithms

In this section, we review two classic families of learning algorithms [87] — *online mirror descent* and *(optimistic) follow-the-regularized-leader*. Both algorithmic frameworks share common assumptions, including that each loss function f^t is convex and that the learner has access to strongly convex regularizers ψ (or possibly ψ^t). Additionally, certain concepts, such as Bregman divergence, are fundamental to both frameworks. We therefore introduce these preliminaries before presenting the details of both frameworks.

2.4.1 Preliminaries for Convex Analysis

Convex Set and Functions. We begin by briefly reviewing the notion of a convex function. A precise definition of a convex function first requires the concept of a convex set.

Definition 2.8 (Convex Set)**.** A set is said to be *convex* if, for any two points in the set, the entire line segment connecting them lies entirely within the set.

Intuitively, a convex set contains all points "between" any two of its elements, i.e., there are no "dents" or "holes" in the set.

Let $\text{dom}(\psi)$ denote the *effective domain* of ψ— the set of all input values p for which $\psi(p) < +\infty$. With this notion in place, we define convex functions as follows:

Definition 2.9 (Convex Function)**.** A function $\psi : \text{dom}(\psi) \to \mathbb{R}$ is said to be *convex* if its domain $\text{dom}(\psi)$ is convex, and for all $p, q \in \text{dom}(\psi)$ and all $\lambda \in [0, 1]$, the following inequality holds:

$$\psi(\lambda p + (1 - \lambda)q) \leq \lambda\psi(p) + (1 - \lambda)\psi(q).$$

The function is said to be *strictly convex* if the inequality is strict for all $p \neq q$.

Intuitively, a convex function lies below the straight line (chord) connecting any two points on its graph. In the strictly convex case, the graph of the function lies strictly below the chord, indicating that the function curves downward with no flat regions. Strict convexity also implies that a minimizer of the function, if it exists, is unique.

Interior and Relative Interior. Since a convex function may not be differentiable on the boundary of its domain, we usually study the concept of the interior or relative interior of its domain, as defined below.

Definition 2.10 (Interior)**.** For a set $\Omega \subseteq \mathbb{R}^n$, a point $p \in \Omega$ is called an *interior point* of Ω if there exists an open ball $B(p, \varepsilon) = \{q \in \mathbb{R}^n : \|q - p\| < \varepsilon\}$ such that $B(p, \varepsilon) \subseteq \Omega$. The set of all such points is called the *interior* of Ω, denoted $\text{int}(\Omega)$.

Intuitively, points in $\text{int}(\Omega)$ lie strictly inside Ω with some "breathing room", away from the boundary. However, when Ω lies in a lower-dimensional affine

subspace of $\mathbb{R}^n$, the interior in the ambient topology may be empty. A typical example is the simplex over d actions in online learning problems: $\Delta^{d-1} := \left\{p \in \mathbb{R}^d_{\geq 0} : \sum_{a\in[d]} p(a) = 1\right\}$, which has an empty interior in $\mathbb{R}^d$. In this case, we use the notion of *relative interior*:

Definition 2.11 (Relative Interior)**.** A point $p \in \Omega \subseteq \mathbb{R}^n$ is in the relative interior, denoted relint(Ω), if there exists an open ball in the affine hull of Ω that is contained in Ω. Equivalently,

$$\text{relint}(\Omega) = \{p \in \Omega : \exists \varepsilon > 0,\ B_{\text{aff}(\Omega)}(p, \varepsilon) \subseteq \Omega\},$$

where $\text{aff}(\Omega) = \left\{\sum_{i=1}^k \lambda_i q_i : k \geq 1, q_i \in \Omega, \sum_{i=1}^k \lambda_i = 1\right\}$ is the smallest affine subspace containing Ω, and $B_{\text{aff}(\Omega)}(p, \varepsilon) = \{q \in \text{aff}(\Omega) : \|q - p\| < \varepsilon\}$ is the open ball defined on aff(Ω).

For example, a simplex in $\mathbb{R}^n$ has empty interior in $\mathbb{R}^n$, but its relative interior is nonempty and consists of all convex combinations with strictly positive coefficients, denoted by $\text{relint}(\Delta^{d-1}) := \left\{p \in \mathbb{R}^d_{>0} : \sum_{a\in[d]} p(a) = 1\right\}$.

Suppose the convex function ψ is also differentiable in the interior of its domain, i.e., int(dom(ψ)). We have the *first-order optimality condition* for such a minimizer $p^\star$ of the function ψ:

$$\langle \nabla\psi\left(p^\star\right), q - p^\star\rangle \geq 0, \forall q \in \text{dom}(\psi). \tag{2.3}$$

Bregman Divergence. When ψ is strictly convex, we can define a notion of distance that reflects the geometry of the function by incorporating its curvature. This leads to the following definition of the Bregman divergence, which generalizes the concept of distance beyond the Euclidean setting.

Definition 2.12 (Bregman Divergence)**.** Let $\psi : \text{dom}(\psi) \to \mathbb{R}$ be strictly convex and differentiable on relint(dom$(\psi)) \neq \{\}$. The Bregman divergence with regard to ψ is denoted by $B_\psi : \text{dom}(\psi) \times \text{relint}(\text{dom}(\psi)) \to \mathbb{R}$ defined as

$$B_\psi(p; q) = \psi(p) - \psi(q) - \langle \nabla\psi(q), p - q\rangle .$$

The following lemma, due to [22], characterizes a key property of the Bregman divergence and will be instrumental in our algorithmic analysis. The proof is omitted, as it follows directly from the definition of Bregman divergence.

Lemma 2.3. For any three points $p, q \in \text{relint}(\text{dom}(\psi))$ and $u \in \text{dom}(\psi)$, we have $B_\psi(u; p) + B_\psi(p; q) - B_\psi(u; q) = \langle \nabla\psi(q) - \nabla\psi(p), u - p\rangle$.

Norm. In convex analysis, we will often use the *norm* of a vector to measure the distance from the origin in Euclidean space. The formal definition of norm is given as follows.

Definition 2.13 (Norm)**.** A norm on a vector space $\mathbb{R}^d$ is a function $\|\cdot\| : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$ that satisfies the following three properties for all $p, q \in \mathbb{R}^d$ and all scalars $c \in \mathbb{R}$.

1. Non-negativity: $\|p\| \geq 0$, and the equality holds when $p = \mathbf{0}$.
2. Positive homogeneity: $\|cp\| = |c|\|p\|$.
3. Triangle inequality: $\|p + q\| \leq \|p\| + \|q\|$.

A closely related concept is *dual norm*, defined as follows.

Definition 2.14 (Dual Norms)**.** The dual norm $\|\cdot\|_\star$ of a norm $\|\cdot\|$ is defined as $\|\theta\|_\star = \max_{p:\|p\|\leq 1} \langle \theta, p\rangle$.

We now present two examples of norms that are widely used in convex analysis. The first is the L_2 norm.

Example 2.1 (L_2 Norm)**.** The L_2 norm of a vector $p \in \mathbb{R}^d$ is defined as

$$\|p\|_2 := \left(\sum_{a\in[d]} p_a^2\right)^{1/2}.$$

The dual norm of the L_2 norm is itself an L_2 norm, i.e., $\|p\|_2$.

The second example is the *quadratic norm*, which depends on a positive definite matrix.

Example 2.2 (Quadratic Norm)**.** Given a positive definite matrix $\mathbf{A} \in \mathbb{R}^{d\times d}$, the quadratic norm of a vector $p \in \mathbb{R}^d$ with respect to $\mathbf{A}$ is defined as

$$\|p\|_{\mathbf{A}} := \sqrt{p^\top \mathbf{A} p}.$$

The dual norm of a quadratic norm is also a quadratic norm, but defined with respect to the inverse matrix: $\|p\|_{\mathbf{A}^{-1}}$.

This notion naturally generalizes to *local norms* when the positive definite matrix varies with the point. Indeed, by Taylor's theorem, the Bregman divergence $B_\psi(p;q)$ induced by a twice-differentiable function ψ can be locally approximated by a quadratic form:

$$B_\psi(p;q) := \psi(p) - \psi(q) - \langle \nabla\psi(q), p-q\rangle = \frac{1}{2}\|p-q\|^2_{\nabla^2\psi(\tilde{z})},$$

where the *local norm* induced by ψ is defined by the Hessian of ψ at some point $\tilde{z}$ on the line segment between p and q:

$$\|v\|_{\nabla^2\psi(\tilde{z})} := \sqrt{v^\top \nabla^2\psi(\tilde{z})v}.$$

In this way, the quadratic norm with a fixed matrix **A** is a special, global case of a local norm where the matrix is constant across the domain.

Self-Concordant functions. A concept closely tied to local norms is that of *self-concordant functions*, a special class of convex functions whose curvature changes are smoothly controlled. Self-concordant functions were introduced by Nesterov and Nemirovski in the context of interior-point methods [82, 83, 84]. The key idea is that the third-order directional derivative in any direction is bounded by the cube of the local norm, ensuring well-behaved curvature for optimization.

Definition 2.15 (Self-Concordant Function)**.** A convex and three-times differentiable function $\psi : \mathrm{dom}(\psi) \to \mathbb{R}$ is said to be *self-concordant* on a nonempty, open, convex set $\mathrm{dom}(\psi) \subseteq \mathbb{R}^d$ if for all $p \in \mathrm{dom}(\psi)$ and all directions $h \in \mathbb{R}^d$, it satisfies:

$$\left|D^3\psi(p)[h,h,h]\right| \le 2\left(h^\top\nabla^2\psi(p)h\right)^{3/2} = 2\|h\|^3_{\nabla^2\psi(p)},$$

where $D^3\psi(p)[h,h,h]$ denotes the third directional derivative of ψ at p in direction h, and $\|h\|_{\nabla^2\psi(p)} := \sqrt{h^\top\nabla^2\psi(p)h}$ is the local norm induced by ψ at p.

Furthermore, we call ψ a θ-self-concordant barrier, if for all $p \in \mathrm{dom}(\psi)$ and $h \in \mathbb{R}^d$,

$$|\langle \nabla\psi(p), h\rangle|^2 \le \theta\|h\|^2_{\nabla^2\psi(p)}. \tag{2.4}$$

We provide an example of self-concordant functions as follows.

Example 2.3 (Log-Barrier)**.** A canonical example of a self-concordant function is

the log-barrier function defined on the positive orthant $\mathbb{R}^d_{>0}$ by

$$\psi(p) = -\sum_{a\in[d]} \ln p(a).$$

This function is convex and diverges to $+\infty$ as any $p(a) \to 0^+$, making it a barrier function. Its Hessian is diagonal with entries $\nabla^2\psi(p)_{aa} = \frac{1}{p(a)^2}$, and its third-order directional derivative satisfies

$$|D^3\psi(p)[h,h,h]| = 2\left(h^\top \nabla^2\psi(p)h\right)^{3/2},$$

So it meets the self-concordance condition with equality. In addition, $\psi(p)$ is a d-self-concordant barrier for $p \in \mathbb{R}^d_{>0}$ according to (2.4), because

$$|\langle \nabla\psi(p), h\rangle|^2 = \left|\sum_{i=1}^d \left(-\frac{1}{p_i}\right) h_i\right|^2 = \left|\sum_{i=1}^d \frac{h_i}{p_i}\right|^2 \le \left(\sum_{i=1}^d \frac{h_i^2}{p_i^2}\right)\cdot d = d\cdot \|h\|^2_{\nabla^2\psi(p)},$$

where the last inequality follows from the Cauchy-Schwarz inequality.

Now, we present a lemma from (2.3.3) in Proposition 2.3.2 of [84] that bounds the difference in function values of a self-concordant barrier based on a certain notion of distance between two interior points.

Lemma 2.4. For any θ-self-concordant barrier ψ and $p, q \in \mathrm{int}(\mathrm{dom}(\psi))$, we have

$$\psi(p) - \psi(q) \le \theta \ln\left(\frac{1}{1-\pi(p;q)}\right),$$

where

$$\pi(p;q) = \inf\left\{s \ge 0 : q + s^{-1}(p-q) \in \mathrm{dom}(\psi)\right\}. \tag{2.5}$$

With the above tools, we are ready to introduce the two classic algorithms for online learning.

2.4.2 Online Mirror Descent

The core idea of Online Mirror Descent (OMD), as shown in Alg. 2.1, is to approximate the loss function at each round with a linear surrogate and then perform an update regularized by a Bregman divergence. Specifically, after observing the loss

y^t, OMD selects the next point by minimizing this linearized objective combined with a Bregman divergence:

$$p^{t+1} = \operatorname*{argmin}_{p\in\Omega} \ \langle p, y^t\rangle + \frac{1}{\eta} B_\psi(p; p^t). \tag{2.6}$$

Note that the update in (2.6) can be implemented in two steps. First, we solve the optimization in an unconstrained setting to obtain a minimizer $\tilde{p} \in \mathbb{R}^d$. Then, we project $\tilde{p}$ back onto the feasible set Ω:

$$\begin{aligned} \tilde{p}^{t+1} &= \operatorname*{argmin}_{p\in\mathrm{dom}(\psi)} \left\langle y^t, p\right\rangle + \frac{1}{\eta} B_\psi\left(p; p^t\right), \\ p^{t+1} &= \operatorname*{argmin}_{p\in\Omega} \ B_\psi\left(p; \tilde{p}^{t+1}\right). \end{aligned} \tag{2.7}$$

The benefit of this two-step implementation is that we can sometimes solve two easy optimization problems instead of one.

Algorithm 2.1 Online Mirror Descent

1: **Input:** $\psi, \eta_t > 0, \forall t \in [T]$
2: Set $p^1 = \operatorname*{argmin}_{p\in\Omega} \psi(p)$
3: **for** $t = 1, \ldots, T$ **do**
4: Output p^t
5: Observe a loss vector y^t
6: Set $p^{t+1} = \operatorname*{argmin}_{p\in\Omega} \ \langle p, y^t\rangle + \frac{1}{\eta} B_\psi(p; p^t)$
7: **end for**

OMD has the following theoretical guarantees using local norms (see Definition 2.2).

Lemma 2.5. Assume the solutions $\tilde{p}^{t+1}$ and p^{t+1} to (2.7) exist. Let $\tilde{z}^t$ lie on the line segment between p^t and $\tilde{p}^{t+1}$. Then, for any sequence of loss $y^1, \ldots, y^T \geq 0$, and any $p \in \Omega$, the OMD's output $p^1, \ldots, p^T$ satisfies:

$$\sum_{t=1}^{T} \left\langle p^t - p, y^t\right\rangle \leq \max_{1\leq t\leq T} \frac{B_\psi(p; p^t)}{\eta} + \frac{\eta}{2} \sum_{t=1}^{T} \left\|y^t\right\|^2_{(\nabla^2\psi(\tilde{z}^t))^{-1}}.$$

Local norms allow regret to be bounded in terms of geometry at the current iteration, yielding adaptive bounds that can be significantly tighter than worst-case global norms.

Next, we present two examples of OMD on the probability simplex Δ^{d-1} instantiated with the negative Shannon entropy, and negative Tsallis entropy, which serve as building blocks for the algorithm proposed in Chapter 3.

Example 2.4 (OMD-Shannon)**.** Consider OMD using the (negative) Shannon entropy as the regularizer defined by:

$$\psi(p) := \sum_{a\in[d]} p(a)\ln(p(a)).$$

Then, the two-step update rule becomes (2.7) for all $a \in [d]$ as follows:

$$\begin{aligned} \tilde{p}^{t+1}(a) &= p^t(a)\exp(-\eta y_a^t), \\ p^{t+1}(a) &= \frac{\tilde{p}^{t+1}(a)}{\sum_{a\in[d]}\tilde{p}^{t+1}(a)}, \end{aligned} \tag{2.8}$$

which is equivalent to the following one-step update rule for any $a \in [d]$:

$$p^{t+1}(a) = \frac{\exp\left(-\eta\sum_{s=1}^{t} y_a^s\right)}{\sum_{a\in[d]}\exp\left(-\eta\sum_{s=1}^{t} y_a^s\right)},$$

which recovers the famous Hedge algorithm for the expert problems, and the Exp3 algorithm [9] for the bandit problems. Now, we state the negative Shannon-entropy version of Lemma 2.5.

Lemma 2.6. With the condition in Lemma 2.5, for any $y^1, \ldots, y^T \geq 0$ and for any $p \in \Delta^{d-1}$, OMD with the negative Shannon entropy satisfies:

$$\sum_{t=1}^{T}\left\langle p^t - p, y^t\right\rangle \leq \frac{\ln d}{\eta} + \frac{\eta}{2}\sum_{t=1}^{T}\sum_{a\in[d]} p^t(a)(y_a^t)^2.$$

Proof. The negative Shannon entropy $\psi(p)$ is maximized when p concentrates on one action, and is minimized when p is uniform. Thus, $B_\psi(p, p') \leq \max_p \psi(p) - \min_p \psi(p) = \ln d$. On the other hand,

$$\|y^t\|^2_{(\nabla^2(\psi(\tilde{z}^t))^{-1})} = \sum_{a\in[d]} \tilde{z}^t(a)(y_a^t)^2 \leq \sum_{a\in[d]} p^t(a)(y_a^t)^2,$$

where the first equality follows from the definition of the local norm and the Hessian of the negative Shannon regularizer: $\nabla^2\psi(\tilde{z}^t) = \text{diag}(\frac{1}{\tilde{z}^t(1)}, \ldots, \frac{1}{\tilde{z}^t(d)})$. The inequality follows from the fact that $\tilde{z}^t$ lies on the line segment between p^t and $\tilde{p}^{t+1}$, and the first equation in the two-step update rule of (2.8) shows that $\tilde{p}^{t+1}(a) = p^t(a)e^{-\eta y_a^t} \leq p^t(a)$, implying that $\tilde{z}^t(a) \leq p^t(a)$ for all $a \in [d]$. □

Example 2.5 (OMD-Tsallis)**.** Consider OMD using the (negative) Tsallis entropy [2] as the regularizer defined by:

$$\psi(p) := \frac{1 - \sum_{a\in[d]} p(a)^\beta}{1-\beta},$$

where $\beta \in (0,1)$. The Tsallis entropy can be seen as a generalization of the Shannon entropy. As β approaches 1, the expression $\frac{1-\sum_{a\in[d]} p(a)^\beta}{\beta-1} \to -\sum_{a\in[d]} p(a)\ln(p(a))$. The two-step update rule from (2.7) then becomes the solutions to the following two equations for each $a \in [d]$:

$$\begin{aligned} \frac{1}{(\tilde{p}^{t+1}(a))^{1-\beta}} &= \frac{1}{(p^t(a))^{1-\beta}} + \frac{1-\beta}{\beta}\eta y^t, \\ \frac{1}{(p^{t+1}(a))^{1-\beta}} &= \frac{1}{(\tilde{p}^{t+1}(a))^{1-\beta}} + \lambda^t, \end{aligned} \tag{2.9}$$

where λ^t is a constant from the method of Lagrange multipliers that makes p^{t+1} a distribution, i.e., $\sum_{a\in[d]} p_a^{t+1} = 1$, which can be computed efficiently by a binary search or Newton methods. We now state the Tsallis-entropy version of Lemma 2.5 with regard to any fixed $p \in \Delta^{d-1}$.

Lemma 2.7. With the condition in Lemma 2.5, for $y^1, \ldots, y^T \geq 0$ and for any $p \in \Delta^{d-1}$, OMD with the negative Tsallis entropy satisfies:

$$\sum_{t=1}^T \langle p^t - p, y^t\rangle \leq \frac{(d)^{1-\beta}-1}{(1-\beta)\eta} + \frac{\eta}{2\beta}\sum_{t=1}^T \sum_{a\in[d]} \left(p^t(a)\right)^{2-\beta}(y_a^t)^2.$$

Proof. The negative Tsallis entropy $\psi(p)$ is maximized when p concentrates on one action, and is minimized when p is uniform. Thus, $B_\psi(p,p') \leq \max_p \psi(p) -$

$\min_p \psi(p) = \frac{d^{1-\beta}-1}{1-\beta}$. On the other hand,

$$\|y^t\|^2_{(\nabla^2(\psi(\tilde{z}^t))^{-1})} = \sum_{a\in[d]} \frac{1}{\beta}\left(\tilde{z}^t(a)\right)^{2-\beta}(y_a^t)^2 \le \sum_{a\in[d]} \frac{1}{\beta}\left(p^t(a)\right)^{2-\beta}(y_a^t)^2,$$

where the first equality follows from the definition of the local norm and the Hessian of the Tsallis regularizer: $\nabla^2\psi(\tilde{z}^t)^{-1}(a,a) = \frac{1}{\beta}\left(\tilde{z}^t(a)\right)^{2-\beta}$. The inequality follows from the fact that $\tilde{z}^t$ lies on the line segment between p^t and $\tilde{p}^{t+1}$, and the first equation in the two-step update rule of (2.9) ensures that $\tilde{p}^{t+1}(a) \le p^t(a)$, implying $\tilde{z}^t(a) \le p^t(a)$ and hence $(\tilde{z}^t(a))^{2-\beta} \le (p^t(a))^{2-\beta}$ for $\beta < 2$. Now, invoking Lemma 2.5 gives the desired results. □

2.4.3 (Optimistic) Follow-The-Regularized-Leader

In this section, we transition from loss-based to reward-based formulations to maintain consistency with Chapter 4. Unlike OMD, which relies only on the previous round's information, the *Follow-The-Regularized-Leader (FTRL)* algorithm selects actions based on the entire history of observed rewards and a (time-varying) regularizer ψ. Specifically, the decision at round t for linear reward is given by:

$$p^t \in \arg\max_{p\in\Omega} \eta \left\langle \sum_{s=1}^{t-1} u^s, p \right\rangle - \psi(p),$$

where $\eta > 0$ is a predefined learning rate.

Algorithm 2.2 Follow-The-Regularized-Leader

1: **Input:** $\psi, \eta_t > 0, \forall t \in [T]$
2: **for** $t = 1, \ldots, T$ **do**
3: Output $p^t \in \arg\max_{p\in\Omega} \eta \left\langle \sum_{s=1}^{t-1} u^s, p \right\rangle - \psi(p)$
4: Observe a reward vector u^t
5: **end for**

Next, we will discuss a special FTRL algorithm called *optimistic FTRL (OFTRL)*, as shown in Alg. 2.2. The main idea of OFTRL is to add vector m^t as a predictor of the current-round reward before observing it:

$$p^t \in \arg\max_{p\in\Omega} \eta \left\langle m^t + \sum_{s=1}^{t-1} u^s, p \right\rangle - \psi(p).$$

Algorithm 2.3 Optimistic Follow-The-Regularized-Leader

Input: $\psi, \eta_t > 0, \forall t \in [T]$
for $t = 1, \ldots, T$ **do**
  Output $p^t \in \arg\max_{p \in \Omega} \eta \left\langle m^t + \sum_{s=1}^{t-1} u^s, p \right\rangle - \psi(p)$
  Observe a reward vector u^t
end for

Thus, if we had perfect predictions, i.e., if $m^t = y^t$, we could select the optimal action for the current round. However, in practice, accurately predicting the actual loss is generally infeasible. Despite this, even when the prediction is inaccurate, the OFTRL algorithm retains its worst-case performance guarantee.

OFTRL is quite useful for accelerating convergence to equilibria in learning games. Although the well-known dependence of no-regret learning algorithms on T is $O(\sqrt{T})$ in adversarial environments, we can get a faster rate in a game setting where every player plays a no-regret algorithm with a self-concordant function (see Definition 2.15) as the regularizer. A self-concordant regularizer exhibits controlled curvature growth, meaning its second derivative changes smoothly. This property helps bound the variance between consecutive decisions, enabling more stable updates and faster convergence to equilibrium.

The following result is adapted from Corollary B.3 of [6], which demonstrates that OFTRL with a self-concordant regularizer satisfies the desired property. For completeness, we reproduce their proof in Appendix A.1.5.

Lemma 2.8. Assume ψ is a nondegenerate (i.e., $\nabla^2\psi(p)$ is positive definite) self-concordant function in $\text{int}(\Omega)$ satisfying that $\psi(p) \geq 0, \forall p \in \text{int}(\Omega)$ and $\nabla^2\psi(\tilde{p}) \preceq 2\nabla^2\psi(p)$ for any $p, \tilde{p} \in \text{int}(\Omega)$ with $\|p - \tilde{p}\|_{\nabla^2\psi(\tilde{p})} \leq \frac{1}{4}$. Let $g^t := \arg\max_{g \in \Omega} \eta \left\langle \sum_{s=1}^{t} u^s, g \right\rangle - \psi(g)$ be the auxiliary sequence. If $\eta \|u^t - m^t\|_{(\nabla^2\psi(p^t))^{-1}} \leq \frac{1}{8}$ and $\eta \|m^t\|_{(\nabla^2\psi(g^{t-1}))^{-1}} \leq \frac{1}{2}$ for all $t \in [T]$, then for any reward sequence $u^1 \ldots u^T \geq 0$ and any $p \in \text{int}(\Omega)$, we have

$$\sum_{t=1}^{T} \left\langle p - p^t, u^t \right\rangle \leq \frac{\psi(p)}{\eta} + 2\eta \sum_{t=1}^{T} \left\| u^t - m^t \right\|^2_{(\nabla^2\psi(p^t))^{-1}} - \frac{1}{16\eta} \sum_{t=1}^{T} \|p^t - p^{t-1}\|^2_{\nabla^2\psi(p^{t-1})}.$$

Next, we revisit the example from [6], in which OFTRL is instantiated with the log-barrier regularizer on the probability simplex.

Example 2.6 (OFTRL-Log-Barrier)**.** Define the log-barrier regularizer over the probability simplex Δ^{d-1} as follows:

$$\psi(p) := -\sum_{a\in[d]} \ln(p(a)). \tag{2.10}$$

Using this regularizer, and applying Lemma 2.8, the authors of [6] show that the OFTRL algorithm admits the following guarantee for any $p \in \text{relint}(\Delta^{d-1}) := \left\{p \in \mathbb{R}^d_{>0} : \sum_{a\in[d]} p(a) = 1\right\}$:

Lemma 2.9. Let $g^t := \arg\max_{g\in\Omega} \eta \left\langle \sum_{s=1}^t u^s, g\right\rangle - \psi(g)$ be the auxiliary sequence. Assume $\eta\|u^t - m^t\|_{(\nabla^2\psi(p^t))^{-1}} \le \frac{1}{8}$ and $\eta \|m^t\|_{(\nabla^2\psi(g^{t-1}))^{-1}} \le \frac{1}{2}$ for all $t \in [T]$. Then for any reward sequence $u^1 \dots u^t \ge 0$ and any $p \in \text{relint}(\Delta^{d-1})$, OFTRL with ψ defined in (2.10) satisfies

$$\sum_{t=1}^T \left\langle p - p^t, u^t\right\rangle \le \frac{\psi(p)}{\eta} + 2\eta \sum_{t=1}^T \left\|u^t - m^t\right\|^2_{(\nabla^2\psi(p^t))^{-1}} - \frac{1}{16\eta}\sum_{t=1}^T \|p^t - p^{t-1}\|^2_{\nabla^2\psi(p^{t-1})}.$$

Proof Sketch. Notice that Lemma 2.8 requires the decision set Ω to have a non-empty interior. However, the standard simplex over d actions has empty interior in the ambient space $\mathbb{R}^d$, since it lies on a $(d-1)$-dimensional hyperplane. To address this, we restrict our attention to a $(d-1)$-dimensional representation and convert our learning problem to be on the domain as $\Delta^\circ := \left\{p \in \mathbb{R}^{d-1}_{\ge 0} : \sum_{a\in[d-1]} p(a) \le 1\right\}$. For notational convenience, we define $p(d) := 1 - \sum_{a\in[d-1]} p(a)$. The log-barrier regularizer is then given by:

$$\tilde{\psi}(p) := -\sum_{a\in[d-1]} \ln(p(a)) - \ln\left(1 - \sum_{a\in[d-1]} p(a)\right).$$

To work in a $(d-1)$-dimensional space, we first transform the d-dimensional reward vector into a $(d-1)$-dimensional one. For each $a \in [d-1]$, define $\tilde{u}^t(a) := u^t(a) - u^t(d)$, and similarly, $\tilde{m}^t(a) := m^t(a) - m^t(d)$. This transformation preserves the

regret. Specifically, for any $p, p^t \in \text{int}(\Delta^\circ)$, we have

$$\sum_{a\in[d-1]} \tilde{u}^t(a)\left(p(a)-p^t(a)\right) = \sum_{a\in[d-1]} u^t(a)\left(p(a)-p^t(a)\right) - u^t(d)\sum_{a\in[d-1]}\left(p(a)-p^t(a)\right)$$
$$= \sum_{a\in[d]} u^t(a)\left(p(a)-p^t(a)\right),$$

where the last equality uses the notation $p(d) := 1 - \sum_{a\in[d-1]} p(a)$, and similarly for $p^t(d)$. Thus, the transformed reward vector yields the same regret expression as the original.

Now, what remains is to show that the assumptions of Lemma 2.8 are satisfied. Then, the lemma follows by invoking Lemma 2.8, and by showing that $\|\tilde{u}^t - \tilde{m}^t\|^2_{(\nabla^2\tilde{\psi}(p^t))^{-1}} \le \|u^t - m^t\|^2_{(\nabla^2\psi(p^t))^{-1}}$ and $\|p^t - p^{t-1}\|^2_{\nabla^2\tilde{\psi}(p^{t-1})} = \|p^t - p^{t-1}\|^2_{\nabla^2\psi(p^{t-1})}$. The detailed proof can be found in Appendix A.1.6 □

Chapter 3

High-Probability Upper Bounds for Swap Regret

3.1 Introduction

In this chapter, we study the setting of *unknown games with bandit feedback (referred to as unknown-game bandits)*, closely related to the black-box game framework introduced in [81]. A set of players $[N] = \{1, \ldots, N\}$, each with a possibly distinct action set $\mathcal{A}_n$, repeatedly play an unknown general-sum game over T rounds. The players have no knowledge of the game's structure and cannot observe the actions or rewards of others. In each round, player n selects an action $a_n^t \in \mathcal{A}_n$ and receives a reward u_n^t, which depends on the joint action profile of all players. Note that the only feedback available to each player is the reward associated with their own selected action in each round. As a result, each player faces a non-stochastic multi-armed bandit problem against other players' actions.

The goal of each player is twofold: to accumulate as much reward as possible, and to ensure that the empirical joint distribution of all players' actions converges to an *ϵ-correlated equilibrium (CE)* [10] within T rounds. The correlated equilibrium is a generalization of the well-known Nash equilibrium. Intuitively, an ϵ-CE is a distribution over joint actions such that no player can gain more than $\epsilon \geq 0$ in expected reward by unilaterally deviating from a recommended action, where the expectation is taken with respect to the joint distribution.

The motivation comes from many practical network optimization problems, such as wireless access control and end-to-end congestion control in computer net-

works. In the problem of wireless medium access control, a set of devices need to access a shared communication channel to send packets in each time slot without collisions. In the case of end-to-end congestion control, each host has no information about others and needs to choose a transmission rate or congestion window, hoping to maximize its throughput without congesting the network. unknown-game bandits can effectively encapsulate the dynamics due to competition and information constraints in these network optimization problems, offering an opportunity to develop innovative algorithms from a game-theoretic learning perspective.

As each player has limited knowledge about the environment and can only learn from the rewards of the played arm affected by others' actions in each round, the algorithm to address unknown-game bandits must be carefully designed to balance the tradeoff between exploration and exploitation. The performance of an algorithm is usually measured by *regret*. As we have introduced in Sec. 2.2.3, the most oft-used definition of regret in the bandit literature is called *external regret* [20, 69], which measures the performance loss of an algorithm against a set of competitors always playing a fixed action. However, minimizing the external regret is insufficient for unknown-game bandits, as it does not guarantee convergence to an ϵ-CE (see discussions in Sec. 2.3). Thus, we study a stronger regret notion called *swap regret* introduced by [12], comparing the performance of a learning algorithm against a broader set of competitors. The swap regret uses swap functions F that take the arms played by an algorithm as input and output an arm to be compared. Notably, external regret is a special case of swap regret when F always maps to a fixed action. Minimizing the swap regret not only ensures convergence to ϵ-CE [44, 20], but also provides robustness in more complex and interactive environments.

To ensure the empirical joint plays of all players converge to a set of correlated equilibria, one has to give a more meaningful and stronger bound on the instantaneous swap regret (i.e., Definition 2.6) for any sequence of actions and rewards. However, the previous studies on swap regret [12, 94, 58, 61] only bound pseudo-regret or conditionally expected swap regret. In addition, the analysis of regret bounds for single-player bandits may not be directly applicable to multi-player games, because the reward or loss vector at each round is not determined at the beginning of the round, but is realized only after all players have selected their actions. As such, we are motivated to design a learning algorithm that can minimize the instantaneous swap regret with high probability, with the regret analysis

accounting for the joint randomness across all players.

The main contributions of this chapter are as follows.

- First, we present an *online-mirror-descent (OMD)*-based algorithmic framework to address unknown-game bandits, which is based on the swap-regret-minimizing framework proposed by [12], and the main idea is to call A_n OMD algorithms with the Implicit eXploration (IX) technique [64, 85] as subroutines. Then, the probability of selecting an arm is obtained by the Markov steady-state distribution of the Markov process among A_n subroutines, and the reward is proportionally fed to the subroutines for updates.

- Then, based on the framework, we show two algorithms with theoretical guarantees, LCE-IX and OMD-LCE-IX, which are based on negative Shannon entropy and negative Tsallis entropy, respectively.

  Note that the existing concentration inequality for the IX technique cannot be simply applied to the analysis of swap regret. The main difficulties are twofold. First, while the swap regret can only be represented as the aggregate of external regrets of subroutine algorithms *in expectation*, this representation does not hold for the instantaneous swap regret. Second, the reward of each arm results from all players' actions, which is not determined at the beginning of each round as in the single-player bandit setting (see more discussions in Sec. 3.3).

  To address this problem, we prove a new concentration inequality between the IX loss estimator and the swapped loss based on a refined martingale analysis by treating the A_n subroutine algorithms as a whole.[1] Based on this concentration inequality, we show that with probability at least $1-\delta$ for $\delta \in (0,1)$, the instantaneous swap regrets of LCE-IX and OMD-LCE-IX are bounded by $O(A_n\sqrt{T\ln(A_n/\delta)})$ for each player $n \in [N]$.

- Furthermore, we prove that with high probability, OMD-LCE-IX can converge to ϵ-correlated equilibria for unknown general-sum games in a polynomial number of rounds if the algorithm is played by all players.

[1] We use loss in analysis for convenience without loss of generality, as a reward-based algorithm can be easily converted to a loss-based algorithm by the convention we mentioned at the end of Sec. 2.2.1.

- Moreover, we provide empirical validation through numerical experiments, demonstrating that the unknown-game bandit framework can be applied to practical scenarios such as wireless medium access control.

The remainder of this chapter is organized as follows. Sec. 3.2 reviews related work on swap-regret minimization. Sec. 3.3 formally defines the problem setting. The proposed algorithms are introduced in Sec. 3.4, followed by their theoretical analysis in Sec. 3.5 and experimental evaluation in Sec. 3.6. Finally, Sec. 3.7 concludes the chapter.

3.2 Related Works

Multi-player bandits: Multi-player bandits consider a group of players participating in decision making, and aim to improve learning efficiency through collaborations. The works about multi-player bandits are mainly focused on improving rewards by communication [16, 21, 65, 100], identifying the best arm to avoid collision [15, 74, 47, 96, 59], and voting for playing arms [34]. All the above bandit settings assume the arm set for each player is identical, and the reward for a player does not depend on the actions of other players, or just follows a simple collision model. Unknown-game bandits consider (possibly) varied arm sets for different players and more general competitions among players.

Table 3.1: Swap-regret bounds in the bandit settings

Upper bound, Computational cost, Regret notion
$O\left(\sqrt{TA_n^3}\right)$, poly-time, pseudo-regret [12]
$O\left(\sqrt{TA_n^2\ln(A_n)}\right)$, exp-time, pseudo-regret [94]
$O\left(\sqrt{TA_n^2}\right)$, poly-time, pseudo-regret [58]
$O\left(\sqrt{TA_n^2\ln(A_n/\delta)}\right)$, poly-time, conditionally expected regret [61]
$O\left(\sqrt{TA_n^2\ln(A_n/\delta)}\right)$, poly-time, instantaneous regret (Theorem 3.4)
$O\left(\sqrt{TA_n^2\ln(A_n/\delta))}\right)$, poly-time, instantaneous regret (Theorem 3.5)

Learning in games: The history of learning in games can be traced back to the fictitious play for the two-player zero-sum games [14, 91]. Nevertheless, such a fictitious play requires that the decisions of opponents can be observed, and thus

it cannot be applied to the unknown games where the players can only observe their own outcomes (or rewards). To address the challenges of unknown games, online learning has been introduced by many works for specific games such as potential games [28, 26, 11, 76], and mean-field games [77, 101, 106]. However, the above solutions for specific games depend on corresponding properties (e.g., potential functions for potential games), and thus cannot be easily extended to the general-sum games. Thus, we focus on the learning in the unknown general-sum games (i.e., black-box games [81]), which provides theoretical foundations for learning in general-sum Markov games [73].

Regarding the unknown general-sum games, there are mainly two lines of research depending on the observability of rewards. If the reward of an action can be observed regardless of whether it is played or not, we call it the *full-information feedback* [20], and if only the reward of a played action can be observed, then it is the *bandit feedback*. Recent years have witnessed steady progress in learning general-sum games under the full-information feedback [66, 88, 23, 31, 5, 39]. However, the results for the full-information feedback cannot be easily extended to the bandit-feedback model, as less information is observed in each round. The first work that addressed the unknown general-sum games with bandit feedback is [9], where an exponential-weighting-based technique is proposed to minimize the external regret. However, as shown in [20], only minimizing external regret cannot converge to correlated equilibria.

The authors of [12] generalized the notion of external regret to swap regret, and proposed a polynomial-time swap-regret-minimizing framework based on A_n external-regret-minimizing subalgorithms, where A_n is the number of arms. They proved that if the external regret of each subalgorithm can be represented by a concave function $r(T)$ (the dependency on A_n is ignored in $r(T)$), the swap pseudo-regret of their proposed algorithm is $A_n \cdot r(T)$. Since the minimax-optimal external regret bound is $r(T) = O(\sqrt{TA_n})$ [2], the analysis of [12] gives a regret bound of $O(A_n\sqrt{TA_n})$.

Later, this bound was improved by a new algorithm [94] to $O(A_n\sqrt{T \ln A_n})$ but with an exponential computation complexity. Furthermore, the author of [58] improved the upper bound for the swap regret to $A_n \cdot r(T/A_n)$ with a polynomial-time algorithm by adding another layer of randomness to the original framework [12], where in each round only one subroutine is selected according to the calculated Markov steady distribution. The selected subroutine selects an arm, and the re-

ward will be *entirely* fed to this subroutine algorithm for updates. The modified framework gives a regret of $O(\sqrt{TA_n^2})$ for the minimax-optimal external-regret-minimizing subalgorithms. It was also proved by [58] that the lower bound for swap regret is $\Omega\left(\sqrt{TA_n \ln A_n}\right)$, which is tight in the full-information feedback, and it remains an open problem whether the lower bound is tight in the bandit settings. However, all of the above swap regrets are in the form of pseudo-regret, i.e., $\max_F \mathbf{E}[R(T,F)]$, with the notations defined in Sec. 2.2.3.

The author of [61] proved a high-probability bound of $O(A_n\sqrt{T\ln(A_n/\delta)})$ for the conditionally expected swap regret, i.e., $\max_F \sum_{t=1}^T \mathbf{E}_{t-1}^T\left[r_{a^t,F(a^t)}\right]$, where $r_{a^t,F(a^t)}$ is the instantaneous regret in round t with the swap function F. This bound enables control of the pseudo-regret through tail integration.

In this chapter, we present a swap-regret-minimizing algorithmic framework based on online mirror descent. Using the negative entropy and Tsallis entropy as the regularizer, our algorithm achieves a swap regret bound of $O(A_n\sqrt{T\ln(A_n/\delta)})$, with probability at least $1-\delta$. Table 3.1 summarizes the existing literature on swap-regret bounds in bandit settings.

3.3 Problem Formulation

3.3.1 Unknown General-Sum Games with Bandit Feedback

We consider a general-sum game with multiple players making decisions over a number of rounds. The game is unknown because each player has no prior knowledge about the environment, such as the number of players, the reward of each action, and the actions of other players. Each player needs to make their own decision based solely on the feedback they received for their past played actions (i.e., bandit feedback). Thus, each player is playing a multi-armed bandit problem against other players, as the reward for each arm (i.e. action) is determined by the actions of all players.

A simple example of such an unknown game with two players and two arms for each player is shown in Fig. 3.1, where in the current round, player 1 plays arm a and only observes a normalized reward of 0.8, and player 2 plays arm c and only observes a normalized reward of 0.2. Both players have no information about the arm played by the other player, nor the rewards of the arms that are not played.

Formally, let $[N] := \{1,\ldots,N\}$ be the set of all players and each player $n \in [N]$

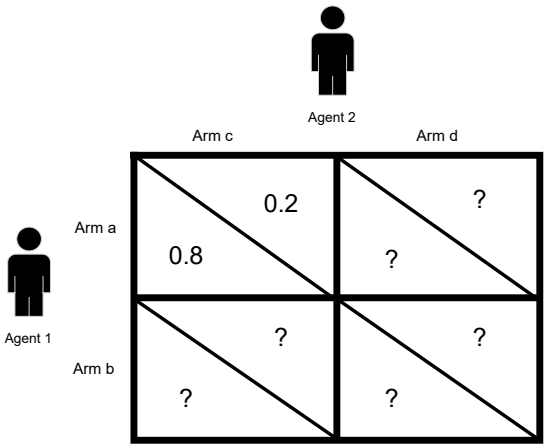


Figure 3.1: An example of unknown games with two players and two arms for each player.

is associated with a finite set of arms (i.e., actions) $\mathcal{A}_n$ with size $A_n := |\mathcal{A}_n|$, where $|\cdot|$ is the cardinality function. The arm set for each player is not required to be identical. Let $\mathcal{A} = \bigotimes_{n=1}^{N} \mathcal{A}_n$ be the space of all such arm sets, and $A \in \mathcal{A}$ be an action profile (i.e., a vector of all players' actions). The reward for player n playing arm $a_n^t \in \mathcal{A}_n$ in round t is determined by function $u_n : \mathcal{A} \to [0, 1]$, which maps the actions of all players to player n's rewards $u_n(a_n^t; a_{-n}^t)$, where $(a_n^t; a_{-n}^t)$ is an abbreviation of $A^t := (a_1^t, \ldots, a_n^t, \ldots, a_N^t)$ with a highlight of player n's action a_n against other players' actions. Note that our algorithm and analyses also work for a time-varying reward function u_n^t. In addition, u_n^t can be determined in either an oblivious way or a non-oblivious (i.e., adaptive) way, corresponding to the oblivious adversary or the non-oblivious adversary in the single-player bandits. In an oblivious way, $\{u_n^t\}_{t>0}$ is chosen at the beginning of the game, while in a non-oblivious way, each u_n^t is determined conditioned on all the players' actions in the past.

One of the main differences between multi-player bandits and single-player bandits is the measurability of the rewards. If we are in the single-player bandits, regardless of whether u_n^t is determined obliviously or non-obliviously, the reward of each arm in each round t is determined at the beginning of that round, before the player plays an action. However, in the multi-player bandits, as the reward of each arm for each player is conditioned on other players' actions, the reward of each arm in each round cannot be determined until all players have played an action in that round.

In each round $t = 1, \ldots, T$, each player $n \in [N]$ selects an arm $a_n^t \in \mathcal{A}_n$ by sampling from a *mixed strategy* $p_n^t \in \Delta(\mathcal{A}_n)$, where $p_n^t(a)$ denotes the probability of choosing arm $a \in \mathcal{A}_n$. Then, with a slight abuse of notation, each player observes only her own instantaneous reward, defined as $u_n^t(a_n^t) := u_n(a_n^t; a_{-n}^t)$, which depends on her action and the joint actions of the other players. For ease of algorithmic description and analysis, we also define the equivalent notion of instantaneous loss by $y_a^t := 1 - u_n(a; a_{-n}^t)$, when context is clear for the loss incurred by player n when playing action a at round t. Each player has access only to her own realized loss or reward, without observing the actions or even the number of the other players. The goal of each player is to maximize her cumulative reward over the T rounds.

3.3.2 Problem Formulation

Since each player has limited knowledge of the environment, incurring regret is inevitable. In what follows, we present the game-theoretic counterparts of the regret notions introduced in Sec. 2.2.3.

The most basic regret notion is the *external regret* [20]. Let $\mathbf{1}[a_n^t = a]$ be the indicator function that returns 1 if a is the played arm in round t and 0 otherwise. The external regret $R_n^{\text{ext}}(T)$ for player n compares the cumulative reward of a learning algorithm with that of a set of competitors that always play a fixed arm up to round T, which is defined as follows:

$$R_n^{\text{ext}}(T) := \max_{a' \in \mathcal{A}_n} \sum_{t=1}^{T} u_n(a'; a_{-n}^t) - \sum_{t=1}^{T} \sum_{a \in \mathcal{A}_n} \mathbf{1}[a_n^t = a] u_n(a; a_{-n}^t),$$

However, only minimizing the external regret cannot guarantee that the plays of players will reach an (ϵ-)*correlated equilibrium (CE)* (see Definition 2.3).

To achieve this objective, we consider a more general notion of regret, called *swap regret* [12], which generalize the external regret by a swap function $F_n : \mathcal{A}_n \to \mathcal{A}_n$ that takes $a \in \mathcal{A}_n$ as input and outputs $a' \in \mathcal{A}_n$. Let $\mathcal{F}_n$ be the set of all possible F_n. Then, the instantaneous swap regret for player n with $\mathcal{F}_n$ up to round T is defined as follows:

$$R_n^{\text{swa}}(T, \mathcal{F}_n) = \max_{F \in \mathcal{F}} \sum_{t=1}^{T} \sum_{a \in \mathcal{A}_n} \mathbf{1}[a_n^t = a] \left(u_n(F(a); a_{-n}^t) - u_n(a; a_{-n}^t) \right). \tag{3.1}$$

We can boil down the swap regret to the external regret by restricting $\mathcal{F}_n$ to be a set of A_n functions such that for any $a \in \mathcal{A}_n$, $F_a : \mathcal{A}_n \to a$.

Thus, by minimizing the swap regret of a learning algorithm for a general $\mathcal{F}_n$ of any possible mappings F, we can

1. ensure that the algorithm maintains a bounded performance gap relative to a broader class of competitors, and
2. guarantee that the empirical joint distribution of actions converges to an ϵ-CE.

More specifically, for objective (2), the empirical joint distribution

$$\hat{P}^T(A) := \frac{1}{T}\sum_{t=1}^{T} \mathbf{1}[A^t = A], \forall A \in \mathcal{A}$$

converges to an ϵ-CE if the swap regret is bounded with high probability. However, existing results typically provide only pseudo-regret bounds, which are insufficient to establish high-probability convergence of $\hat{P}^T(A)$.

3.4 The Algorithmic Framework

3.4.1 Framework Setup

Our high-probability swap-regret-minimizing algorithmic framework is based on the external-regret-to-swap-regret techniques introduced by [12], calling A_n OMD algorithms introduced in Sec. 2.4.2 with implicit exploration techniques [64, 85] as subroutines. Each subroutine maintains a meta-distribution, and the action selection probability is calculated from the meta-distributions. The observed reward or loss will be assigned proportionally to each subroutine to update the meta-distributions.

For each player n, we define a meta-distribution $q_a^t \in \Delta(\mathcal{A}_n)$ for each arm $a \in \mathcal{A}_n$ such that $q_{a,a'}^t \in [0,1]$ and $\sum_{a' \in \mathcal{A}_n} q_{a,a'}^t = 1$. Each meta-distribution q_a^t is updated by its corresponding OMD subroutine, and we use the same index a to refer to the subroutine managing q_a^t.

Let Q_n^t be the $A_n \times A_n$ matrix whose a-th row is $(q_a^t)^\top$. The sampling distribution $p_n^t \in \Delta(\mathcal{A}_n)$ is then defined as the stationary distribution satisfying the fixed-point

condition:

$$(p_n^t)^\top = (p_n^t)^\top Q_n^t, \tag{3.2}$$

where p_n^t is a row vector of $p_n^t(a), \forall a \in \mathcal{A}_n$ and $\sum_{a \in \mathcal{A}_n} p_n^t(a) = 1$. That is, for each $a' \in \mathcal{A}_n$, we have $p_{a'}^t = \sum_{a \in \mathcal{A}_n} p_a^t q_{a,a'}^t$, which is similar to the calculation of the stationary distribution of a Markov process with the transition matrix being Q_n^t. The intuition behind (3.2) is to make the probability of playing arm $a' \in \mathcal{A}_n$ directly according to p_n^t equivalent to the probability of first sampling any arm $a \sim p_n^t$ and then playing a' according to q_a^t.

The suffix "IX" of OMD-LCE-IX stands for implicit exploration, which is justified by the γ_t-biased reward estimator defined as follows. Denote by $Y_{a,a'}^t := \frac{\mathbf{1}[a_n^t = a'] p_a^t q_{a,a'}^t}{p_{a'}^t}(1 - u_n^t(a'))$ the loss of arm a' observed by subroutine a. Let γ_t be a non-negative and non-increasing parameter. We then define the γ_t-biased estimated loss for $Y_{a,a'}^t$ as follows:

$$\hat{Y}_{a,a'}^t := \frac{Y_{a,a'}^t}{q_{a,a'}^t + \gamma_t}.$$

This bias factor γ_t is introduced in [64, 85], which is used to smooth the meta-distributions so that the arms with high loss in the past can still be chosen occasionally for exploration.

Next, we instantiate the above algorithmic framework with the negative entropy and the negative Tsallis entropy as follows.

3.4.2 LCE-IX

The LCE-IX algorithm is presented in (3.1), where LCE stands for Learning for Correlated Equilibrium. The algorithm invokes K_n instances of the Exp3-IX algorithm [64, 85] as subroutines. Each subroutine maintains a meta-distribution over the action set $\mathcal{A}_n$, and the overall action-selection probability is derived from these meta-distributions. After observing the outcome, the reward (or loss) is allocated proportionally to the relevant subroutines, which then update their meta-distributions according to the following rule:

$$q_{a,a'}^{t+1} = \frac{\exp\left(-\eta \hat{L}_{a,a'}^{t}\right)}{\sum_{a'' \in \mathcal{A}_n} \exp\left(-\eta \hat{L}_{a,a''}^{t}\right)}. \tag{3.3}$$

Here, $\hat{L}_{a,a'}^{t} = \hat{L}_{a,a'}^{t-1} + \hat{Y}_{a,a'}^{t}$ denotes the estimated cumulative loss incurred by subroutine a when simulating action a' up to round t. In fact, the updating rule can be derived by setting $\psi(p) = \sum_{a \in A_n} p(a) \ln p(a)$ in the OMD algorithm (see Example 2.4).

Algorithm 3.1 The LCE-IX algorithm

Input: $n, \mathcal{A}_n, \eta, \gamma_t$
// Initialization
Set $q_{a,a'}^{1} = \frac{1}{A_n}$ and $\hat{L}_{a,a'}^{0} = 0, \forall a, a' \in \mathcal{A}_n$
for $t = 1, \ldots, T$ **do**
  // Compute the sample distribution, play arms, and observe rewards
  Calculate p_n^t based on (3.2)
  Play an arm $a_n^t \sim p_n^t$
  Observe reward X_n^t
  // Update each meta-distribution
  for $a \in \mathcal{A}_n$ **do**
    $Y_{a,a'}^{t} := \frac{\mathbf{1}[a_n^t = a'] p_a^t q_{a,a'}^t}{p_{a'}^t}(1 - u_n^t(a')), \forall a' \in \mathcal{A}_n$
    $\hat{Y}_{a,a'}^{t} := \frac{Y_{a,a'}^{t}}{q_{a,a'}^{t} + \gamma_t}, \forall a' \in \mathcal{A}_n$
    $\hat{L}_{a,a'}^{t} = \hat{L}_{a,a'}^{t-1} + \hat{Y}_{a,a'}^{t}, \forall a' \in \mathcal{A}_n$
    Calculate q_a^{t+1} based on (3.3)
  end for
end for

3.4.3 OMD-LCE-IX

Next, we show how to adapt the OMD with Tsallis entropy to the swap-regret-minimizing framework, as shown in Alg. 3.2. For each player n, OMD-LCE-IX calls OMD with Tsallis entropy $\psi_a(q) := \frac{1 - \sum_{a' \in \mathcal{A}_n} q(a')^\beta}{1-\beta}$ as each subroutine $a \in \mathcal{A}_n$ for any $\beta \in (0, 1)$ and distribution $q \in \Delta(\mathcal{A}_n)$. At the very beginning, each subroutine calculates a meta-distribution q_a^1 according to

$$q_a^1 = \arg\min_{q \in \Delta(\mathcal{A}_n)} \psi_a(q).$$

Algorithm 3.2 The OMD-LCE-IX algorithm

Input: n, A_n, η, γ_t
// Initialization
Set $q^t_{a,a'} = \frac{1}{A_n}$ and $\hat{L}^0_{a,a'} = 0, \forall a, a' \in \mathcal{A}_n$
for $t = 1, \ldots, T$ **do**
  // Compute the sample distribution, play arms and observe rewards
  Calculate p^t_n based on (3.2)
  Play an arm $a^t_n \sim p^t_n$
  Observe reward X^t_n
  // Update each meta-distribution
  for $a \in \mathcal{A}_n$ **do**
    $Y^t_{a,a'} := \frac{\mathbf{1}[a^t_n = a']p^t_a q^t_{a,a'}}{p^t_{a'}}(1 - X^t_n), \forall a' \in \mathcal{A}_n$
    $\hat{Y}^t_{a,a'} := \frac{Y^t_{a,a'}}{q^t_{a,a'} + \gamma_t}, \forall a' \in \mathcal{A}_n$
    Calculate $\tilde{q}^{t+1}_a$ in the dual space according to

$$\frac{1}{(\tilde{q}^{t+1}_{a,a'})^{1-\beta}} = \frac{1}{(q^t_{a,a'})^{1-\beta}} + \frac{1-\beta}{\beta}\eta\hat{Y}^t_{a,a'}$$

    Calculate q^{t+1}_a by projecting $\tilde{q}^{t+1}_a$ back to the primal space according to

$$\frac{1}{(q^{t+1}_{a,a'})^{1-\beta}} = \frac{1}{(\tilde{q}^{t+1}_{a,a'})^{1-\beta}} + \lambda^t_a$$

  end for
end for

By using the method of Lagrange multipliers, each entry $q^1_{a,a'}$ of q^1_a can be solved as $q^1_{a,a'} = \frac{1}{A_n}$. Then, OMD-LCE-IX first calculates P^t_n based on (3.2) and constructs a γ_t-biased loss estimator based on the played arm and its reward.

Then, as in (2.9) in Example 2.5, we can obtain a two-step process to obtain q^{t+1}_a in the subsequent rounds for each subroutine:

$$\tilde{q}^{t+1}_a = \underset{q \in \text{dom}(\psi_a)}{\arg\min}\ \eta \left\langle q, \hat{Y}^t_a \right\rangle + D_{\psi_a}\left(q, q^t_a\right)$$
$$q^{t+1}_a = \arg\min_{q \in \Delta(\mathcal{A}_n)} D_{\psi_a}\left(q, \tilde{q}^{t+1}_a\right),$$

where $\hat{Y}^t_a$ is the vector of $\hat{Y}^t_{a,a'}$ for all $a' \in \mathcal{A}_n$.

The idea of the above two-step update rule is first to calculate a solution $\tilde{q}^{t+1}_a$ in the dual space (i.e., the domain of the Tsallis entropy), and then project back to

the primal space (i.e., $\Delta(\mathcal{A}_n)$) by using the Bregman divergence. Notice that the solution to the first step is the solution of the following equation:

$$\eta \hat{Y}^t_{a,a'} + \nabla\psi_a\left(\tilde{q}^{t+1}_a\right) - \nabla\psi_a\left(q^t_a\right) = 0. \tag{3.4}$$

Similarly, we can evaluate the second step explicitly, and the two-step update rule becomes the solutions for the following two equations:

$$\frac{1}{(\tilde{q}^{t+1}_{a,a'})^{1-\beta}} = \frac{1}{(q^t_{a,a'})^{1-\beta}} + \frac{1-\beta}{\beta}\eta\hat{Y}^t_{a,a'},$$
$$\frac{1}{(q^{t+1}_{a,a'})^{1-\beta}} = \frac{1}{(\tilde{q}^{t+1}_{a,a'})^{1-\beta}} + \lambda^t_a,$$

where λ^t_a is a constant from the method of Lagrange multipliers that makes $q^{t+1}_a := \{\tilde{q}^{t+1}_{a,a'} : \forall a' \in \mathcal{A}_n\}$ a distribution, i.e., $\sum_{a'\in\mathcal{A}_n} q^{t+1}_{a,a'} = 1$, as shown in Lines 13 to 14 of Alg. 3.2. The possible values of γ_t, β and η can be found in Theorem 3.5, and the value of λ^t_a can be computed efficiently by a binary search or Newton methods.

3.5 Analytical Results

3.5.1 Concentration Inequality

Notations. As the regret analysis is for any fixed individual player $n \in [N]$, without confusion, we drop the subscript n in some notations for brevity when the context is clear. For example, we will use $y^t_a := 1 - u^t_n(a)$ to represent the loss suffered by player n. Let $\mathcal{G}_t$ denote the σ-algebra generated by the history information of all players up to round t, i.e., $\mathcal{G}_t := \sigma\left(\{a^1_n, r^1_n, \ldots, a^t_n, r^t_n\}_{n\in\mathcal{N}}\right)$.

We first state a novel concentration bound for the γ_t-biased loss estimator, which shows that the cumulative gap between the biased loss estimator $\hat{Y}^t_{a,a'}$ and $p^t_n(a)y^t_{a'}$ for each player $n \in \mathcal{N}$ is bounded with a high probability. In the following, we will slightly abuse the notation β with subscripts and superscripts to represent a random variable. In cases where no subscripts or superscripts are associated with β, it refers to the parameter utilized in the OMD-LCE-IX algorithm.

Lemma 3.1. Let $\delta \in (0,1)$ and let $\beta^t_{a,a'}$ be any nonnegative and non-increasing (over time t) $\mathcal{G}_{t-1}$-measurable random variables (i.e., given $\mathcal{G}_{t-1}$, $\beta^t_{a,a'}$ is determined)

satisfying $\beta^t_{a,a'} \leq 2\gamma_t$ for all pairs $a, a' \in \mathcal{A}_n$. With probability at least $1-\delta$, we have the following inequality held:

$$\sum_{t=1}^{T} \sum_{a\in\mathcal{A}_n} \sum_{a'\in\mathcal{A}_n} \beta^t_{a,a'} \left(\hat{Y}^t_{a,a'} - p^t_n(a) y^t_{a'}\right) \leq \ln \frac{1}{\delta} \,. \tag{3.5}$$

Proof Sketch. We only give a proof sketch here; the detailed proof can be found in Appendix A.2.2. First, we construct a sequence of random variables $\{Z_t\}_{t\geq 0}$, where $Z_t := \exp\left\{\sum_{s=1}^{t} \sum_{a\in\mathcal{A}_n} \sum_{a'\in\mathcal{A}_n} \beta^s_{a,a'} \left(\hat{Y}^s_{a,a'} - p^s_a y^s_{a'}\right)\right\}$ for $t > 0$ and $Z_0 = 1$, and then prove that $\{Z_t\}_{t\geq 0}$ is a supermartingale with respect to filtration $\{\mathcal{G}_t\}_{t\geq 0}$, i.e., $\mathbf{E}\left[Z_t | \mathcal{G}_{t-1}\right] \leq Z_{t-1}$. By the property of supermartingale, we have that $\mathbf{E}[Z_T] \leq \cdots \leq \mathbf{E}[Z_0] = 1$. Finally, the lemma follows the Markov inequality:

$$\begin{aligned}
&\Pr\left(\sum_{t=1}^{T} \sum_{a\in\mathcal{A}_n} \sum_{a'\in\mathcal{A}_n} \beta^t_{a,a'} \left(\hat{Y}^t_{a,a'} - p^t_n(a) y^t_{a'}\right) \geq \ln \frac{1}{\delta}\right) \\
&= \Pr\left(\exp\left\{\sum_{t=1}^{T} \sum_{a\in\mathcal{A}_n} \sum_{a'\in\mathcal{A}_n} \beta^t_{a,a'} \left(\hat{Y}^t_{a,a'} - p^t_n(a) y^t_{a'}\right)\right\} \geq \frac{1}{\delta}\right) \\
&\leq \frac{\mathbf{E}[Z_T]}{\delta^{-1}} \leq \delta.
\end{aligned}$$

□

The proof for Lemma 3.1 is refined beyond the IX concentration bounds studied in [85]. The original approach used in [85] is for external regret. However, we cannot simply adapt their concentration bound to analyze the instantaneous swap regret for the following reasons. First, while the swap regret can only be represented as the aggregate of external regrets of subroutine algorithms *in expectation*, this representation does not hold for the instantaneous swap regret. Second, in the original concentration bound, the probability is taken with respect to only one player's randomness, which is unsuitable for the unknown-game bandits setting, as the reward/loss for each player is dependent on all players' actions. To address the issue, Lemma 3.1 considers the A_n subroutines as a whole, and proves a supermartingale between the sum of IX loss estimators for each meta-distribution and the general swapped loss with respect to all players' randomness. The following lemma is a direct result of Lemma 3.1, which is essential for the swap-regret analysis.

Lemma 3.2. Let $\delta \in (0,1)$. Each of the following holds with probability at least $1-\delta$:

$$\sum_{t=1}^{T} \sum_{a\in\mathcal{A}_n} \sum_{a'\in\mathcal{A}_n} \gamma_t \left(\hat{Y}^t_{a,a'} - p^t_n(a) y^t_{a'}\right) \le \ln(\frac{1}{\delta}), \tag{3.6}$$

$$\sum_{t=1}^{T} \sum_{a\in\mathcal{A}_n} \sum_{a'\in\mathcal{A}_n} (q^t_{a,a'})^{1-\beta} \left(\hat{Y}^t_{a,a'} - p^t_n(a) y^t_{a'}\right) \le \frac{1}{\gamma_T} \ln(\frac{1}{\delta}), \tag{3.7}$$

and simultaneously for all $F \in \mathcal{F}_n$,

$$\sum_{t=1}^{T} \sum_{a\in\mathcal{A}_n} \left(\hat{Y}^t_{a,F(a)} - p^t_n(a) y^t_{F(a)}\right) \le \frac{A_n}{\gamma_T} \ln(\frac{A_n}{\delta}). \tag{3.8}$$

Proof. (3.6) is obtained by invoking Lemma 3.1 with $\beta^t_{a,a'} = \gamma_t$ and $\beta^t_{a,a'} = 2\gamma_t$, respectively. (3.7) is obtained by invoking Lemma 3.1 with $\beta^t_{a,a'} := \gamma_T (q^t_{a,a'})^{1-\beta}$ for all $a, a' \in \mathcal{A}_n$. (3.8) is obtained by invoking Lemma 3.1 with $\beta^t_{a,a'} := \gamma_T \mathbf{1}[a' = F(a)]$ for all $a, a' \in \mathcal{A}_n$, and a union bound for all $F \in \mathcal{F}_n$. □

We also need the following lemma to bound the gap between $\bar{L}^T_{a,F(a)} := p^t_n(a) y^t_{F(a)}$ and $\tilde{L}^T_{a,F(a)} := \mathbf{1}[a^t_n = a] y^t_{F(a)}$ simultaneously for all $F \in \mathcal{F}_n$ with high probability:

Lemma 3.3. Let $\delta \in (0,1)$. With probability at least $1-\delta$, the following inequality holds simultaneously for all $F \in \mathcal{F}_n$,

$$\sum_{t=1}^{T} \sum_{a\in\mathcal{A}_n} \left(p^t_n(a) y^t_{F(a)} - \mathbf{1}[a^t_n = a] y^t_{F(a)}\right) \le \sqrt{2TA_n \ln \frac{A_n}{\delta}}\,. \tag{3.9}$$

Proof. Fix any $F \in \mathcal{F}_n$, let $S_t(F) := \sum_{s=1}^{t} \sum_{a\in\mathcal{A}_n} p^s_n(a) y^s_{F(a)} - \sum_{s=1}^{t} \sum_{a\in\mathcal{A}_n} \mathbf{1}[a^s_n = a] y^s_{F(a)}$ and $S_0(F) = 0$. Recall that $\mathbf{E}_{n,t-1}[\cdot] := \mathbf{E}_{t-1}\left[\cdot \mid a^t_{-n}\right]$, where a^t_{-n} is the actions played by other players in round t. Then, we have that

$$\sum_{a\in\mathcal{A}_n} p^t_n(a) y^t_{F(a)} = \mathbf{E}_{n,t-1}\left[\sum_{a\in\mathcal{A}_n} \mathbf{1}[a^t_n = a] y^t_{F(a)}\right],$$

which indicates that $S_t(F)$ is a martingale sequence, i.e., $\mathbf{E}_{n,t-1}[S_t] = S_{t-1}$. For

each t, notice that $S_t(F) - S_{t-1}(F)$ is bounded as follows:

$$\left| \sum_{a \in \mathcal{A}_n} p_n^t(a) y_{F(a)}^t - \sum_{a \in \mathcal{A}_n} \mathbf{1}[a_n^t = a] y_{F(a)}^t \right| \leq 1,$$

which is due to the fact that $\sum_{a \in \mathcal{A}_n} p_n^t(a) y_{F(a)}^t \leq \sum_{a \in \mathcal{A}_n} p_n^t(a) = 1$, and $\sum_{a \in \mathcal{A}_n} \mathbf{1}[a_n^t = a] y_{F(a)}^t \leq \sum_{a \in \mathcal{A}_n} \mathbf{1}[a_n^t = a] = 1$.

Then, applying Azuma's inequality (Lemma A.5), we have that for any fixed $F \in \mathcal{F}$:

$$\Pr(S_T(F) - S_0(F) > \epsilon) \leq \exp\left\{ -\frac{\epsilon^2}{2T} \right\}$$

Let $\frac{\delta}{A_n^{A_n}} = \exp\left\{ -\frac{\epsilon^2}{2T} \right\}$, and we have that

$$\Pr\left(S_T(F) > \sqrt{2TA_n \ln \frac{A_n}{\delta}} \right) \leq \Pr\left(S_T(F) > \sqrt{2T \ln \frac{A_n^{A_n}}{\delta}} \right) \leq \frac{\delta}{A_n^{A_n}}.$$

Since there are totally $A_n^{A_n}$ functions in $\mathcal{F}_n$, taking the union bound for all $F \in \mathcal{F}$ gives

$$\begin{aligned}
\Pr\left(\bigcap_{F \in \mathcal{F}} S_T(F) \leq \sqrt{2TA_n \ln \frac{A_n}{\delta}} \right) &= 1 - \Pr\left(\bigcup_{F \in \mathcal{F}} S_T(F) > \sqrt{2TA_n \ln \frac{A_n}{\delta}} \right) \\
&\geq 1 - \sum_{F \in \mathcal{F}} \Pr\left(S_T(F) > \sqrt{2TA_n \ln \frac{A_n}{\delta}} \right) \\
&\geq 1 - \sum_{F \in \mathcal{F}} \frac{\delta}{A_n^{A_n}} = 1 - \delta,
\end{aligned}$$

□

3.5.2 Regret Bounds

Regret Bounds for LCE-IX

The regret defined in (3.1) for each player $n \in [N]$ playing the LCE-IX algorithm is guaranteed by the following theorem.

Theorem 3.4. Let $\delta \in (0, 1)$. With probability at least $1 - \delta$, $\gamma_t = \frac{\eta}{2} = \frac{1}{2}\sqrt{\frac{\ln \frac{4A_n}{\delta}}{T}}$, the instantaneous swap regret for playing the LCE-IX algorithm over T rounds is

bounded as follows:

$$R_n^{\text{swa}}(T, \mathcal{F}_n) \leq 4A_n\sqrt{T \ln \frac{4A_n}{\delta}} + \sqrt{2TA_n \ln \frac{4A_n}{\delta}} + 2\ln\left(\frac{4}{\delta}\right). \tag{3.10}$$

Proof Sketch. The regret defined in (3.1) can be rewritten in the loss form and can be decomposed as follows:

$$\begin{aligned} R_n^{\text{swap}}(T, \mathcal{F}_n) \leq \max_{F \in \mathcal{F}} & \underbrace{\sum_{a \in \mathcal{A}_n} (L_a^T - \hat{L}_a^T)}_{=:\text{(a)}} + \underbrace{\sum_{a \in \mathcal{A}_n} (\hat{L}_a^T - \hat{L}_{a,F(a)}^T)}_{=:\text{(b)}} \\ & + \underbrace{\sum_{a \in \mathcal{A}_n} (\hat{L}_{a,F(a)}^T - \bar{L}_{a,F(a)}^T)}_{=:\text{(c)}} + \underbrace{\sum_{a \in \mathcal{A}_n} (\bar{L}_{a,F(a)}^T - \tilde{L}_{a,F(a)}^T)}_{=:\text{(d)}}, \end{aligned} \tag{3.11}$$

where $L_a^T := \sum_{t=1}^T \sum_{a' \in \mathcal{A}_n} Y_{a,a'}^t$, $\hat{L}_a^t := \sum_{t=1}^T \sum_{a' \in \mathcal{A}_n} q_{a,a'}^t \hat{Y}_{a,a'}^t$, $\hat{L}_{a,F(a)}^t := \sum_{t=1}^T \hat{Y}_{a,F(a)}^t$, $\bar{L}_{a,F(a)}^T := \sum_{t=1}^T p_n^t(a) y_{F(a)}^t$, and $\tilde{L}_{a,F(a)}^T := \sum_{t=1}^T \mathbf{1}[a_n^t = a] y_{F(a)}^t$. Then, the proof follows the following steps. In the following proof, we consider a fixed γ over time.

1. We bound (a) by $\sum_{a \in \mathcal{A}_n} \sum_{t=1}^T \gamma_t \sum_{a' \in \mathcal{A}_n} \hat{Y}_{a,a'}^t$, which is a straightforward result by the definition of $\hat{Y}_{a,a'}^t$. Then, invoking (3.6) of Lemma 3.2, we have with probability at least $1 - \frac{\delta}{4}$:

$$\text{(a)} \leq \gamma \sum_{t=1}^T \sum_{a' \in \mathcal{A}_n} p_n^t(a) y_{a'}^t + \ln\left(\frac{4}{\delta}\right).$$

2. We bound (b) by $\sum_{t=1}^T \sum_{a \in \mathcal{A}_n} \frac{\ln d}{\eta} + \eta \sum_{t=1}^T \sum_{a \in \mathcal{A}_n} \sum_{a \in \mathcal{A}_n} \hat{Y}_{a,a'}^t$ by a refined analysis leveraging Lemma 2.6. Then, invoking (3.6) of Lemma 3.2 again, we have with probability at least $1 - \frac{\delta}{4}$:

$$\text{(b)} \leq \frac{A_n \ln A_n}{\eta} + \frac{\eta}{2} \sum_{t=1}^T \sum_{a \in \mathcal{A}_n} \sum_{a' \in \mathcal{A}_n} p_n^t(a) y_{a'}^t + \frac{\eta}{2\gamma} \ln\left(\frac{4}{\delta}\right).$$

3. (c) can be bounded directly by invoking (3.8) of Lemma 3.2, and we have

that for all $F \in \mathcal{F}_n$ with probability at least $1 - \frac{\delta}{4}$,

$$(\mathrm{c}) \leq \frac{A_n}{\gamma} \ln\left(\frac{4A_n}{\delta}\right).$$

4. (d) can be bounded by invoking Lemma 3.3, with probability at least $1 - \frac{\delta}{4}$, the following inequality holds simultaneously for all $F \in \mathcal{F}_n$,

$$(\mathrm{d}) \leq \sqrt{2A_n T \ln\left(\frac{4A_n}{\delta}\right)}.$$

5. Finally, combining the three items with union bound and letting $\gamma = \frac{\eta}{2}$, we have with probability at least $1 - \delta$:

$$R_n^{\mathsf{swa}}(T, \mathcal{F}_n) \leq \frac{A_n \ln A_n}{\eta} + \eta T A_n + 2\ln\left(\frac{4}{\delta}\right) + \frac{2A_n}{\eta} \ln\left(\frac{4A_n}{\delta}\right) + \sqrt{2TA_n \ln\frac{4A_n}{\delta}},$$

and the theorem follows from letting $\eta = \sqrt{\frac{\ln \frac{4A_n}{\delta}}{T}}$.

□

Regret Bounds for OMD-LCE-IX

The regret defined in (3.1) for each player $n \in [N]$ playing the OMD-LCE-IX algorithm is guaranteed by the following theorem.

Theorem 3.5. Let $\delta \in (0, 1)$. With probability at least $1-\delta$, $\gamma_t = \sqrt{\frac{\ln(4A_n/\delta)}{T}}, \forall t \in [T]$, $\beta = \frac{1}{2}$, and $\eta = \sqrt{\frac{A_n}{T}}$, the instantaneous swap regret for playing the OMD-LCE-IX algorithm over T rounds is bounded as follows:

$$R_n^{\mathsf{swa}}(T, \mathcal{F}_n) \leq 2A_n\sqrt{T \ln\left(\frac{4A_n}{\delta}\right)} + 2A_n\sqrt{T} + \ln\left(\frac{4}{\delta}\right) + \sqrt{2A_n T \ln\left(\frac{4A_n}{\delta}\right)}. \tag{3.12}$$

Proof Sketch. The regret defined in (3.1) can be rewritten in the loss form as (3.11). In the following proof, we consider a fixed γ over time.

1. We can bound (a) in the same way by $\sum_{a\in\mathcal{A}_n}\sum_{t=1}^{T}\gamma_t\sum_{a'\in\mathcal{A}_n}\hat{Y}^t_{a,a'}$. Invoking (3.6) in Lemma 3.2, we have that with probability $1-\frac{\delta}{4}$:

$$(\text{a}) \leq \gamma T A_n + \ln\left(\frac{4}{\delta}\right),$$

2. To bound (b), we need to invoke Lemma 2.7, which gives $\sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\frac{(A_n)^{1-\beta}-1}{(1-\beta)\eta}+\frac{\eta}{\beta}\sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\sum_{a\in\mathcal{A}_n}(q^t_{a,a'})^{1-\beta}\hat{Y}^t_{a,a'}$. Now, summing over $a\in\mathcal{A}_n$ and invoking (3.7) with $\beta=0.5$, we have with probability $1-\frac{\delta}{4}$ that

$$(b) \leq \frac{2A_n\sqrt{A_n}-2A_n}{\eta}+2\eta T\sqrt{A_n}+\frac{2\eta}{\gamma}\ln\left(\frac{4}{\delta}\right).$$

3. (c) can be bounded directly by invoking (3.8) of Lemma 3.2, and we have that for all $F\in\mathcal{F}_n$ with probability at least $1-\frac{\delta}{4}$,

$$(\text{c}) \leq \frac{A_n}{\gamma}\ln\left(\frac{4A_n}{\delta}\right).$$

4. (d) can be bounded by invoking Lemma 3.3, with probability at least $1-\frac{\delta}{4}$, the following inequality holds simultaneously for all $F\in\mathcal{F}_n$,

$$(\text{d}) \leq \sqrt{2A_nT\ln\left(\frac{4A_n}{\delta}\right)}.$$

5. Finally, combining the above terms with the union bound, we have with probability at least $1-\delta$:

$$\begin{aligned}R_n^{\text{swa}}(T,\mathcal{F}_n) \leq & \gamma T A_n + \frac{2A_n\sqrt{A_n}-2A_n}{\eta}+2\eta T\sqrt{A_n}+\left(1+\frac{2\eta}{\gamma}\right)\ln\left(\frac{4}{\delta}\right)\\ &+\frac{A_n}{\gamma}\ln(\frac{4A_n}{\delta})+\sqrt{2A_nT\ln\left(\frac{4A_n}{\delta}\right)},\end{aligned}$$

where the theorem follows from letting $\gamma=\sqrt{\frac{\ln(4A_n/\delta)}{T}}$ and $\eta=\sqrt{\frac{A_n}{T}}$.

□

LCE-IX is easier to implement than OMD-LCE-IX, due to its closed-form update rule, but OMD-LCE-IX has a practical performance as shown in Sec. 3.6. Both LCE-IX and OMD-LCE-IX achieve a high-probability swap-regret bound of $O\left(A_n\sqrt{T\ln\left(\frac{A_n}{\delta}\right)}+\ln\left(\frac{4}{\delta}\right)\right)$.

As for tightness, the best known lower bound is $\Omega(\sqrt{A_nT\ln A_n})$ [58], which is minimax-optimal in the full-information setting. However, it remains an open question whether this lower bound also holds under bandit feedback.

Next, we present the convergence results toward correlated equilibrium. Since the proofs are identical for both algorithms, we only state the result for OMD-LCE-IX, which achieves a tighter swap-regret bound.

3.5.3 Convergence to Correlated Equilibrium

Bounding the instantaneous swap regret has a significant benefit – it allows us to prove that the OMD-LCE-IX algorithm can converge to a set of correlated equilibria.

Theorem 3.6. If every player $n \in [N]$ plays the OMD-LCE-IX algorithm for T rounds, then the empirical distribution of the joint actions played by all players $\hat{P}^T$ is an ϵ-CE with probability at least $1-\delta$, where $\epsilon = O(\max_{n\in[N]} A_n\sqrt{\frac{\ln\left(\frac{4NA_n}{\delta}\right)}{T}}+\frac{1}{T}\ln\left(\frac{4N}{\delta}\right))$. When $T\to\infty$, the empirical distribution of the joint actions converges to a set of correlated equilibria almost surely.

Proof. The proof follows the following two steps:

1. We first prove the convergence to ϵ-correlated equilibrium with high probability. The proof is straightforward based on Theorems 2.2 and 3.5 with union bounds for all players.

2. The challenge part is to prove the convergence to the correlated equilibrium. We need to use the Borel-Cantelli lemma to convert a high-probability event to an infinitely often event [20].

Step 1. Let $\delta' > 0$. By (3.12), with probability $1-\delta'$, we have $R_n^{\text{swa}}(T,\mathcal{F}_n) \le O(A_n\sqrt{T\ln\frac{4A_n}{\delta'}}+\ln\left(\frac{1}{\delta'}\right))$ for player n. By using the union bound over all the N players and letting $\delta' = \delta/N$, we have that with probability at least $1-\delta$ for all

$n \in [N]$:

$$\max_{F_n:\mathcal{A}_n\to\mathcal{A}_n} \frac{1}{T}\sum_{t=1}^{T}\left[u_n(F_n(a_n^t), a_{-n}^t) - u_n(a_n^t, a_{-n}^t)\right] \le \max_{n\in[N]} \frac{1}{T} R_n^{\text{swap}}(T, \mathcal{F}_n) \le O\left(\max_{n\in[N]} A_n \sqrt{\frac{\ln\frac{4NA_n}{\delta}}{T}} + \frac{1}{T}\ln\left(\frac{4N}{\delta}\right)\right),$$

where the last inequality is due to Theorem 3.5. Then, by Theorem 2.2, we have with probability with at least $1-\delta$, the empirical joint distribution of actions $\hat{P}^T$ converge to the set of $O\left(\max_{n\in[N]} A_n \sqrt{\frac{\ln\frac{4NA_n}{\delta}}{T}} + \frac{1}{T}\ln\left(\frac{4N}{\delta}\right)\right)$-correlated equilibria.

Step 2. Denote by

$$E_T := \left\{\forall n \in [N] : \max_{F_n:\mathcal{A}_n\to\mathcal{A}_n} \sum_{A\in\mathcal{A}} \hat{P}^T(A)\left(u_n\left(F_n(a_n); a_{-n}\right) - u_n\left(A\right)\right) \le O\left(\max_{n\in[N]} A_n \sqrt{\frac{\ln\frac{4NA_n}{\delta}}{T}} + \frac{1}{T}\ln\left(\frac{4N}{\delta}\right)\right)\right\},$$

and we have that $\Pr(E_T) \ge 1-\delta = c$ for some constant $c \in (0,1)$. Thus, $\sum_{T=1}^{\infty} \Pr(E_T) \ge \sum_{T=1}^{\infty} c = \infty$. Now, consider a sequence of independent runs with different time horizons $T = 1 \to \infty$, and the second Borel-Cantelli Lemma states that if $\sum_{T=1}^{\infty} \Pr(E_T) = \infty$, then $\Pr(\limsup_{T\to\infty} E_T) = 1$. That is, when $T \to \infty$, we have

$$\limsup_{T\to\infty} \max_{n\in[N]} \sum_{A\in\mathcal{A}} \hat{P}^T(A)\left(u_n\left(F(a_n); a_{-n}\right) - u_n\left(A\right)\right) \le \limsup_{T\to\infty} O(\max_{n\in[N]} A_n \sqrt{\frac{\ln\frac{4A_nN}{\delta}}{T}} + \frac{1}{T}\ln\left(\frac{4N}{\delta}\right)) = 0$$

almost surely, which indicates the empirical distribution of joint actions converges to the set of correlated equilibria. □

Solving the equation $\epsilon = O\left(\max_{n\in[N]} A_n \sqrt{\frac{\ln\frac{4A_nN}{\delta}}{T}} + \frac{1}{T}\ln\left(\frac{4N}{\delta}\right)\right)$ for T implies that the empirical joint distribution $\hat{P}^T$ for all players meets the definition of an ϵ-CE for the unknown games after $T = \Omega\left(\max_{n\in[N]} \frac{A_n^2 \ln(4A_nN/\delta)}{\epsilon^2} + \frac{\ln\left(\frac{4N}{\delta}\right)}{\epsilon}\right)$ rounds, i.e., the ϵ-CE is achieved.

3.5.4 Time and Space Complexity

In each round, each player needs first to calculate P_n^t based on (4.2), which can be regarded as the calculation of a stationary distribution for an ergodic (i.e., aperiodic and irreducible) Markov chain with A_n states defined by Q_n^t. Such a station-

ary distribution can be precisely computed in $O(A_n^2)$ time [40], and approximately computed in almost linear time [27]. Then, updating each meta-distribution involves finding λ_a^t. Using binary search, the efficient discovery of λ_a^t only takes $O(\ln(m))$ time, where m is the precision of the number of digits. Thus, when the precision is less than $O(\exp(A_n))$, the time complexity of OMD-LCE-IX is still $O(A_n^2)$. Regarding the space complexity, it is $O(A_n^2)$.

3.6 Numerical Experiments

In this section, we compare OMD-LCE-IX with LCE-IX to show the effectiveness of the OMD technique, and we further compare with LCE (i.e., $\gamma_t = 0$ in LCE-IX) to show the effectiveness of the IX technique. We also compare with a recent algorithm with the full-information feedback called BM-Opt-Hedge [23]. BM-Opt-Hedge is basically the subroutine being the OFTRL algorithm with negative entropy as the regularizer function. The results are the average of 100 independent experiments.

We study a wireless medium access game between two wireless devices (i.e., two players), where the two wireless devices are *hidden* from each other (i.e., each device cannot observe the other), and trying to access one unknown channel in each round. Each device has two options in each round: wait for the next round (W) or access in the current round (A). If a device chooses action W, it will receive a reward of 0.

If a device chooses to access (A), the device has an energy cost of 0.2. When only one device successfully accesses the medium, then this device will receive a reward of 0.8. If both devices choose action A, then there is a collision, and the rewards for both devices are -0.2 due to the wasted transmission energy. The reward matrix (unknown to the players) is shown in Table 3.2.

We assume that all the devices do not adopt the RTS/CTS mechanism, an oft-used technique to solve the hidden terminal problem, as it will introduce new problems among its control messages [93] and the game model still applies to the RTS/CTS message itself. Thus, it is quite challenging to improve the received rewards (i.e., the successful access to the channel) for both devices in a distributed way, as *both wireless devices are hidden terminals to each other so that they cannot observe the actions and rewards of each other and they do not know the total number*

of devices.

Table 3.2: The reward matrix for the medium access game

	W	A
W	$(0, 0)$	$(0, 0.8)$
A	$(0.8, 0)$	$(-0.2, -0.2)$

As the swap regret is a generic performance measure, different swap functions can lead to different regret definitions. In this experiment, we show two metrics that reflect two different regret definitions. The first metric is the time-averaged reward, the gap of which from the optimal actions in hindsight reflects the external regret of an online learning algorithm. The other metric is the convergence to the ϵ-CE. Whether or not an ϵ-CE can be reached reflects whether an online learning algorithm can minimize its internal regret.

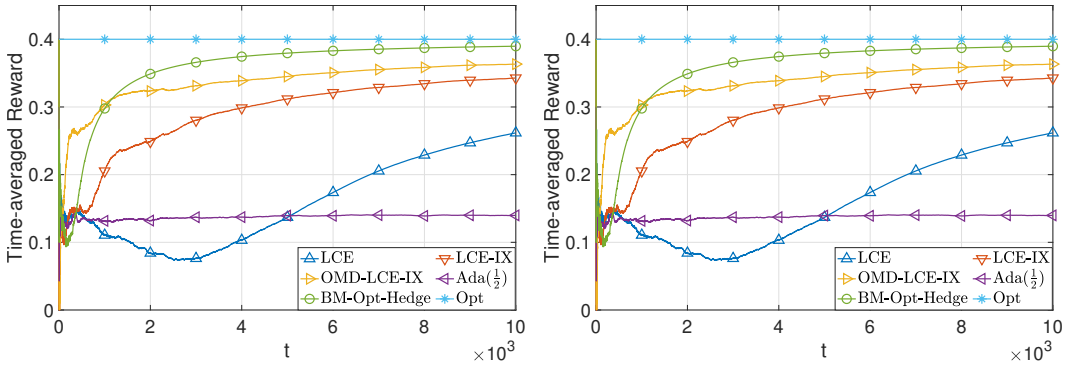
3.6.1 Time-Averaged Reward

We only show the time-averaged reward for all the considered algorithms, as it contains the equivalent information about the cumulative regrets or rewards. For example, if an algorithm has higher time-averaged rewards (or closer to the maximum rewards), then it also has higher cumulative rewards (or lower cumulative regrets).

To compare with a benchmark, we consider an adaptive access technique (denoted by `Ada`) with the prior knowledge about the number of all the hidden devices, which randomly accesses a channel with an initial probability $\frac{1}{2}$ for two devices. If a device fails, the access probability of that device is reduced by half; otherwise, the device uses the initial probability in the next round. `Ada` in our experiments can achieve better performance than the distributed coordination function of IEEE 802.11 used by current WiFi devices in real-world scenarios, as `Ada` in our experiments can set an appropriate initial probability to achieve high throughput. Therefore, `Ada` can be a good benchmark with partial prior knowledge to show the effectiveness of the swap-regret-minimizing algorithms.

In addition, the maximum time-averaged reward (denoted by `Opt`) of 0.4 can be achieved, by a mediator (e.g., wireless access point) with full prior knowledge which either asks player 1 to play W and player 2 to play A, or asks player 1 to play A and player 2 to play W in each round.

The time-averaged rewards of both players in 1×10^4 rounds are shown in Fig. 3.2. As we can see, LCE-IX outperforms both LCE and $\texttt{Ada}(\frac{1}{2})$ in terms of the faster convergence to $\texttt{Opt}$. This shows the effectiveness of the γ_t-biased estimator in smoothing the reward estimation so that the low-reward arm can still be explored occasionally. We can also see that OMD-LCE-IX converge faster than LCE-IX, showing the effectiveness of the OMD technique. On the other hand, BM-Opt-Hedge achieves the fastest result, but we note that BM-Opt-Hedge is with the full-information feedback.



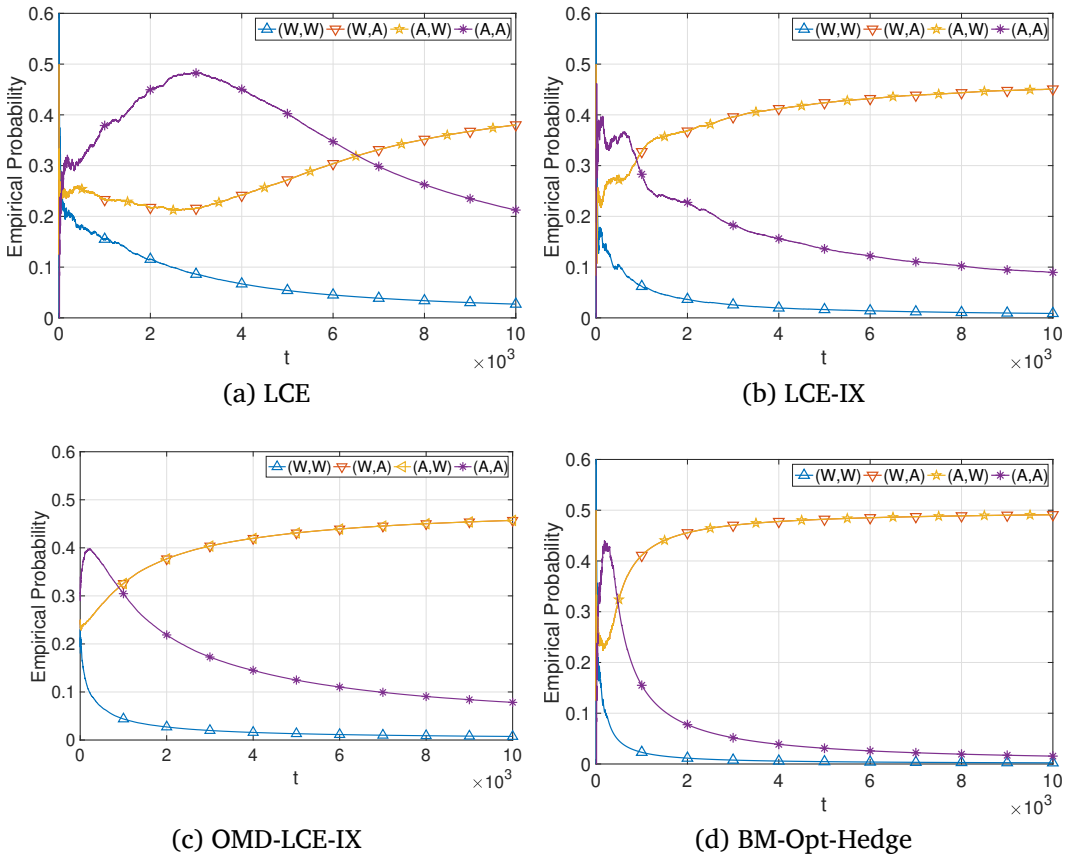
(a) Player 1 (b) Player 2

Figure 3.2: The time-averaged reward for both players.

3.6.2 Convergence to the ϵ-Correlated Equilibrium

The convergence of empirical distribution of joint actions played by the two players in T rounds is shown in Fig. 3.3b, where (W,W) means both players play action W, (W,A) means player 1 plays W and player 2 plays A, and so on. We take the result of LCE-IX to explain the convergence to the CE. The final results in Fig. 3.3a are $\hat{P}^T(\text{W,W}) = 0.0045$, $\hat{P}^T(\text{W,A}) = 0.4687$, $\hat{P}^T(\text{A,W}) = 0.4687$, and $\hat{P}^T(\text{A,A}) = 0.0581$. We can do a simple calculation to verify this empirical distribution is a CE ($\epsilon = 0$). For example, the expected incentives for player 1 to switch from W to A are $\hat{P}^T(\text{W,W}) \cdot u_1(\text{A,W}) + \hat{P}^T(\text{W,A}) \cdot u_1(\text{A,A}) - (\hat{P}(\text{W,W}) \cdot u_1(\text{W,W}) + \hat{P}^T(\text{W,A}) \cdot u_1(\text{W,A})) = -0.0901 < 0$, showing that player 1 does not have incentives to switch from W to A when both players follow the joint distribution $\hat{P}^T$. In the same way, we can verify the empirical joint distribution $\hat{P}^T$ is a CE for both players.

Figs. 3.3b and 3.3c show that both OMD-LCE-IX and LCE-IX have a faster convergence than LCE to a CE, as the empirical probabilities of the optimal action



(a) LCE (b) LCE-IX

(c) OMD-LCE-IX (d) BM-Opt-Hedge

Figure 3.3: The empirical distribution of joint actions by two players in T rounds.

pairs of (A,W) and (W,A) increase faster than that of LCE. This again shows the effectiveness of the γ_t-biased estimator in controlling the variation of the reward estimation.

On the other hand, with full-information feedback, BM-Opt-Hedge can achieve a faster convergence rate than OMD-LCE-IX and LCE-IX. It remains to be explored whether the techniques used in BM-Opt-Hedge can be adapted to the bandit-feedback model to accelerate the convergence rate for swap regret.

3.7 Conclusion

In this chapter, we consider the unknown games with bandit feedback, and present an algorithmic framework based on the swap-regret-minimizing techniques [12], calling OMD algorithms with implicit exploration [85] as subroutines. With regard to the randomness of all agents' actions, we provided the high-probability

bounds for the instantaneous swap regret, which further shows the convergence to the correlated equilibrium. Furthermore, we conducted numerical experiments to show the possible applications in wireless access control.

Regarding future work, we aim to close the gap between the upper and lower bounds for swap regret. We conjecture the swap-regret bound of $O(A_n\sqrt{T})$ is minimax optimal, and will refine the lower bound for swap regret in the bandit settings.

Chapter 4

Faster Convergence for Swap Regret

4.1 Introduction

In this chapter, we continue our study of the unknown-game bandit framework introduced in the previous chapter [9, 12, 20, 58, 52]. This framework captures the core challenges underlying many problems in computer and communication networks, including congestion control [63, 89, 53] and heterogeneous network selection [54, 86, 72]. In end-to-end congestion control, each (TCP) flow must decide the sending rate to the network, treating the network as a black box. The only available knowledge is the feedback (e.g., throughput, RTT, jitters, etc.) observed from prior decisions. Similarly, in the heterogeneous wireless network selection problem, each client must select a network to attach to and can only observe the feedback (e.g., throughput, packet loss rate, etc.) from the attached network.

A commonality in these problems is that multiple agents independently make *uncoupled* decisions, each striving to optimize its own objectives in competition for limited resources. In end-to-end congestion control, if multiple agents send more packets than the network can handle, congestion occurs, leading to packet loss. Similarly, in heterogeneous network selection, if multiple devices attach to the same network, the throughput for all those devices is adversely affected. The term "uncoupled" indicates that each agent interacts independently with a black-box environment, basing its strategy solely on feedback from its own actions. Due to protocol constraints or privacy considerations, direct communication between agents is restricted, necessitating independent optimization with limited information.

Formally, we consider unknown general-sum games, or black-box games as studied in [81, 102], where a set of agents are playing an unknown general-sum game repeated for T rounds. In each round, each agent selects an action and observes the corresponding reward, where the reward is the expected value over all possible actions of the other players as in [102]. The game is *unknown*, because each agent is unaware of the underlying game structure, the number of other agents, or their actions. The only information available to each agent is the observed reward for the played action.

The objective for each agent is to minimize *swap regret* [12]. Swap regret is a stronger notion than external regret, as it compares against a broader class of competitors that include those considered by external regret. Minimizing swap regret offers two significant advantages. First, a no-swap-regret algorithm demonstrates robustness against a wider range of competitors compared to a no-external-regret algorithm, where *no regret* implies that time-averaged regret vanishes as time approaches infinity. Second, minimizing swap regret can lead to convergence to a set of *correlated equilibria (CE)* [10], a concept more general than the well-known Nash equilibrium. CE ensures that no agent finds it beneficial to unilaterally change their strategy given their recommended action and the joint distribution of all agents' actions, thereby achieving optimal overall performance for all agents.

In network applications, the slow convergence rate often poses a challenge for learning algorithms. However, unknown-game bandits offer a potential solution with faster convergence rates. For these bandits, the sum of external regret across all agents can scale with time as $\tilde{O}(T^{\frac{1}{4}})$ [102], when the reward observed by player is an expectation over the actions of the other players, implying a bound of $\tilde{O}(T^{-\frac{3}{4}})$ on the sum of *time-averaged* external regret. In contrast, in scenarios with full-information feedback (where rewards for unplayed actions are also observable), the time-dependence of swap regret can be significantly reduced to $O(\ln T)$. However, for bandit feedback, the best-known swap-regret bound has a time-dependence of $O(T^{\frac{1}{2}})$ [61, 58, 52]. This raises a fundamental question: can we achieve a faster convergence rate of swap regret for unknown-game bandits with expected rewards over opponents' actions?

Our work makes a significant contribution by narrowing the gap in the literature, achieving a $\tilde{O}(T^{\frac{1}{4}})$ time-dependence for swap regret. This faster convergence is realized through the adoption of OFTRL-LogBar-Bandit algorithms by all agents. Our approach refines the BM-OFTRL-LogBar method from [6] to accommodate

bandit feedback. This refinement is non-trivial; it involves careful design of the reward estimator and prediction vectors to ensure the convergence of the OFTRL algorithm while accelerating the convergence rate. Further details can be found in Sections 4.4 and 4.5. Our algorithm builds upon the swap-regret-minimizing framework proposed by [12], utilizing A_n OFTRL subroutines with logarithmic barrier regularizers, where A_n denotes the number of actions available to agent n. Additionally, we demonstrate in Sec. 4.6 the efficacy of our approach in the context of heterogeneous network selection with both numerical and simulation-based experiments.

4.2 Related Works

Learning for Unknown Games: The pursuit of strategies for unknown games traces its roots back to the fictitious play in two-player zero-sum games [14, 91], which relies on knowledge of opponents' past plays. However, the effective tackling of challenges posed by unknown games only became feasible with the advent of online learning techniques. This development uncovered an intrinsic connection between game equilibria and strategies for regret minimization. Specifically, minimizing external regret can lead to Nash equilibria in two-player zero-sum games and coarse correlated equilibria in multi-player general-sum games [20]. Furthermore, minimizing swap regret, a stronger notion than external regret, has been shown to lead to correlated equilibria in multi-agent general-sum games [12, 90, 30].

When considering the observability of rewards, there are two classic feedback models: *full-information feedback* and *bandit feedback*. While steady progress has been made in learning general-sum games under the full-information model [66, 88, 23, 31, 5, 39], extending these results to bandit feedback poses challenges due to the limited information available in each round. The earliest work on bandit feedback dates back to [9], which introduced an exponential-weighting-based technique for minimizing external regret.

To achieve the correlated equilibrium, the authors of [94] proposed an algorithm but with an exponential computation complexity. Subsequently, the authors of [12] introduced the stronger notion of swap regret and devised a framework to efficiently transform external-regret-minimizing algorithms into swap-regret-

minimizing ones with a polynomial computation complexity. Building on this framework, subsequent research has aimed to improve the swap regret bounds for bandit feedback [58, 61, 52], resulting in the recognition that the best swap regret bound for bandit feedback now achieves a time-dependence of $O(T^{\frac{1}{2}})$.

Recently, more efficient conversion techniques with faster convergence rates have been introduced for scenarios where a distribution over actions is played directly in each round, rather than sampling and playing a single action from the distribution [30, 90]. In this chapter, we still focus on playing a single action in each round. Although it is known that $O(T^{\frac{1}{2}})$ is minimax-optimal in adversarial environments, there has been progress, as discussed below, that faster regret convergence is possible for learning agents in the unknown-game environment.

Faster Regret Convergence for Unknown Games: In this chapter, regret convergence refers to the vanishing of time-averaged regret over time, which is different from the notions of last-iterate convergence [18, 17] and frequent-iterate convergence [70] in literature.

While faster regret convergence for bandits remains an ongoing area of exploration, significant advancements have been made in understanding unknown games with full-information feedback over the last decade. The seminal work by [95] demonstrated that the sum of external regret for all agents can be bounded by $O(1)$, with individual external regret scaling as $O(T^{\frac{1}{4}})$. Building on this foundation, the authors of [23] improved the individual external regret bound to $O(T^{\frac{1}{6}})$ for two-player games and extended these results to swap regret, achieving a time-dependence of $O(T^{\frac{1}{4}})$.

Further progress was made by [31], who demonstrated that $O(\ln T)$ dependence for individual external regret is achievable in multi-player general-sum games. Recently, the authors of [5, 6] achieved breakthroughs in the swap regret, showing that a time-dependence of $O(\ln T)$ can be attained.

However, all the above progress is for full-information feedback. Regarding the bandit feedback, there is still a large gap in the literature. The only results are from [102], where they proved that the sum of external regret for two-player zero-sum games enjoys a time-dependence of $O(T^{\frac{1}{4}})$. Neither the individual external regret nor the swap regret is guaranteed. Thus, it remains an open question in the literature whether the time-dependence of $\tilde{O}(T^{\frac{1}{2}})$ for sum of swap regret over players can be further improved.

In this chapter, we take a step forward to this open question by showing a

swap regret with time-dependence of $\tilde{O}(T^{\frac{1}{4}})$ in a bandit setting where the observed reward for the played action is the expected reward against all possible opponent actions. Given that swap regret is a stronger measure than external regret and is always non-negative, our results also guarantee that individual external regret exhibits a time-dependence of $\tilde{O}(T^{\frac{1}{4}})$.

4.3 Problem Formulation

We consider general-sum games involving N agents, repeated over T rounds. Each agent $n \in [N] := \{1, \ldots, N\}$ has an action set $\mathcal{A}_n$ of finite size $A_n := |\mathcal{A}_n|$ and a reward function $u_n : \mathcal{A} \to [0, 1]$, which maps joint actions of all agents $\mathcal{A} := \bigotimes_{n=1}^{N} \mathcal{A}_n$ to values in the range $[0, 1]$.

In each round t, each agent $n \in [N]$ plays an action $a_n^t \in \mathcal{A}_n$ according to a mixed strategy $p_n^t \in \Delta(\mathcal{A}_n) := \{p \in \mathbb{R}_{\geq 0}^{A_n} : \sum_{a \in \mathcal{A}_n} p(a) = 1\}$, i.e., a probability distribution among action set $\mathcal{A}_n$. Unlike in Chapter 3 where each player observes an instantaneous reward (i.e., without expectation), here the agent observes the expected reward for the played action, i.e., $u_n^t(a_n^t) := \mathbf{E}_{a_{-n}^t \sim p_{-n}^t}[u_n(a_n^t; a_{-n}^t)]$, where a_{-n}^t are the actions chosen by agents other than n, and p_{-n}^t are their corresponding mixed strategies.

Note that all agents operate in a black-box setting with bandit feedback, meaning they have limited knowledge about the environment, such as the underlying game structure, the number of agents, or the actions of other agents. The only information available to them is the observed rewards for their own played actions in each round. Furthermore, the reward for each agent depends on the joint actions of all agents in each round, creating a competitive environment. Such a general game setting is applicable to numerous critical network problems, including end-to-end congestion control and heterogeneous network selection.

In fact, each agent in unknown-game bandits faces a special multi-armed bandit problem. In single-agent bandit problems, the primary objective is to design a learning algorithm to minimize *external regret*, which is the performance gap compared to competitors always playing a fixed action. However, in unknown-game bandits, where multiple agents act adaptively to compete with each other, minimizing external regret alone does not ensure optimal overall performance for all agents. Instead, each agent $n \in [N]$ aims to compare with a broader class of

competitors $\mathcal{F}_n := \{F : \mathcal{A}_n \to \mathcal{A}_n\}$, which results in the following pseudo-regret definition of *swap regret*:

$$R_n^{\text{swa}}(T) = \max_{F \in \mathcal{F}_n} \mathbf{E}\left[\sum_{t=1}^{T} \sum_{a \in \mathcal{A}_n} \mathbf{1}[a_n^t = a]\left(u_n^t(F(a)) - u_n^t(a)\right)\right]. \tag{4.1}$$

We can reduce swap regret to external regret by restricting $\mathcal{F}_n$ to the subset $\{F_a, \forall a \in \mathcal{A}_n : \mathcal{A}_n \to a\}$. Therefore, swap regret is a stricter notion than external regret, and minimizing swap regret will also minimize external regret.

More importantly, if all agents employ a learning algorithm that can minimize the swap regret, their expected empirical joint distribution of plays, i.e., $\mathbf{E}[\frac{1}{T}\sum_{t=1}^{T}\sum_{A \in \mathcal{A}} \mathbf{1}[A^t = A]]$ converges to the set of *ϵ-correlated equilibria (CE)* [20, 12], where $A^t := (a_1^t, \ldots, a_N^t)$ is the joint actions played by all agents in round t. The notion of ϵ-CE is defined in Definition 2.3.

Thus, swap regret is an important performance metric in unknown-game bandits. It not only extends the concept of external regret, used as a performance metric in standard bandits, but also guarantees convergence to ϵ-CE. The primary objective of this work is to achieve a faster convergence rate for swap regret, i.e., minimizing the dependence of $R_n^{\text{swa}}(T)$ on T as much as possible.

4.4 The OFTRL-LogBar-Bandit Algorithm

The OFTRL-LogBar-Bandit algorithm is designed for independent execution by all agents to achieve a faster convergence rate, as described in Alg. 4.1. It refines the full-information optimistic follow-the-regularized leader algorithm BM-OFTRL-LogBar [6], incorporating new prediction vectors and reward estimators tailored for bandit feedback [64, 85].

The main idea of OFTRL-LogBar-Bandit is to use the swap-regret-minimizing framework introduced by [12], calling A_n *optimistic follow-the-regularized-leader* algorithms with the log-barrier regularizer (see Example 2.6) as subroutines. Since OFTRL-LogBar-Bandit runs independently for each agent, we will fix an agent $n \in [N]$ and describe how OFTRL-LogBar-Bandit operates.

Each subroutine is indexed by $a \in \mathcal{A}_n$, and maintains a meta-distribution $q_a^t \in \Delta(\mathcal{A}_n)$ by following the OFTRL framework with log-barrier regularizer. Then, let Q_n^t be a $A_n \times A_n$ stochastic matrix with each row being $(q_a^t)^\top$. The action selection

Algorithm 4.1 The OFTRL-LogBar-Bandit algorithm

Input: $n, \mathcal{A}_n, \eta$
// Initialization
Set $q_a^1(a') = \frac{1}{A_n}, \forall a, a' \in \mathcal{A}_n$
for $t = 1, \ldots, T$ **do**
  // Compute the sample distribution, play arms and observe rewards
  Calculate p_n^t based on (4.2)
  Play an action $a_n^t \sim p_n^t$
  Construct the estimated reward $\hat{u}_n^t$ according to (4.4)
  // Update each meta-distribution
  for $a \in \mathcal{A}_n$ **do**
    Update q_a^{t+1} according to (4.3)
  end for
end for

probability p_n^t is then calculated from the meta-distributions as follows:

$$(p_n^t)^\top = (p_n^t)^\top Q_n^t, \tag{4.2}$$

which is equivalent to calculating the stationary distribution of a Markov chain described by Q_n^t.

Next, we give details on how each subroutine maintains q_a^t. Denote by $\hat{u}_n^t$ the estimated reward vector observed by OFTRL-LogBar-Bandit for all actions in round t, where the construction of $\hat{u}_n^t$ will be introduced later. Then, each subroutine $a \in \mathcal{A}_n$ observes a portion of the estimated reward by $p_n^t(a)\hat{u}_n^t$ and calculates q_a^t by solving the following optimization problem:

$$q_a^t := \underset{q \in \text{relint}\, \Delta(\mathcal{A}_n)}{\arg\max} \left\{ \eta \left\langle q, p_n^{t-1}(a) m_n^t + \sum_{s=1}^{t-1} p_n^s(a) \hat{u}_n^s \right\rangle + \sum_{a' \in \mathcal{A}_n} \ln(q(a')) \right\}, \tag{4.3}$$

where m_n^t is the predictor vector to make FTRL "optimistic". To see this, if one were able to obtain $\hat{u}_n^t$ in advance and let $m_n^t := \hat{u}_n^t$, the best q_a^t can be found. Thus, if one can construct m_n^t closer to $\hat{u}_n^t$, better performance can be achieved.

In this work, we follow the convention in [102] to construct m_n^t as follows but with a key challenge as discussed at the end of this section. Let $\tau_t(a) := \arg\max_{s<t} \{s : \mathbf{1}[a_n^s = a]\}$ denote the last round before t when action a was played. Then, let $m_n^t(a) := u_n^{\tau_t(a)}(a)$ for all $a \in \mathcal{A}_n$, i.e., we set the predictor for each action a to be its last observed reward. For analytical convenience, and without loss of generality,

we set $u_n^0(a) = 0$ and $p_n^0(a) = \frac{1}{A_n}$ for all $a \in \mathcal{A}_n$.

Since we consider bandit feedback, only the reward for the played action can be observed. Therefore, we need to construct a reward estimator that, in expectation, is equivalent to the reward received in the full-information setting:

$$\hat{u}_n^t(a') = m_n^t(a') - \frac{(m_n^t(a') - u_n^t(a'))\mathbf{1}[a_n^t = a']}{p_n^t(a')}. \tag{4.4}$$

A key distinction between our reward estimator and that used in [102] is that each subroutine in our algorithm receives only a portion of $\hat{u}_n^t(a')$. In [102], the entire reward estimator is fed into the algorithm, allowing m_n^t in the algorithm to be canceled out by its counterpart in the reward estimator, which simplifies regret analysis. However, such cancellation does not occur in swap-regret analysis, where an additional term $(p_n^t(a) - p_n^{t-1}(a))m_n^t$ introduces further analytical complexity.

4.5 Analytical Results for OFTRL-LogBar-Bandit

We begin by introducing the notations specific to our regret analysis. Denote by $\|x\|_{q_a^t} := \sqrt{\sum_{a' \in \mathcal{A}_n} (\frac{x(a')}{q_a^t(a')})^2}$ the local norm of vector $x \in \mathbb{R}^{A_n}$ with regard to the log-barrier function with value q_a^t, and by $\|x\|_{*,q_a^t} := \sqrt{\sum_{a' \in \mathcal{A}_n} (x(a')q_a^t(a'))^2}$ the dual local norm. Furthermore, denote by $\mathcal{F}_t$ the σ-algebra formed by the history of all agents' plays and rewards up to the end of round t.

4.5.1 Regret Bounds

The following lemma is one of our key contributions that plays a crucial role in deriving a faster convergence rate for regret bounds, which bounds the gap between the predictor m_n^t compared with the estimated rewards $\hat{u}_n^t$ for all subroutines over T rounds.

Lemma 4.1. For any $n \in [N]$, we have that

$$\begin{aligned}
&\sum_{t=1}^{T} \sum_{a \in \mathcal{A}_n} \left\| p_n^t(a)\hat{u}_n^t - p_n^{t-1}(a)m_n^t \right\|_{*,q_a^t}^2 \\
&\leq 2\sum_{t=1}^{T} \|u_n^t - u_n^{t-1}\|_1 + 2\sum_{t=1}^{T} \|p_n^t - p_n^{t-1}\|_1.
\end{aligned}$$

Proof Sketch. The proof follows two steps, and the details for each step can be found in the Appendix A.3.2.

1. By applying Cauchy-Schwarz inequality and the definition of the dual local norm, we first prove that for any $t \in [T]$ and $n \in [N]$:

$$\sum_{a \in \mathcal{A}_n} \left\| p_n^t(a)\hat{u}_n^t - p_n^{t-1}(a)m_n^t \right\|_{*,q_a^t}^2 \leq 2 \sum_{a \in \mathcal{A}_n} \left| u_n^t(a) - m_n^t(a) \right| \mathbf{1}[a_n^t = a] + 2 \left\| p_n^t - p_n^{t-1} \right\|_1. \tag{4.5}$$

2. The rest is to bound the first term when summed over T rounds. Recall the $m_n^t(a) = u_n^{\tau_t(a)}(a)$ is the last-round reward for arm a before t. We can rewrite it as a telescoping series and apply the triangle inequality to bound it:

$$\begin{aligned}
&\sum_{t=1}^{T} \sum_{a \in \mathcal{A}_n} \left| u_n^t(a) - m_n^t(a) \right| \mathbf{1}[a_n^t = a] \\
&\leq \sum_{t=1}^{T} \sum_{a \in \mathcal{A}_n} \mathbf{1}[a_n^t = a] \sum_{s=\tau_t(a)+1}^{t} |u_n^s(a) - u_n^{s-1}(a)| \\
&= \sum_{t=1}^{T} \sum_{a \in \mathcal{A}_n} |u_n^t(a) - u_n^{t-1}(a)|,
\end{aligned}$$

   where the last equality is due to $\mathbf{1}[a_n^t = a]$ and that $\tau_t(a)$ is the last time before t when a is played.

□

The above lemma converts the gap between estimated rewards and predictors for each subroutine into the gap between actual observed rewards and the difference between mixed strategies in two consecutive rounds. This helps derive the following individual swap regret for each agent.

Theorem 4.2. For $\eta \leq \frac{1}{16}$, the swap regret for each agent $n \in [N]$ is upper bounded as follows.

$$R_n^{\text{swa}}(T) \leq 2 + \mathbf{E}\left[\frac{2(A_n)^2 \ln T}{\eta} + 4\eta \sum_{t=1}^{T} \sum_{m \in [N]} \left\| p_m^t - p_m^{t-1} \right\|_1 - \frac{1}{1024 A_n \eta} \sum_{t=1}^{T} \left\| p_n^t - p_n^{t-1} \right\|_1^2 \right].$$

Proof Sketch. The proof follows the four steps. The details can be found in the Appendix A.3.4.

1. We first convert the regret defined in (4.1) as follows:

$$R_n^{\text{swa}}(T) = \max_{F \in \mathcal{F}_n} \sum_{a \in \mathcal{A}_n} \underbrace{\mathbf{E}\left[\sum_{t=1}^{T} p_n^t(a)\hat{u}_n^t(F(a)) - \left\langle q_a^t, p_n^t(a)\hat{u}_n^t \right\rangle\right]}_{=:R_a^T}.$$

   Such a conversion requires the application of the tower rule on $\sum_{a \in \mathcal{A}_n} p_n^t(a)\hat{u}_n^t$ and the definition of p_n^t in (4.2).

2. Next, notice that R_a^t can be bounded as follows:

$$R_a^T \leq \max_{q \in \Delta(\mathcal{A}_n)} \mathbf{E}\left[\underbrace{\sum_{t=1}^{T} \left\langle q - q_a^t, p_n^t(a)\hat{u}_n^t \right\rangle}_{=:R_a^T(q)}\right].$$

   We want to leverage Lemma 2.9 to bound the RHS of the above equation. However, Lemma 2.9 requires $q \in \text{relint}(\Delta(\mathcal{A}_n))$, while $q \in \Delta(\mathcal{A}_n)$ for $R_a^T(q)$. Thus, we need to find a surrogate point $\tilde{q}_a \in \text{relint}(\Delta(\mathcal{A}_n))$ such that $R_a^T(q) = R_a^T(\tilde{q}_a) + \text{some penalty terms}$.

   Let $q_a^\star = \arg\max_{q \in \Delta(\mathcal{A}_n)} \mathbf{E}[R_a^T(q)]$, and $q_a^c(a') = \frac{1}{A_n}, \forall a' \in \mathcal{A}_n$. Now, we define $\tilde{q}_a = \left(1 - \frac{1}{T}\right) q_a^\star + \frac{1}{T} q_a^c$, and thus $\tilde{q}_a \in \text{relint}(\Delta(\mathcal{A}_n))$. With $\tilde{q}_a$, we have $R_a^T(q_a^\star)$ bounded by

$$\begin{aligned}
\mathbf{E}\left[R_a^T(q_a^*)\right] &= \mathbf{E}\left[\sum_{t=1}^{T} \frac{1}{T}\left\langle q_a^\star - q_a^c, p_n^t(a)\hat{u}_n^t \right\rangle\right] + \mathbf{E}\left[R_a^T(\tilde{q}_a)\right] \\
&\leq \mathbf{E}\left[\sum_{t=1}^{T} \frac{1}{T}\|q_a^\star - q_a^c\|_1 \|p_n^t(a)\hat{u}_n^t\|_\infty\right] + \mathbf{E}\left[R_a^T(\tilde{q}_a)\right] \\
&\leq \mathbf{E}\left[\sum_{t=1}^{T} \frac{2}{T} p_n^t(a)\right] + \mathbf{E}\left[R_a^T(\tilde{q}_a)\right],
\end{aligned}$$

   where the first inequality is due to Holder's inequality, and the last inequality is due to $\|q_a^\star - q_a^c\|_1 \leq \|q_a^\star\|_1 + \|q_a^c\|_1 \leq 2$ and $u_n^t \leq 1$, and the RHS of the above

equation can be bounded by invoking Lemma 2.9 as follows:

$$\mathbf{E}\left[R_a^T(\tilde{q}_a)\right] \leq \mathbf{E}\left[\frac{2A_n \ln T}{\eta} + 2\eta \sum_{t=1}^{T} \left\|p_n^t(a)\hat{u}_n^t - p_n^{t-1}(a)m_n^t\right\|_{*,q_a^t}^2 - \frac{1}{16\eta}\sum_{t=1}^{T}\left\|q_a^t - q_a^{t-1}\right\|_{q_a^{t-1}}^2\right]. \tag{4.6}$$

While leveraging Lemma 2.9, our proof is non-trivial due to several analytical challenges. First, our prediction vector m_n^t differs from those used in full-information settings, necessitating a meticulous analysis to ensure that the conditions for applying Lemma 2.9 still hold with this adapted vector for bandit feedback. Second, Lemma 2.9 provides a standard analysis for OFTRL with self-concordant functions; our primary challenge lies in bounding the last two terms on the right-hand side of (4.6), as discussed in Lemmas 4.1 and A.7. These bounds require careful consideration, particularly since m_n^t lacks an indicator function typically found in reward estimators.

3. The swap regret is bounded by summing over $a \in \mathcal{A}_n$, using Lemma 4.1 to bound the second term on the RHS of the above inequality, and applying Lemma A.7 (see Appendix A.3.3) to bound that $\sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\|q_a^t - q_a^{t-1}\|_{q_a^{t-1}}^2 \geq \frac{1}{64A_n}\sum_{t=1}^{T}\|p_n^t - p_n^{t-1}\|_1^2$. Then, we obtain

$$\sum_{a\in\mathcal{A}_n} R_a^T \leq 2 + \mathbf{E}\left[\frac{2(A_n)^2 \ln T}{\eta} + 4\eta\sum_{t=1}^{T}\|u_n^t - u_n^{t-1}\|_1 + 4\eta\sum_{t=1}^{T}\|p_n^t - p_n^{t-1}\|_1 - \frac{1}{1024A_n\eta}\sum_{t=1}^{T}\left\|p_n^t - p_n^{t-1}\right\|_1^2\right].$$

Theorem 4.2 follows by proving $\|u_n^t - u_n^{t-1}\|_1 \leq \sum_{m\in[N]\setminus n}\|p_m^t - p_m^{t-1}\|_1$, which utilizes the facts that u_n^t is determined by all agents' joint actions, and the total distance between two product distributions is bounded by the sum of the total variations of each marginal distribution [48].

□

We are now prepared to prove our main claim regarding the faster convergence rate. Let $A_{\max} := \max_{n\in[N]} A_n$. By summing the individual swap regret for all agents $n \in [N]$, we arrive at the following corollary.

Corollary 4.3. When $\eta = \frac{1}{32}(\ln T/T)^{\frac{1}{4}} N^{-\frac{1}{2}}$, we have

$$\sum_{n\in[N]} R_n^{\text{swa}}(T) \leq 192 N^{\frac{3}{2}} (A_{\max})^2 T^{\frac{1}{4}} (\ln T)^{\frac{3}{4}} + 2N.$$

Proof. Since $\eta \leq \frac{1}{16}$, invoking Theorem 4.2 gives

$$\sum_{n\in[N]} R_n^{\text{swa}}(T) \leq 2N + \frac{2N(A_{\max})^2 \ln T}{\eta} + \mathbf{E}\left[4\eta N \sum_{t=1}^{T} \sum_{n\in[N]} \left\|p_n^t - p_n^{t-1}\right\|_1 - \frac{1}{1024 A_{\max}\eta} \sum_{n\in[N]} \sum_{t=1}^{T} \left\|p_n^t - p_n^{t-1}\right\|_1^2\right].$$

Let $x_n^t := \|p_n^t - p_n^{t-1}\|_1$, and we obtain the following upper bound on the total swap regret, which takes the form of a quadratic function

$$\begin{aligned}\sum_{n\in[N]} R_n^{\text{swa}}(T) &\leq 2N + \frac{2N(A_{\max})^2 \ln T}{\eta} + \mathbf{E}\left[4096\eta^3 A_{\max} N^3 T - \frac{1}{1024\eta A_{\max}} \sum_{t=1}^{T} \sum_{n\in[N]} \left(x_n^t - 2048\eta^2 A_{\max} N\right)^2\right] \\ &\leq 2N + \frac{2N(A_{\max})^2 \ln T}{\eta} + 4096\eta^3 A_{\max} N^3 T.\end{aligned}$$

The corollary follows by substituting $\eta = \frac{1}{32}(\ln T/T)^{\frac{1}{4}} N^{-\frac{1}{2}}$ into the above inequality. □

We have established that the upper bound of sum of the swap regret for all N agents is $\tilde{O}\left(N^{\frac{3}{2}}(A_{\max})^2 T^{\frac{1}{4}}\right)$, where $\tilde{O}(\cdot)$ hides the logarithmic factors. This implies that the time-averaged swap regret for all agents is $\tilde{O}\left(N^{\frac{3}{2}}(A_{\max})^2 T^{-\frac{3}{4}}\right)$, indicating that the swap regret decays at a rate of $\tilde{O}(T^{\frac{3}{4}})$. This is a significant improvement over previous results for bandit [12, 58, 61, 52] which depend on $O(\sqrt{T})$ for their swap regret bounds, i.e., the time-averaged regret decays at a rate of $O(T^{\frac{1}{2}})$. The intuition behind the improvement lies in leveraging the fact that each agent employs a swap-regret-minimizing algorithm. This behavior, being predictable through the OFTRL framework, facilitates a convergence speed-up.

Since swap regret is always non-negative for each agent $n \in [N]$, Corollary 4.3

also implies the individual swap regret decays at a rate of at least $\tilde{O}(T^{\frac{3}{4}})$. Because bounding swap regret also bounds external regret, this provides a stronger guarantee than the results in [102]. While they demonstrated a faster convergence rate for the sum of external regret, their results do not guarantee the same rate for the individual external regret, as external regret for some agents can be negative [49].

Compared with the results for the full-information setting [6, 5], which demonstrate a $\tilde{O}(T^{-1})$ time-dependence for time-averaged swap regret, smaller than our time-dependence of $\tilde{O}(T^{-\frac{3}{4}})$. Since we consider the expected reward over the opponent's mixed actions, it will be interesting to see whether the convergence rate for the full bandit feedback can be improved.

4.5.2 Time and Space Complexity

The time and space complexities are similar to those in previous works [61, 52] based on the swap-regret-minimizing framework [12]. Note that the optimization problem in (4.3) is a linear optimization with self-concordant functions, which has efficient solutions, making the primary computational complexity stem from calculating the stationary distribution of the Markov chain with A_n states. This stationary distribution can be precisely computed in $O(A_n^2)$ time [40], and approximately computed in almost linear time [27]. Regarding the space complexity, it is $O(A_n^2)$ because each subroutine process needs to maintain meta-distributions and reward vectors for A_n actions. With no communications between agents, there is no communication overhead.

4.6 Experiments

Heterogeneous network selection is a typical application for unknown-game bandits [86, 54]. In this context, each device acts as an agent in the unknown-game bandit model, deciding which heterogeneous network—such as LTE, WiFi, and 5G—to attach to. Therefore, the action set for each agent consists of the available heterogeneous networks, and the reward corresponds to the observed PHY rates of the network to which the agent attaches.

In this section, we adopted experiment settings consistent with those in [54] for both numerical and simulation experiments. Our aim is to compare our OFTPL-based algorithm with two online learning algorithms—Lights and RLNF—studied

in [86, 54], showing our faster regret convergence. Lights [54] is the adaptation of LCE-IX (see Alg. 3.1) to the network selection problem. On the other hand, RLNF is based on a regret minimization approach proposed in [45].

The experiment settings are described as follows.

- Setting 1 is a numerical experiment, focusing on a game between two clients connecting to LTE and WiFi networks. The game is characterized by the reward matrix defined in Table 4.1, which specifies the maximum throughput in Mbps achievable for each client.
- Setting 2 utilizes the Matlab Communications Toolbox™ Wireless Network Simulation Library. This scenario involves 20 clients following a waypoint mobility model within a square area measuring 150 meters by 150 meters. Base stations equipped with the latest technologies, including IEEE 802.11be WiFi and 5G, are strategically deployed throughout the area (refer to Fig. 2 in [54] for further details, which are used for comparison).

Table 4.1: The unnormalized reward matrix for Setting 1

		Client 2	
		LTE	WiFi
Client 1	LTE	(17.5,17.5)	(35,24)
	WiFi	(48,35)	(16,16)

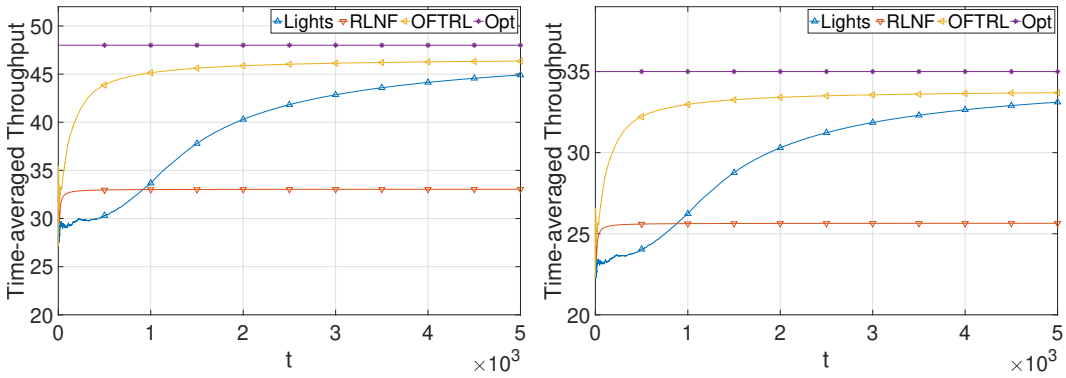
4.6.1 Time-Averaged Throughput

The time-averaged throughput results for Setting 1 are shown in Fig. 4.1 for two clients, while the simulation results for Setting 2 are shown in Fig. 4.2, where our OFTRL-LogBar-Bandit algorithm is denoted as OFTRL.

Fig. 4.2a displays the mean time-averaged throughput across 20 clients, and Fig. 4.2b presents the variability in time-averaged throughput among individual clients using boxplots.

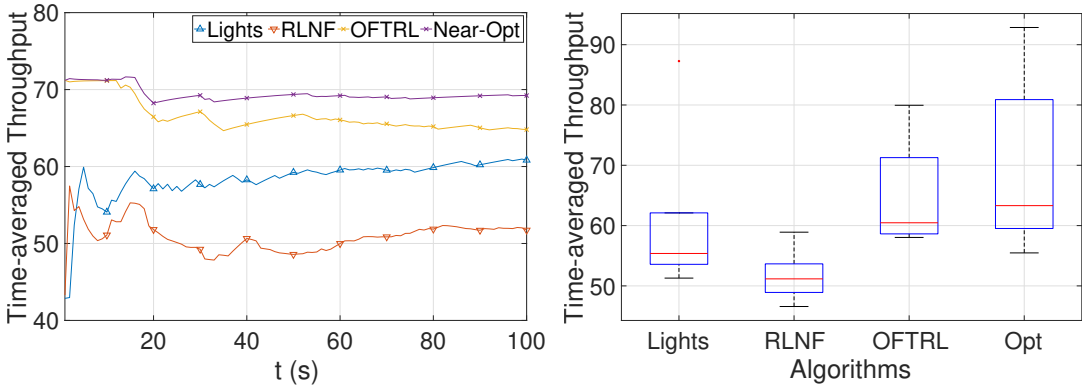
The plot for regret is omitted because time-averaged throughput effectively conveys similar information. A smaller gap from the optimal actions in hindsight (denoted as Opt) corresponds to lower regret incurred by an algorithm.

OFTRL-LogBar-Bandit demonstrates a faster convergence rate compared to other algorithms in both settings, validating the consistency of our analytical results.



(a) Client 1 (b) Client 2

Figure 4.1: The time-averaged throughput (Mbps) for Setting 1.

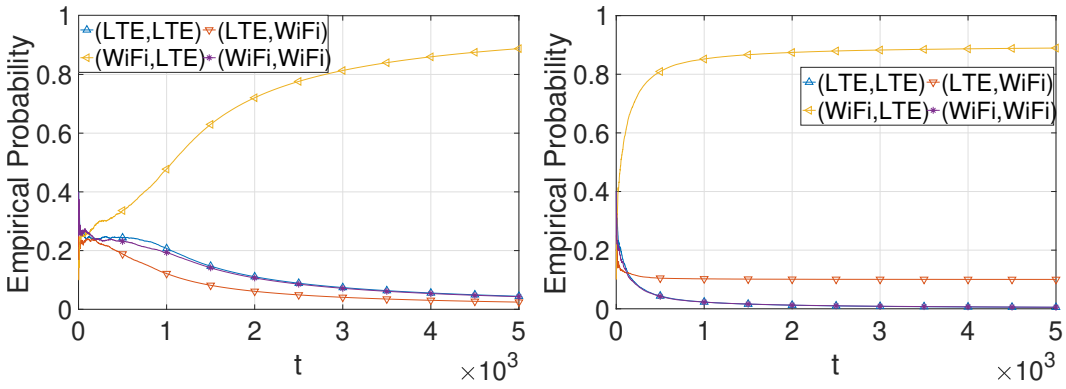


(a) Average Throughput (b) Box Plot

Figure 4.2: The time-averaged throughput (Mbps) for Setting 2.

4.6.2 Convergence to Correlated Equilibrium

Fig. 4.3 demonstrates the convergence of empirical distributions of joint actions for Lights and OFTRL-LogBar-Bandit in Setting 1, where (WiFi, LTE) is the optimal solution when Client 1 attaches to WiFi and Client 2 attaches to LTE. It is evident that both algorithms converge towards a CE, Notably, OFTRL-LogBar-Bandit exhibits a faster convergence rate to the CE, benefiting from a swap regret that is less dependent on the number of rounds T.



(a) Lights (b) OFTRL-LogBar-Bandit

Figure 4.3: The empirical joint distribution over time in Setting 1.

4.7 Conclusion

In this chapter, we demonstrate that the OFTRL-LogBar-Bandit algorithm achieves the time-averaged swap regret of $\tilde{O}(T^{3/4})$, i.e., a decay rate of $\tilde{O}(T^{-3/4})$ for the time-averaged swap-regret, in unknown general-sum games.

Future work will explore whether swap regret with a time dependence comparable to that of full-information feedback can be achieved.

Chapter 5

End-to-End Congestion Control as Learning for Unknown Games with Bandit Feedback

5.1 Introduction

In this chapter, we apply the tools developed in the previous chapters to address the problem of congestion control in computer networks. This problem has remained a vibrant area of research due to the inherent complexity of coordinating distributed flows under limited information. The end-to-end design philosophy of the modern Internet delegates congestion control to end hosts, resulting in a strategic environment where individual flows compete for shared network resources. Analyzing such environments lies at the heart of algorithmic game theory, which provides a principled framework for understanding the dynamics of network congestion and evaluating the performance of congestion control mechanisms [89].

Although many game-theoretic works have been done for congestion control, most of them are focused on analyzing the existing TCP congestion control algorithms or router policies (e.g., drop-tail) [92, 3, 42, 41, 75, 24, 98, 25, 35, 78, 99]. Those works usually assume game models with all information available, e.g., the number of flows, the strategies, and the router policies are known a priori. Such game models work well when designing router policies, as a router has information about incoming flows. However, when it comes to the design of end-to-end congestion control algorithms, such game models fail to capture the reality where

each flow has limited information about others and can only observe the outcome for its chosen *congestion window (cwnd)* or *sending rate (srate)*.

The very first game model for designing end-to-end congestion control algorithms was proposed by Karp et al. [63] in FOCS 2000 where the authors formulated the end-to-end congestion control as a repeated game between a flow and an adversary. In each round of the game, the flow sends a *cwnd* of packets to the network, and the adversary chooses available network bandwidth for the flow but the flow cannot observe it. Then, by the end of the round, the flow will observe a utility determined by the number of sent packets and the available network bandwidth. Such a model is simple yet effective in capturing the interaction between one flow and the network. However, available bandwidth is simply assumed to be dynamically chosen by an adversary, while in reality, the dynamic of available bandwidth is a result of competition among multiple flows. Thus, the author of [63] proposed several open problems, including finding equilibria in a more realistic game model considering the competition among multiple flows, and designing randomized algorithms to deal with the dynamic available bandwidth.

Over the past decades, the above open problems still remain unsolved. Although many works in recent years adopt online learning techniques to design end-to-end congestion control algorithms [71, 60, 36, 1, 105, 37], they are either explicitly or implicitly based on the simple model proposed in [63]. The PCC-Vivace [33] algorithm, on the other hand, implicitly formulates the end-to-end congestion control as a concave game based on the theoretical results in [38]: when minimizing the so-called external regret in concave games for each player, all players will reach Nash equilibria. Albeit concave games capture some game-theoretic essence of the end-to-end congestion control, the assumption about the concave utility function is quite strong. In addition, *external regret* is a performance metric measuring the maximum performance loss between an online learning algorithm and a set of competitors always playing a fixed action, but even the best fixed action may not be the optimal solution for the game. Thus, a more realistic game model and learning algorithms are needed to address the open problems.

We take a step further for the open problems by formulating end-to-end congestion control as learning for repeated unknown general-sum games with bandit feedback. In each round of the unknown general-sum games, or equivalently, black-box games [81], each flow needs to make decisions *independently* in a *distributed* manner with *limited information*. The limited information means that each

flow may not know the number of all the flows in the same network, and each flow cannot observe the information (e.g., the congestion window and packet loss) about other flows, or communicate with other flows. The bandit feedback means that each flow can only know its own utility (e.g., throughput) for its chosen *cwnd* or *srate*. The objective of each flow is to accumulate as many utilities as possible, and all the flows converge to equilibria. Instead of Nash equilibria, we consider a generalization of Nash equilibrium called the correlated equilibrium [62]. The correlated equilibria usually require a central controller to give recommendations to the players involved in the game so that the system can achieve maximum efficiency. However, in the unknown games, we do not assume that each flow can obtain any recommendations. Is there any strategy, if played by all the flows, can achieve the correlated equilibria as if there were a central controller?

This problem is very challenging as flows are affecting each other, and each flow with such limited information needs to trade off between exploring (i.e., probing) the reward for each *cwnd* (or *srate*) and exploiting the current knowledge learned from the exploration to make the best decisions. Motivated by the swap regret [12, 58], a generic performance measure for online learning algorithms, we apply the OMD-LCE-IX algorithm proposed in Chapter 3 for each flow in the unknown games with bandit feedback. OMD-LCE-IX is a no-swap-regret learning algorithm, i.e., the time-averaged swap regret vanishes asymptotically over time. The advantages of minimizing swap regret are twofold. First, a no-swap-regret learning algorithm is robust to a larger set of competitors (see more discussions in Sec. 5.3). Second, minimizing swap regret is a computationally-efficient way to find a correlated equilibrium (see Theorem 2.2). OMD-LCE-IX is designed to be a building block that can be used to design end-to-end congestion control algorithms that address the unknown games and achieve correlated equilibria efficiently.

To sum up, the contributions of our work are as follows:

- We are the first in the literature to formulate the end-to-end congestion control as repeated unknown general-sum games with bandit feedback, which takes a step further to address the open problems raised in [63] by capturing more game-theoretic essence of the end-to-end congestion control in reality.

- We apply the polynomial-time algorithm OMD-LCE-IX proposed in Chapter 3 to address the unknown games, which has a cumulative swap regret upper bounded by $O(A_n\sqrt{T\log(A_n\delta^{-1})})$ with probability at least $1-\delta$ for any $\delta \in$

$(0, 1)$, where A_n is the number of actions for flow n and T is the total length of the game. Furthermore, the OMD-LCE-IX algorithm can achieve an ϵ-correlated equilibrium in a polynomial number of rounds.

- Third, we implement OMD-LCE-IX through the Linux kernel 5.13.12 based on the congestion control plane [80], a new API for writing congestion control algorithms. We first perform TCP fairness-related experiments in Mininet to compare with the TCP CUBIC and TCP BBR version 2. Then we perform experiments driven by U.S. cellular network traces with Pantheon [107] with an additional comparison to PCC-Vivace [33]. The experiment results show that OMD-LCE-IX is TCP-friendly and competitive, and can adapt to dynamic network environments.

The rest of the chapter is organized as follows. Sec. 5.2 reviews related works. The problem settings are described in Sec. 5.3. We show the throughput and fairness-related experiments in Sec. 5.5. Sec. 5.6 concludes the chapter.

5.2 Related Works

In this section, we first give a detailed review of the game-theoretic congestion control, and then briefly discuss recent progress in learning-based congestion control. Furthermore, we will review equilibrium learning in game theory.

Game-theoretic Congestion Control: Game theory has been extensively studied in congestion control, and there are mainly two lines of research for game-theoretic congestion control. One is focused on router-based congestion control, which manages the incoming packets from different flows for a router, and the other studies the end-to-end congestion control, which decides how many packets to be sent at a time for each flow.

For router-based congestion control, the main goal is to analyze the existing TCP congestion control algorithms with given router policies (e.g., drop-tail) or design new router policies. The earliest work can be traced back to [79], which gave game-theoretic implications of switching disciplines and their relevance to congestion control. It was later followed by the works of [92, 3, 42, 41, 75, 24, 98, 25, 35], where the independent data flows are considered as selfish players in a game, and the mechanism of the game is determined by the router policies. In such games, both the router policies and the end-to-end congestion control algorithms

(i.e., the strategies of players) are known a priori. Although the above works can be effective in designing and analyzing router policies, they cannot provide an end-to-end congestion control solution for data flows.

The other line of research studies the design and analysis of end-to-end congestion control algorithms from a game-theoretic point of view. One of the earliest works is [63], where the author modeled the congestion control problem as a game between a flow and an adversary. In each round, a flow selects an action (e.g., *cwnd*) and the adversary selects a bandwidth for triggering penalties if congestion happens or there is wasted bandwidth. By the end of that paper, they raised several open questions, including how to model a more realistic case where the available bandwidth is a result of the competition among flows instead of being chosen by the adversary, and whether equilibria exist in such scenarios. Such a multi-flow game problem is challenging, as each flow has very limited information about other flows in the end-to-end congestion control.

Later, in the work of [4], the author modeled the end-to-end congestion control as a noncooperative game, and proved the existence and uniqueness of Nash equilibrium with convexity assumptions for utility functions. They also design gradient algorithms to achieve the Nash equilibrium. However, the gradient algorithm requires knowledge about the total number of flows and the network capacity, which is not practical in reality. To address this issue, the authors of [7] tried to model the end-to-end congestion control as a Bayesian game. Although each flow does not need to know the exact information about other flows, a prior belief about others is still required.

Some theoretical progress has been made in the work of [33], where the authors designed PCC-Vivace based on the theoretical work [38] for equilibrium learning in concave games, i.e., the utility function for each flow is concave, where the Nash equilibrium can be reached if each player plays an external-regret-minimizing algorithm. However, it is not realistic to assume the utility for each flow must be a concave function. On the other hand, unknown general-sum games can well capture the game-theoretic nature of end-to-end congestion control, as there are no unrealistic assumptions for either the flows or the utility functions.

To the best of our knowledge, the open problem proposed by [63] has still remained unsolved. We take a step further by modeling the competition of multiple flows as a repeated unknown general-sum game with bandit feedback for the first time in the literature and proposing a swap-regret-minimizing algorithm to

asymptotically achieve the correlated equilibria that are more general than the well-known Nash equilibria.

Learning-based Congestion Control: Recent years have witnessed a line of research on congestion control based on machine learning techniques. Remy [103] and Indigo [107] are two representative works for offline-learning congestion control algorithms. Such algorithms have limited adaptivity to new situations in practice. Therefore, *reinforcement learning (RL)* techniques have been introduced to congestion control to alleviate such problems, such as QTCP [71], Aurora [60], Eagle [36], Orca [1], MOCC [105] and Pareto [37]. However, a certain amount of offline training is often needed for the above RL models to guarantee an efficient and effective deployment. Lightweight online learning techniques do not require a pre-trained model. The typical example is PCC-Vivace [33], which relies on online (convex) optimization to update the sending rate. Although the above works share some similarities to the equilibrium learning in our work, the common limitation of the above learning-based algorithms is that they can only minimize external regret, i.e., the maximum performance gap from the set of competitors always playing a fixed action is bounded. There are no theoretical guarantees for the convergence to correlated equilibria.

Equilibrium Learning: The study of unknown game models (or the black-box games [81]) has a long history that can be traced back to the fictitious play for the two-player zero-sum games [14, 91]. However, it was not until the start of this century that much progress has been made with the development of online learning techniques [20], particularly for games with specific structures. For example, the authors of [29] studied the congestion game with bandit feedback, where the authors tried to minimize a Nash regret, which is the sum of the maximal external regret among players in each round. A similar work of [13] studied a specific congestion game, where each resource is equally shared among the players who choose it. Nevertheless, end-to-end congestion control is not necessarily a congestion game, as one may not find a potential function that is the essence of the congestion game. The authors of [97] studied augmented games by utilizing communications between players. However, such a methodology is not suitable for unknown games, as the players in unknown games do not know each other and will not communicate with each other. There are many other equilibrium learning works for some specific games, e.g., potential games [28, 26, 11, 76], and mean-field games [77, 101, 106]. As all their results require specific game structures,

they cannot be applied to unknown general-sum games.

Regarding the learning for unknown general-sum games, there are mainly two situations depending on the observability of feedback. If the utility of an action can be observed regardless of whether it is played or not, we call it the *full-information feedback* [88, 66, 23], and if only the utility of a played action can be observed, then it is the *bandit feedback*. As in end-to-end congestion control, each flow can only observe the feedback for its selected *cwnd* (or *srate*), we will focus on learning for the bandit feedback.

The first work that addressed the unknown general-sum game problem with bandit feedback is [9], where an exponential-weight technique is proposed to minimize external regret. However, it is shown in [20] that the external-regret-minimizing algorithm can only converge to the set of Nash equilibria for the two-person zero-sum game. It was later proved in [38] that minimizing external regret can converge to Nash equilibria for concave games. However, for end-to-end congestion control, we cannot take for granted that the utility function is concave. As we want to come up with a building block that can be adapted to any congestion control algorithm, we are more interested in unknown general-sum games, and the correlated equilibrium can only be achieved if the internal regret can be minimized asymptotically. Since minimizing the swap regret can also minimize both the external and internal regret [12] and be more robust against a larger set of competitors, we are motivated to address the unknown games for end-to-end congestion control from the swap-regret viewpoint, which is different from the Nash regret that is prevalent in the equilibrium-learning literature.

5.3 Model and Problem Formulation

5.3.1 Unknown General-Sum Game Model with Bandit Feedback

We consider a network of N flows competing for the same resource (e.g., bandwidth) in the network, as shown in Fig. 5.1. The congestion happens when the number of packets sent by all flows is beyond the network capacity, and the overflowed packets will be dropped according to a router policy (e.g., drop-tail). The competitive interaction between multiple flows can be modeled by a repeated un-

known general-sum game involving N flows with the following two justifications that are also widely used in the related works [35, 3].

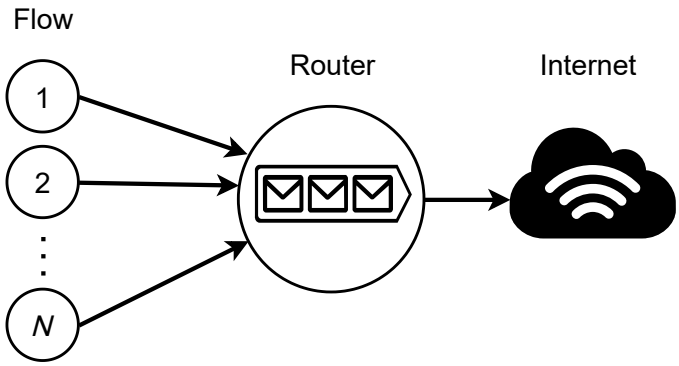


Figure 5.1: A network model for the unknown general-sum games.

First, the length of each round of the repeated game is chosen appropriately such that all the flows can finish one round of interaction in one round of the game, i.e., send a *cwnd* of packets and receive ACKs for these packets. Second, we assume the packet loss is only caused by the congestion, as the congestion control scheme in the current TCP also assumes the packet loss is congestion-induced.

Therefore, we can formally define the repeated unknown general-sum game with bandit feedback for end-to-end congestion control as follows. Denote by $[N] := \{1, \ldots, N\}$ the set of all flows in the game, and each flow $n \in [N]$ is associated with a (possibly different) finite set of actions (i.e., *cwnd* or *srate*) $W_n := \{w_{\min}, \ldots, w_{\max}\}$, where $w_{\max}$ is the maximum *cwnd* (or *srate*) of flow n. Let $A_n := |W_n|$ be the total number of actions for flow n. In an actual implementation, $w_{\min}$ and $w_{\max}$ can be either determined by an initial probing phase, or determined by the network physical-layer capabilities. The game is repeated for T rounds. In each round $t = 1, \ldots, T$, each flow selects a *cwnd* (or *srate*) $w_n^t \in W_n$. By the end of round t, each flow observes feedback information (e.g., packet loss and round-trip time) and calculates utility $u_n^t \in [0, 1]$. We only require that the calculated utility should be normalized between 0 and 1, and do not give any specific form of the utility function here because we want our solution to be general enough for any utility functions, so that the solution to the unknown games can be acting as a building block for other congestion control algorithms.

For any given network capacity and router policies, the utility for each flow n is not only dependent on its own action w_n^t but also determined by the actions of all

other flows. Denote by $W^t := \{w_n^t : \forall n \in [N]\}$ the *action profile* in round t. To emphasize the dependency of the utility on all the flows, we further write u_n^t as $u_n(W^t)$ or $u_n(w_n^t, w_{-n}^t)$, where $(w_n^t; w_{-n}^t)$ is an abbreviation of $W^t := (w_1^t, \ldots, w_n^t, \ldots, w_N^t)$ with a highlight of flow n's action w_n^t against other flows' actions.

We do not directly model the flow importance (e.g., the QoS requirements) in the game, as the importance of the flows is taken into consideration by router policies. For example, one can design a router policy that drops fewer packets for the flows with a higher QoS requirement. As the focus of this chapter is the design of a solution for the unknown game with any router policies, our solution still works even if the flows are of different importance.

Note that each flow is in a *bandit feedback setting*, i.e., neither the actions nor the loss of other flows can be observed, and each flow n can only observe the information, such as packet loss and round-trip time, to calculate its own utility for the chosen *cwnd* (or *srate*). Also, neither the number of flows nor the router policy is known a priori to each flow. The reason for considering such a limited information setting is to make the model more realistic so that our algorithm is more deployable to the end systems without modifying the intermediate nodes.

5.3.2 Problem Formulation

The goal of each flow in the unknown general-sum games with bandit feedback is to accumulate as many utilities as possible without getting the network congested. Network congestion results in packet loss and queuing delay, which further reduces the utilities for each flow. Such a goal can be easily achieved if a router can act as a central controller to allocate *cwnd* (or *srate*) for each flow by sending control messages. For example, if all the N flows are of the same importance, i.e., all the N flows have the same QoS requirements, then the optimal *srate* for each flow is C/N, where C is the network capacity. To accommodate more general situations where the importance of flows can be different or time-varying, we use the notion of ϵ-*correlated equilibrium* to measure the optimality of a solution (see Definition 2.3).

Intuitively, let $\mathbf{P}$ be a joint distribution for all players' actions, and the router draws an action profile from $\mathbf{P}$ and privately recommends the *srate* to each flow. For example, in the case of flows with equal importance, $\mathbf{P}(w_n = \frac{C}{N}, \forall n \in [N]) = 1$ and the probabilities for drawing other profiles are equal to 0. As no flow will gain more than ϵ to choose a different *srate*, provided that other flows follow the

router's recommendation, such a **P** is an ϵ-correlated equilibrium. Compared with other forms of equilibria, correlated equilibrium considers the joint instead of the marginal distribution of the action space and does not assume independence among different action sets. Thus, correlated equilibrium is more general and useful.

However, the goal of each flow to accumulate maximum utilities becomes more challenging when each flow in the unknown games makes decisions independently. The problem of learning for unknown general-sum games is stated as follows: When the only information revealed to each flow is the utility of its chosen *cwnd* (or *srate*) in each round, is there any algorithm that can help each flow accumulate more utilities and converge to the ϵ-correlated equilibrium as if there were a central controller?

Thanks to Theorems 2.2 and 3.6, if all agents employ a swap-regret-minimizing algorithm, the system is guaranteed to converge to an ϵ-correlated equilibrium. Thus, our goal is to minimize the swap regret for each flow n, with respect to a class of swap functions $\mathcal{F}_n$, up to round T:

$$R_n^{\text{swa}}(T, \mathcal{F}_n) = \max_{F \in \mathcal{F}_n} \sum_{t=1}^{T} \sum_{w \in W_n} \mathbf{1}[w_n^t = w] \left(u_n(F(w); w_{-n}^t) - u_n(w; w_{-n}^t) \right), \tag{5.1}$$

where w_n^t denotes the action taken by player n at time t, and w_{-n}^t represents the joint actions of all other players at round t. This formulation captures the maximum regret incurred by not switching actions according to any swap function in $\mathcal{F}_n$.

Therefore, we will adapt the OMD-LCE-IX algorithm proposed in Chapter 3 to address the end-to-end congestion control problem. Key notations for this section are summarized in Table 5.1.

5.4 Adaptation of OMD-LCE-IX

In this section, we apply the framework of game-theoretic bandits to the end-to-end congestion control problem, where each player corresponds to a data flow, and the action set consists of possible congestion window sizes or sending rates. We define a round for each flow as a *round-trip time (RTT)*—the duration required to send a batch of packets and receive their acknowledgments.

Importantly, our formulation does not assume synchronized RTTs across flows.

Table 5.1: Summary of key notations

Notations	Definition
N; $[N]$	The number of flows; the set of all the flows
W_n	The action set for flow n
A_n	The number of actions for flow n
$W; W^t$	An action profile; an action profile in round t
$u_n(W^t)$	The utility for flow n in round t given action profile W^t
t; T	The round of time; the total number of rounds
w_n^t	The *cwnd* selected by flow n in round t
w_{-n}^t	The *cwnd* selected by flows other than n in round t
$R_n^{\text{swa}}(T, \mathcal{F}_n)$	The swap regret for flow n up to round T

Since each flow independently runs its own swap-regret-minimizing algorithm, heterogeneous RTTs do not interfere with the regret minimization process. While having uniform RTTs across players can facilitate faster convergence to correlated equilibrium, differing RTTs do not hinder the algorithm's ability to minimize the performance gap relative to the best swap-based competitor.

Additionally, we assume that packet loss is solely due to congestion, mirroring the assumption made by the congestion control mechanism (e.g., CUBIC) in current TCP implementations, which attributes packet loss primarily to congestion.

Then, to adapt OMD-LCE-IX to the TCP congestion control setting, the key design choices lie in defining the action set and the reward function in a way that is both theoretically principled and practically realizable within the TCP framework.

5.4.1 Action Set

The action set for each flow consists of a discrete set of permissible sending rates or congestion window sizes. Inspired by Sprout [104], we discretize the feasible range of congestion window values into a finite set W_n for each flow n, capturing realistic constraints imposed by both the transport protocol and the underlying network. Each action $w \in W_n$ represents a candidate window size that the flow may select during an RTT. The choice of discretization granularity involves a trade-off between decision precision and computational overhead.

In practice, we employ a startup probing phase to initialize the sending rate.

During the early stage of each connection (or recovery period), the sending rate is increased exponentially, similar to the startup behavior of TCP and BBR. Rather than relying solely on packet loss to terminate the probing phase, we monitor whether the estimated throughput continues to grow. If the throughput fails to increase beyond a predefined threshold for several consecutive measurements—or a packet loss is observed—the startup phase terminates. At this point, we define the action set W_n by discretizing the interval between the current sending rate and half of that rate. This adaptive construction of the action space captures the local network condition while maintaining a compact and expressive decision set for efficient online learning.

5.4.2 Reward Function

Each TCP flow can only observe limited local information, such as its own throughput, RTT, and packet loss rate L_n^t. Inspired by PCC-Vivace [33], we design the reward function to balance throughput with delay and loss sensitivity. The reward function for t-th RTT round is defined as:

$$u(w_n^t) := w_n^t \text{RTT}^t - b \cdot w_n^t \cdot (\text{RTT}^t - \text{RTT}^{t-1}) - c \cdot w_n^t \cdot L_n^t,$$

where w_n^t denotes the sending rate (or congestion window size) of flow n at time t, RTT^t is the measured RTT, and L_n^t is the observed packet loss rate. The constants $b > 0$ and $c > 0$ are tunable parameters that penalize delay growth and packet loss, respectively.

Intuitively, in the absence of queue buildup and packet loss, the number of packets sent should closely match the sending rate $w_n^t \text{RTT}^t$. However, when the sending rate exceeds the network's available capacity, packets begin to queue, leading to increased RTT. The term $w_n^t(\text{RTT}^t - \text{RTT}^{t-1})$ approximates the growth in the queueing delay, capturing congestion buildup. Similarly, $w_n^t L_n^t$ estimates the number of packets lost due to congestion.

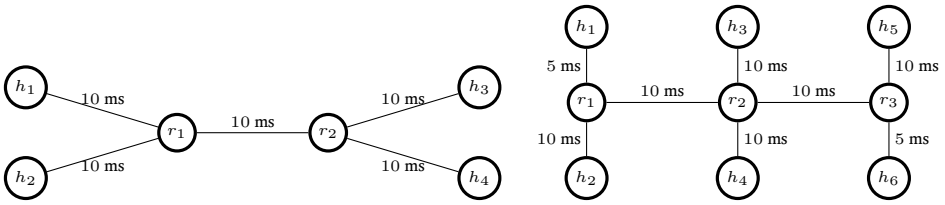
This reward formulation encourages flows to increase throughput while penalizing rising latency and losses, aligning with real-world objectives for efficient and stable network performance.

With the carefully designed action set and reward function, we are able to apply the OMD-LCE-IX algorithm in a fully decentralized and feedback-driven manner

to the congestion control problem. This integration enables each flow to independently learn optimal sending behaviors over time, minimizing swap regret and steering the overall system toward correlated equilibria with improved fairness and efficiency.

5.5 Emulation Experiments

We have implemented the proposed OMD-LCE-IX algorithm through the Linux Kernel 5.13.12 based on the congestion control plane [80], a new API for writing congestion control algorithms. In this section, we start with fairness-related experiments in Mininet [68], where OMD-LCE-IX is compared with CUBIC [43] and BBR version 2 [19] (BBR2 for short in the following). Then, we conduct trace-driven experiments with Pantheon [107], an evaluation platform for congestion control algorithms, with an additional comparison to PCC-Vivace [33]. We have released the source code for the experiments in `https://github.com/Zhiming-Huang/OMD-LCE-IX`.



(a) The Dumbbell Topology (b) The Parking Lot Topology

Figure 5.2: The experiment topology.

In the fairness-related experiments, two classic network topologies recommended by the IETF TCP evaluation suite [46] are considered, i.e., the dumbbell and parking lot topologies, as shown in Fig. 5.2. The bandwidth of all the links in both topologies is 50 Mbps, and the delay of each link is shown in the figures. For both topologies, the queue size on all the links between the two routers is 100 packets. In the dumbbell topology, there are two flows from h_1 to h_3 and from h_2 to h_4, sharing the same link r_1-r_2. In the parking lot topology, there are three flows from h_1 to h_6, from h_2 to h_3, and from h_4 to h_5. We can see that the flow from h_1 to h_6 competes with both the other two flows, while the other two flows are independent of each other. In the experiments, we use `iperf` to generate a 30s test

for the performance of the three congestion control algorithms. As `iperf` outputs the averaged results (i.e., throughput and RTT) every 1s, the points at time 0s in Figs. 5.3 to 5.5 are the averaged results in the initial interval from 0s to 1s, and then we use the exponentially weighted moving average technique to smooth the results in the following time.

In the trace-driven experiments, we use the US cellular network traces (i.e., T-Mobile and Verizon) recorded by the saturator tool [104] while driving. These traces represent the time-varying capacity of the networks experienced by a mobile user, so we can test the adaptability of congestion control algorithms in such network environments.

5.5.1 Dumbbell Results

We first test the scenario where the two flows are homogeneous, i.e., both the flows adopt the same congestion control algorithm. The throughput and RTT results for the homogeneous flows are shown in Figs. 5.3a, 5.3b, 5.3f, and 5.3g. As observed, when all flows implement OMD-LCE-IX, they tend to achieve a fair performance (i.e., similar throughput). This is because OMD-LCE-IX ensures convergence to a CE when adopted by all flows, simulating the effect of a central controller that distributes bandwidth evenly among all participating flows. However, the other two congestion control algorithms, i.e., CUBIC and BBR2, do not guarantee an equal share of resources between the flows. This is due to the intrinsic property of these algorithms (i.e., deterministic strategy) and the slight difference in flow start times, although we made efforts to minimize this difference in the experiments by using a script to control all nodes. On the other hand, we can observe that OMD-LCE-IX performs better than BBR2 and CUBIC in the homogeneous setting in terms of throughput, while having a higher RTT. This is because OMD-LCE-IX does not explicitly incorporate queuing models like BBR2 does, and its randomized action selection makes it less conservative in utilizing network buffers, leading to increased queue lengths.

We also conduct experiments for the heterogeneous flow, i.e., the two flows adopt different congestion control algorithms, as shown in Figs. 5.3c to 5.3e and Figs. 5.3h to 5.3j. When BBR2 and OMD-LCE-IX compete with each other, BBR2 prevails at first, but their gap gradually decreases due to the benefits of learning. On the other hand, OMD-LCE-IX can achieve a similar throughput to CUBIC.

Regarding RTT, we can observe in Fig. 5.3h that in the first few intervals, both OMD-LCE-IX and BBR2 suffer a high RTT because they are competing with each other to exhaust the network bandwidth. Overall, the RTTs for both algorithms are decreasing over time, meaning that the network is getting less congested. Therefore, OMD-LCE-IX is friendly to and competitive with other TCP flows.

5.5.2 Parking Lot Results

The results of the parking lot topology are shown in Figs. 5.4 and 5.5. Similar to the dumbbell topology, we first test the homogeneous flows in the parking lot topology, as shown in Figs. 5.4a to 5.4c and Figs. 5.5a to 5.5c. The flow from h_1 to h_6 will suffer a loss if any one of the other two flows suffers, i.e., the flow from h_1 to h_6 has a higher probability of suffering a loss and thus results in a lower throughput than the other two flows. As we can see, when all three flows play the same congestion control algorithm, OMD-LCE-IX always guarantees that the performance gaps between flows are similar and not excessive. On the other hand, CUBIC and BBR2 have a large performance gap between flows h_1 to h_6 and the other two flows. This reflects OMD-LCE-IX's ability to achieve a stable CE. Also, CUBIC and BBR2 incur a higher RTT than OMD-LCE-IX, because CUBIC and BBR2 will increase the *srate* to probe for possible higher bandwidth, while OMD-LCE-IX can maintain a low RTT.

Then, we perform three different heterogeneous flow settings in the parking lot topology for a different 4-hop flow (i.e., flow h_1 to h_6). In the first heterogeneous flow setting, flow h_1 to h_6 adopts OMD-LCE-IX, flow h_2 to h_3 adopts CUBIC, and flow h_4 to h_5 adopts BBR2, and the results are shown in Figs. 5.4d and 5.5d. In the second setting, the 4-hop flow h_1 to h_6 adopts CUBIC, flow h_2 to h_3 adopts BBR2 and flow h_4 to h_5 adopts OMD-LCE-IX, as shown in Figs. 5.4e and 5.5e. In the third setting, the 4-hop flow h_1 to h_6 would be BBR2, and the other two flows adopt CUBIC and OMD-LCE-IX, respectively, as shown in Figs. 5.4f and 5.5f. We can see that when the 4-hop flow adopts OMD-LCE-IX or CUBIC, it will concede the link bandwidth to the other two flows, but OMD-LCE-IX still maintains a higher throughput than CUBIC, as shown in Figs. 5.4d and 5.4e. However, when the 4-hop flow adopts BBR2, it occupies a comparable link bandwidth with the other two flows, as shown in Fig. 5.4f. Overall, from the above experiments, we can see that OMD-LCE-IX is friendly but competitive to other flows, and maintains fairness

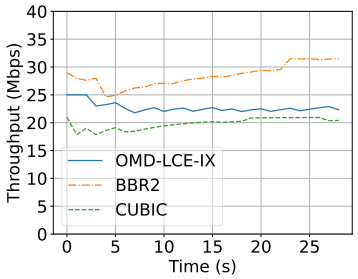
in allocating the link bandwidth.

5.5.3 Trace-Driven Experiments

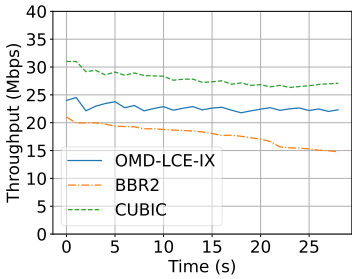
For each trace, we did ten independent runs of experiments, and the mean results for the trace-driven experiments produced by Pantheon are shown in Fig. 5.6. Pantheon evaluates an algorithm based on two metrics, i.e., mean throughput and 95th-percentile one-way delay. In both LTE networks, CUBIC achieves the highest throughput, albeit at the cost of the highest delay. OMD-LCE-IX, on the other hand, offers a throughput that is comparable to CUBIC but with a lower delay. Conversely, BBR2 achieves the lowest delay but sacrifices the throughput, which is lower than that of OMD-LCE-IX. Overall, we can see that OMD-LCE-IX can better balance the tradeoff between the average throughput and delay in these two metrics for the T-Mobile network than the other three algorithms. Thus, OMD-LCE-IX can guarantee good performance in a dynamic network environment.

5.6 Conclusion

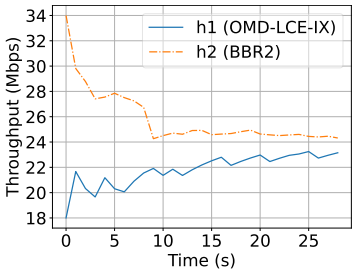
In this chapter, we formulated the end-to-end congestion control as a repeated unknown general-sum game with bandit feedback, and proposed the OMD-LCE-IX algorithm with provable theoretical guarantees. Furthermore, we have implemented OMD-LCE-IX through the Linux kernel and performed extensive experiments to verify the performance of OMD-LCE-IX. For our future research, we would like to develop more realistic game models where we relax the assumption that all flows finish an interaction within one round, and study whether an equilibrium for such game models exists and can be obtained by efficient learning algorithms. We are also interested in using OMD-LCE-IX as a building block to improve the current congestion control algorithms, such as BBR2.



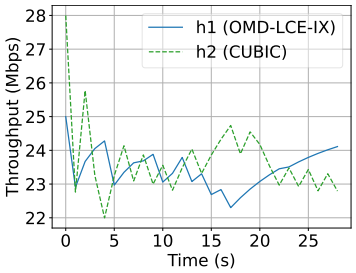
(a) Homo-Flow Throughput on h_1



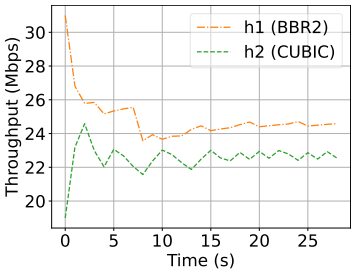
(b) Homo-Flow Throughput on h_2



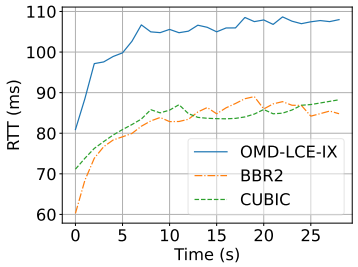
(c) Throughput (OMD-LCE-IX, BBR2)



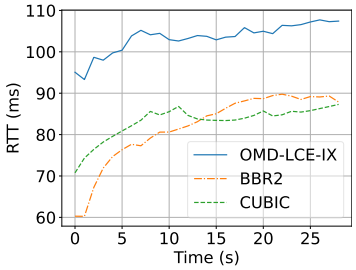
(d) Throughput (OMD-LCE-IX, CUBIC)



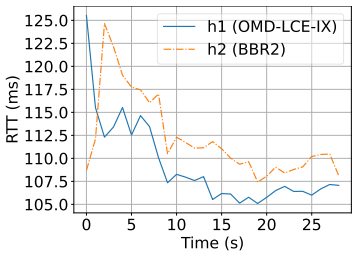
(e) Throughput (BBR2, CUBIC)



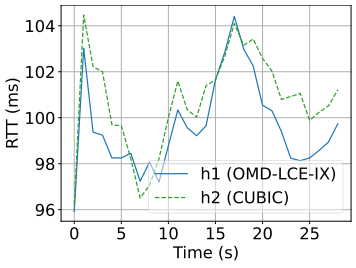
(f) Homo-Flow RTT on h_1



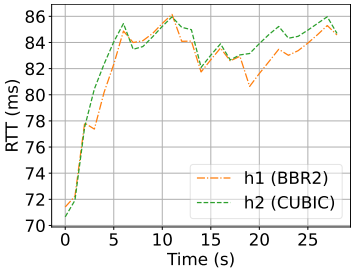
(g) Homo-Flow RTT on h_2



(h) RTT (OMD-LCE-IX, BBR2)

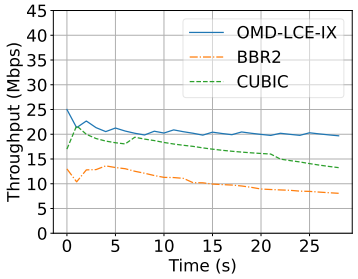


(i) RTT (OMD-LCE-IX, CUBIC)

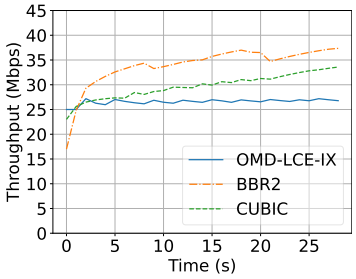


(j) RTT (BBR2, CUBIC)

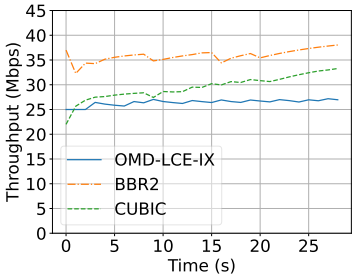
Figure 5.3: The experiment results for the dumbbell topology.



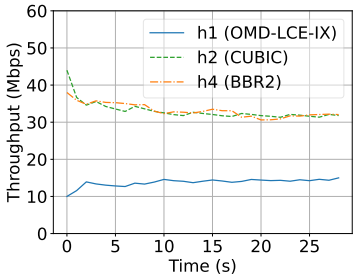
(a) Homo-Flow Throughput on h_1



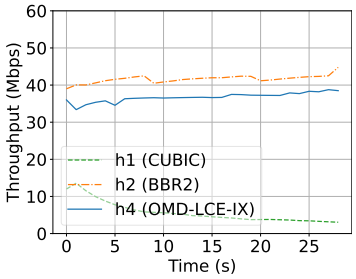
(b) Homo-Flow Throughput on h_2



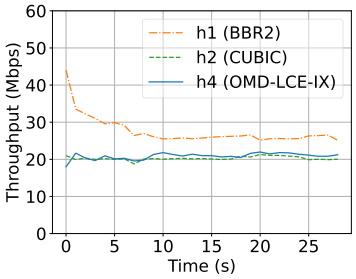
(c) Homo-Flow Throughput on h_4



(d) Throughput for Hete-Flow Setting 1

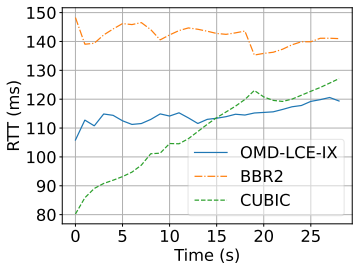


(e) Throughput for Hete-Flow Setting 2

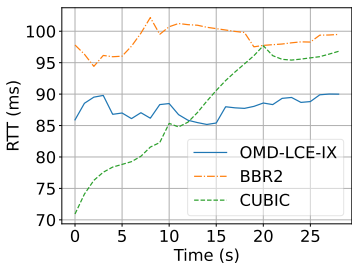


(f) Throughput for Hete-Flow Setting 3

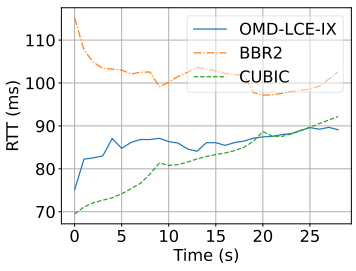
Figure 5.4: The throughput results for the parking lot topology.



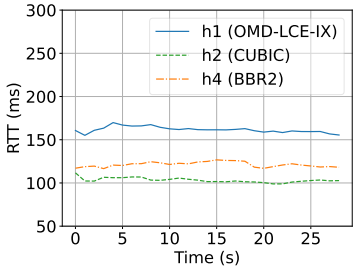
(a) Homo-Flow RTT on h_1



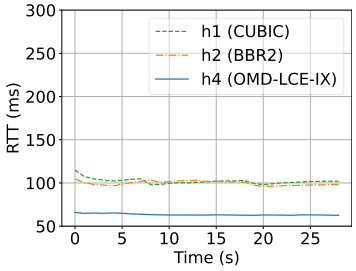
(b) Homo-Flow RTT on h_2



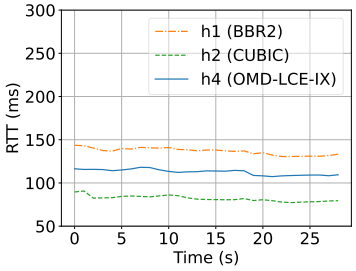
(c) Homo-Flow RTT on h_4



(d) RTT for Hete-Flow Setting 1

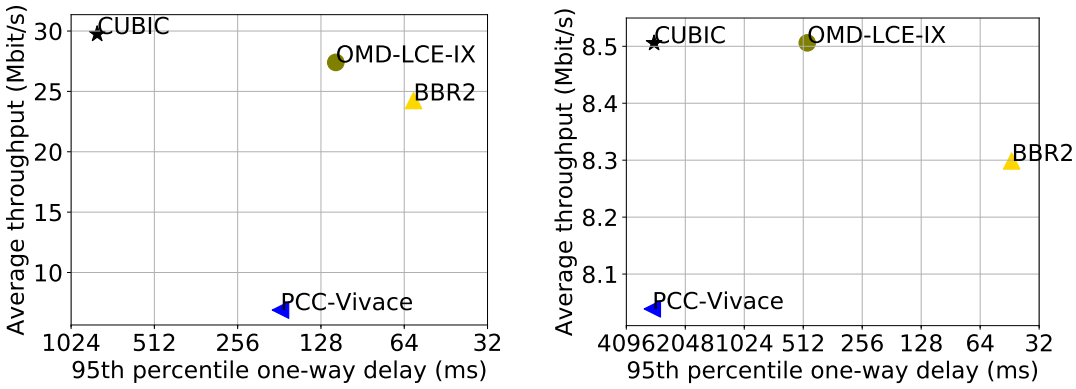


(e) RTT for Hete-Flow Setting 2



(f) RTT for Hete-Flow Setting 3

Figure 5.5: The RTT results for the parking lot topology.



(a) T-Mobile LTE Network

(b) Verizon LTE Network

Figure 5.6: The trace-driven experiment results on Pantheon.

Chapter 6

Conclusions

This dissertation has presented a comprehensive study of game-theoretic bandit learning for distributed network optimization, focusing on the intersection of online learning, equilibrium computation, and real-world system design. Motivated by challenges in congestion control and other networked systems, we modeled decentralized decision-making as repeated games with bandit feedback, where agents seek to optimize their individual performance while interacting strategically under uncertainty.

Our contributions span both theory and practice. On the theoretical side, we developed a new algorithmic framework that achieves high-probability swap regret bounds under bandit feedback, enabling convergence to correlated equilibria in decentralized multi-player settings. We further enhanced these results by incorporating optimistic updates, significantly accelerating convergence without compromising robustness. These advances deepen our understanding of how learning dynamics can drive systems toward equilibrium in the absence of centralized control.

On the practical side, we applied these theoretical insights to TCP congestion control, implementing our algorithms in the Linux kernel and demonstrating improved throughput and fairness in trace-driven network emulations. This serves as a concrete example of how abstract learning-theoretic models can inform and enhance real-world protocol design.

Overall, this dissertation bridges the gap between theoretical guarantees and system-level implementations, contributing to the broader effort of designing scalable, adaptive, and principled solutions for decentralized optimization in natural and engineered networked systems. In such systems, stability, efficiency, and fair-

ness are not imposed from above but arise through the adaptive behavior of individuals navigating strategic environments. The techniques developed here offer one approach to making this emergence more predictable, robust, and efficient.

Several promising directions emerge from this work. First, extending the swap-regret framework to more complex feedback models—such as delayed, structured, or graph-based feedback—could significantly broaden its applicability in practical settings. This is particularly relevant for congestion control, where feedback is inherently delayed due to round-trip times.

Second, examining the tightness of existing high-probability regret bounds under bandit feedback remains an important theoretical challenge, with potential implications for both learning efficiency and equilibrium quality. Another natural extension is to consider settings with infinite or continuous action spaces, which frequently arise in real-world applications—for instance, the action set in congestion control can be extremely large or even continuous.

Finally, a key systems-oriented direction is to more deeply integrate learning-based protocols into production-level networking stacks. This opens the door to self-optimizing infrastructure, where practical systems evolve through continual adaptation driven by online learning algorithms.

As we look ahead, the need for such adaptive and principled designs is only growing. In an increasingly connected world, where systems are large, complex, and decentralized by nature, the ability to harness local learning to drive global coordination is both a theoretical challenge and a practical imperative. This dissertation takes a step in that direction—toward understanding, shaping, and engineering systems where equilibrium is not dictated, but learned.

Appendix A

Proof Details

A.1 Proofs for Chapter 2

A.1.1 Useful Facts

Definition A.1 (Fenchel Conjugate)**.** For a function $\psi : \mathbb{R}^d \to [-\infty, \infty]$, we define the Fenchel conjugate $\psi^\star : \mathbb{R}^d \to [-\infty, \infty]$ as

$$\psi^\star(\theta) = \sup_{p \in \mathbb{R}^d} \langle \theta, p \rangle - \psi(p)$$

The definition of Fenchel conjugate directly leads to Fenchel-Young's inequality:

$$\langle \theta, p \rangle \leq \psi^\star(\theta) + \psi(p), \forall \theta, p \in \mathbb{R}^d. \tag{A.1}$$

The following example will be used in our analysis for the Fenchel conjugate of quadratic norms.

Example A.1 (Fenchel Conjugate of Quadratic Norms)**.** Let $A \in \mathbb{R}^{d \times d}$ be a positive definite matrix, and we define the quadratic norm for any $p \in \mathbb{R}^d$ as $\|p\|_A^2 := p^\top A p$. The Fenchel conjugate of $\frac{1}{2}\|p\|_A^2$ is thus:

$$\langle \theta, p \rangle - p^\top A p \leq \frac{1}{2} \theta^\top A^{-1} \theta = \frac{1}{2} \|\theta\|_{A^{-1}}^2.$$

Furthermore, according to Fenchel-Young's inequality in (A.1), we have that

$$\langle \theta, p \rangle \leq \frac{1}{2} \|\theta\|_{A^{-1}}^2 + \frac{1}{2} \|p\|_A^2, \forall \theta, p \in \mathbb{R}^d. \tag{A.2}$$

Next, we introduce two lemmas from [83]. The first lemma is adapted from Theorem 4.1.6 of [83], showing that a Hessian stability property of self-concordant functions:

Lemma A.1. Let ψ be a self-concordant function. Then, for any $p, \tilde{p} \in \mathrm{dom}(\psi)$:

$$\nabla^2\psi(p) \preceq \frac{1}{(1 - \|p - \tilde{p}\|_{\nabla^2\psi(\tilde{p})})^2} \nabla^2 f(\tilde{p}), \qquad \|p - \tilde{p}\|_{\nabla^2\psi(\tilde{p})} < 1. \tag{A.3}$$

The second lemma from Theorem 4.1.7 of [83] shows that a self-concordant function (see Definition 2.15) has a property that is similar in spirit to strongly convexity:

Lemma A.2. Let ψ be a self-concordant function. Then, for any $p, \tilde{p} \in \mathrm{dom}(\psi)$:

$$\psi(\tilde{p}) \geq \psi(p) + \langle \nabla\psi(p), \tilde{p} - p \rangle + \omega\left(\|\tilde{p} - p\|_{\nabla^2\psi(p)}\right),$$

where $w(s) := s - \log(1 + s)$.

In addition, for any $s \in [0, 1]$, $w(s)$ satisfies that

$$w(s) \geq \frac{s^2}{4}. \tag{A.4}$$

Another important inequality regarding the self-concordant function is given by the following lemma from (2.20) and (2.21) of [82].

Lemma A.3. If ψ is a self-concordant function with $p^\star := \arg\min \psi(p)$, then for some $p \in \mathrm{dom}(\psi)$ such that $\|\nabla\psi(p)\|_{(\nabla^2\psi(p))^{-1}} \leq \frac{1}{2}$, we have

$$\|p - p^*\|_{\nabla^2\psi(p)} \leq 2\|\nabla\psi(p)\|_{(\nabla^2\psi(p))^{-1}},$$
$$\|p - p^*\|_{\nabla^2\psi(p^*)} \leq 2\|\nabla\psi(p)\|_{(\nabla^2\psi(p))^{-1}}.$$

A.1.2 Proof of Theorem 2.1

Proof. Let us denote the average payoff over T rounds as:

$$\bar{V}_T := \frac{1}{T}\sum_{t=1}^{T} u\left(a_1^t, a_2^t\right) = \hat{p}^{T\top} u \hat{q}^T.$$

Because player 1 minimizes external regret, we have:

$$\max_{a \in \mathcal{A}_1} \frac{1}{T} \sum_{t=1}^{T} u\left(a, a_2^t\right) \leq \bar{V}_T + \varepsilon_T,$$

where $\varepsilon_T \to 0$. The left-hand side of the above inequality is equivalent to taking expectations over $\hat{q}^T$:

$$\max_{p \in \Delta(A_1)} p^\top u \hat{q}^T \leq \hat{p}^{T\top} u \hat{q}^T + \varepsilon_T.$$

Similarly, for player 2:

$$\min_{q \in \Delta(A_2)} \hat{p}^{T\top} u q \geq \hat{p}^{T\top} u \hat{q}^T - \varepsilon_T.$$

Combining both:

$$\max_{p \in \Delta(\mathcal{A}_1)} p^\top u \hat{q}^T - \varepsilon_T \leq \hat{p}^{T\top} M \hat{q}^T \leq \min_{q \in \Delta(\mathcal{A}_2)} \hat{p}^{T\top} u q + \varepsilon_T.$$

Hence, the duality gap is at most $2\varepsilon_T \to 0$. By the minimax theorem, the set of Nash equilibria is characterized by:

$$\max_p \min_q p^\top u q = \min_q \max_p p^\top u q = V^*,$$

and so any empirical pair $(\hat{p}^T, \hat{q}^T)$ converges to the set of saddle points, i.e., Nash equilibria. □

A.1.3 Proof of Theorem 2.2

Proof. Recall the definition of the empirical distribution over joint actions as follows:

$$\hat{P}^T(A) := \frac{1}{T} \sum_{t=1}^{T} \mathbf{1}[A^t = A], \qquad A \in \mathcal{A}.$$

Notice that, for any function $f : \mathcal{A} \to \mathbb{R}$, we have

$$\mathbb{E}_{A \sim \hat{P}^T}[f(A)] = \sum_{A \in \mathcal{A}} \hat{P}^T(A) f(A) = \frac{1}{T} \sum_{A \in \mathcal{A}} \sum_{t=1}^{T} \mathbf{1}[A^t = A] f(A) = \frac{1}{T} \sum_{t=1}^{T} f(A^t).$$

Then, for every player $n \in [N]$ and any function $F_n : \mathcal{A}_n \to \mathcal{A}_n$, applying the above identity with $u_n(F_n(A_n), A_{-n}) - u_n(A_n, A_{-n})$ gives

$$\mathbb{E}_{A\sim\hat{P}^T}[u_n(F_n(A_n), A_{-n}) - u_n(A_n, A_{-n})] = \frac{1}{T}\sum_{t=1}^{T}\left[u_n(F_n(a_n^t), a_{-n}^t) - u_n(a_n^t, a_{-n}^t)\right] \le \epsilon_T,$$

where the inequality is by the assumption of Theorem 2.2.

Since the above inequality holds for every player n and every mapping $F_n : \mathcal{A}_n \to \mathcal{A}_n$, it coincides with Definition 2.3, showing that $\hat{P}^T$ is an ϵ_T-correlated equilibrium.

Therefore, the sequence $\hat{P}^T$ lies in the set of ϵ_T-correlated equilibria. If $\epsilon_T \to 0$ as $T \to \infty$, every limit point of $\hat{P}^T$ belongs to the set of correlated equilibria. Equivalently, $\hat{P}^T$ converges to the set of correlated equilibria as $T \to \infty$. □

A.1.4 Proof of Lemma 2.5

Proof. Since OMD update $p^{t+1} = \underset{p\in\Omega}{\operatorname{argmin}}\ \eta_t \langle p, y^t\rangle + B_\psi(p; p^t)$, by the first-order optimality condition in (2.3), we have that

$$\left\langle \eta y^t + \nabla\psi(p^{t+1}) - \nabla\psi(p^t), p - p^{t+1}\right\rangle \ge 0.$$

Then, with Lemma 2.3, we have the one-step relationship as follows.

$$\begin{aligned}
\left\langle \eta y^t, p^t - p\right\rangle =& \left\langle \eta y^t, p^t - p^{t+1}\right\rangle + \left\langle \eta y^t, p^{t+1} - p\right\rangle \\
=& \left\langle \eta y^t, p^t - p^{t+1}\right\rangle + \left\langle \eta y^t + \nabla\psi(p^{t+1}) - \nabla\psi(p^t), p^{t+1} - p\right\rangle \\
& - \left\langle \nabla\psi(p^{t+1}) - \nabla\psi(p^t), p^{t+1} - p\right\rangle \\
\le& \left\langle \eta y^t, p^t - p^{t+1}\right\rangle + \left\langle \nabla\psi(p^{t+1}) - \nabla\psi(p^t), p - p^{t+1}\right\rangle \\
=& B_\psi(p; p^t) - B_\psi(p; p^{t+1}) - B_\psi(p^{t+1}; p^t) + \left\langle \eta y^t, p^t - p^{t+1}\right\rangle \\
\le& B_\psi(p; p^t) - B_\psi(p; p^{t+1}) + \max_{p\in\mathrm{dom}(\psi)} \left\langle \eta y^t, p^t - p\right\rangle - B_\psi(p; p^t).
\end{aligned} \tag{A.5}$$

Let $\tilde{p}^{t+1} = \underset{p\in\mathrm{dom}(\psi)}{\arg\max}\ \langle \eta y^t, p^t - p\rangle - B_\psi(p; p^t)$. Now, by the Fenchel-Young's inequality for quadratic norms in (A.2) and the tayler theorem such that $B_\psi(\tilde{p}^{t+1}; p^t) = \frac{1}{2}(p^{t+1} - p^t)^\top \nabla^2\psi(\tilde{z}^t)(p^{t+1} - p^t)$ for some $\tilde{z}^t$ lies on the line between p^{t+1} and p^t, we

have

$$\begin{aligned}\left\langle \eta y^t, p^t - \tilde{p}^{t+1} \right\rangle - B_\psi(\tilde{p}^{t+1}; p^t) &= \frac{1}{2}\|\eta y^t\|^2_{(\nabla^2\psi(\tilde{z}^t))^{-1}} + \frac{1}{2}\|p^t - \tilde{p}^{t+1}\|_{\nabla^2(\psi(\tilde{z}^t))} - \frac{1}{2}\|p^t - \tilde{p}^{t+1}\|_{\nabla^2(\psi(\tilde{z}^t))} \\ &= \frac{1}{2}\|\eta y^t\|^2_{(\nabla^2\psi(\tilde{z}^t))^{-1}}.\end{aligned}$$

Then, summing over $t = 1 \ldots, T$, and dividing both sides of (A.5), we have that

$$\sum_{t=1}^{T} \left\langle y^t, p^t - p \right\rangle \leq \frac{B_\psi(p; p^1)}{\eta} + \frac{\eta}{2}\sum_{t=1}^{T} \|y^t\|^2_{(\nabla^2\psi(\tilde{z}^t))^{-1}}.$$

□

A.1.5 Proof of Lemma 2.8

Proof. The proof is divided into the following steps.

1. We can break the one-step regret as follows:

$$\left\langle p - p^t, u^t \right\rangle = \underbrace{\left\langle p - g^t, u^t \right\rangle + \left\langle g^t - p^t, m^t \right\rangle}_{=:\text{(a)}} + \underbrace{\left\langle g^t - p^t, u^t - m^t \right\rangle}_{=:\text{(b)}}. \tag{A.6}$$

   In the first step, we will use induction to bound (a) in (A.6) for any $t \in [T]$ and $p \in \text{int}(\Omega)$ as follows.

$$\begin{aligned}\sum_{t=1}^{T} \left\langle p - g^t, u^t \right\rangle + \left\langle g^t - p^t, m^t \right\rangle \leq \frac{\psi(p)}{\eta} &- \frac{1}{\eta}\sum_{t=1}^{T} w(\|p^t - g^t\|_{\nabla^2\psi(p^t)}) \\ &- \frac{1}{\eta}\sum_{t=1}^{T} w(\|p^t - g^{t-1}\|_{\nabla^2\psi(g^{t-1})}),\end{aligned} \tag{A.7}$$

   where $w(s) := s - \log(1 + s)$.

2. Then, by Holder's inequality, we have (b) in (A.6) bounded as follows.

$$\left\langle g^t - p^t, u^t - m^t \right\rangle \leq \|g^t - p^t\|_{\nabla^2\psi(p^t)}\|u^t - m^t\|_{(\nabla^2\psi(p^t))^{-1}}.$$

   In this step, we will further show that $\|g^t - p^t\|_{\nabla^2\psi(p^t)} \leq 2\eta\|u^t - m^t\|_{(\nabla^2\psi(p^t))^{-1}}$.

Hence, we will obtain that

$$\left\langle g^t - p^t, u^t - m^t \right\rangle \leq 2\eta \|u^t - m^t\|^2_{(\nabla^2\psi(p^t))^{-1}}.$$

3. The final step will give a lower bound to the last two terms in (A.7) as follows.

$$\sum_{t=1}^{T} w(\|p^t - g^t\|_{\nabla^2\psi(p^t)}) + w(\|p^t - g^{t-1}\|_{\nabla^2\psi(g^{t-1})}) \geq \frac{1}{16}\sum_{t=1}^{T} \|p^t - p^{t-1}\|^2_{\nabla^2\psi(p^{t-1})}.$$

The proof follows by combining the three steps outlined above. We now provide the details for each step.

Step 1: Prove (A.7). When $T = 0$, (A.7) holds trivially because $\psi(p) \geq 0, \forall p \in \text{int}(\Omega)$. Now, we assume (A.7) holds for T, and prove the case for $T+1$. Let $p = g^T$, and since (A.7) holds for T, we have that

$$\sum_{t=1}^{T} \left\langle g^t - p^t, m^t \right\rangle - \left\langle g^t, u^t \right\rangle \leq -\sum_{t=1}^{T} \left\langle g^T, u^t \right\rangle + \frac{\psi(g^T)}{\eta} - \frac{1}{\eta}\sum_{t=1}^{T} \left(w(\|p^t - g^t\|_{\nabla^2\psi(p^t)}) + w(\|p^t - g^{t-1}\|_{\nabla^2\psi(g^{t-1})}) \right).$$

Now, adding the term $\left\langle g^{T+1} - p^{T+1}, m^{T+1} \right\rangle - \left\langle g^{T+1}, u^{T+1} \right\rangle$ on both sides gives

$$\begin{aligned}\sum_{t=1}^{T+1} \left\langle g^t - p^t, m^t \right\rangle - \left\langle g^t, u^t \right\rangle \leq -\sum_{t=1}^{T} \left\langle g^T, u^t \right\rangle + \frac{\psi(g^T)}{\eta} + \left\langle g^{T+1} - p^{T+1}, m^{T+1} \right\rangle \\ - \left\langle g^{T+1}, u^{T+1} \right\rangle - \frac{1}{\eta}\sum_{t=1}^{T} \left(w(\|p^t - g^t\|_{\nabla^2\psi(p^t)}) + w(\|p^t - g^{t-1}\|_{\nabla^2\psi(g^{t-1})}) \right).\end{aligned} \tag{A.8}$$

Denote by $\Psi^T(g) := \eta \left\langle \sum_{s=1}^{T} u^s, g \right\rangle - \psi(g)$. By the definition of $g^T = \underset{g \in \text{int}(\Omega)}{\arg\max}\, \Psi^T(g)$ and the first-order optimality, we have that $\nabla\Psi^T(g^T) = 0$. Invoking Lemma A.2 with $-\Psi^T$ being the self-concordant function, we have that

$$-\Psi^T(p^{T+1}) + \Psi^T(g^T) \geq w(\|p^{T+1} - g^T\|_{\nabla^2 - \Psi^T(g^T)}) = w(\|p^{T+1} - g^T\|_{\nabla^2\psi(g^T)}).$$

This means that

$$\left\langle g^T, \sum_{t=1}^{T} u^t \right\rangle - \frac{\psi(g^T)}{\eta} \geq \left\langle p^{T+1}, \sum_{t=1}^{T} u^t \right\rangle - \frac{\psi(p^{T+1})}{\eta} + \frac{1}{\eta} w(\|p^{T+1} - g^T\|_{\nabla^2\psi(g^T)}).$$

Substitute the above inequality to (A.8) gives

$$\begin{aligned}\sum_{t=1}^{T+1} \left\langle g^t - p^t, m^t \right\rangle - \left\langle g^t, u^t \right\rangle \leq & - \left\langle p^{T+1}, m^{T+1} + \sum_{t=1}^{T} u^t \right\rangle + \frac{\psi(p^{T+1})}{\eta} \\ & + \left\langle g^{T+1}, m^{T+1} - u^{T+1} \right\rangle - \frac{1}{\eta} \sum_{t=1}^{T} \left(w(\|p^t - g^t\|_{\nabla^2\psi(p^t)}) + w(\|p^t - g^{t-1}\|_{\nabla^2\psi(g^{t-1})}) \right) \\ & - \frac{1}{\eta} w(\|p^{T+1} - g^T\|_{\nabla^2\psi(g^T)}).\end{aligned} \tag{A.9}$$

On the other hand, denote by $\Phi^t(p) := \eta \left\langle m^t + \sum_{s=1}^{t-1} u^s, p \right\rangle - \psi(p)$ be the objection function for OFTRL. Since $p^{T+1} := \arg\max_{p \in \text{int}\,(\Omega)} \Phi^{T+1}(p)$, we have $\nabla\Phi^{T+1}(p^{T+1}) = 0$ by the first-order optimality condition. Invoking Lemma A.2 with $-\Phi^{T+1}$ being the self-concordant function, we have that

$$-\Phi^{T+1}(g^{T+1}) \geq -\Phi^{T+1}(p^{T+1}) + w(\|p^{T+1} - g^{T+1}\|_{\nabla^2\psi(p^{T+1})}),$$

which gives

$$\begin{aligned}\left\langle p^{T+1}, m^{T+1} + \sum_{t=1}^{T} u^t \right\rangle - \frac{\psi(p^{T+1})}{\eta} \geq & \left\langle g^{T+1}, m^{T+1} + \sum_{t=1}^{T} u^t \right\rangle - \frac{\psi(g^{T+1})}{\eta} \\ & + \frac{1}{\eta} w(\|p^{T+1} - g^{T+1}\|_{\nabla^2\psi(p^{T+1})}).\end{aligned}$$

Then, substitute the above inequality into (A.8), and we have

$$\begin{aligned}\sum_{t=1}^{T+1} \left\langle g^t - p^t, m^t \right\rangle - \left\langle g^t, u^t \right\rangle \leq & - \left\langle g^{T+1}, \sum_{t=1}^{T+1} u^t \right\rangle + \frac{\psi(g^{T+1})}{\eta} \\ & - \frac{1}{\eta} \sum_{t=1}^{T+1} \left(w(\|p^t - g^t\|_{\nabla^2\psi(p^t)}) + w(\|p^t - g^{t-1}\|_{\nabla^2\psi(g^{t-1})}) \right)\end{aligned} \tag{A.10}$$

Now, because $\Psi^{T+1}(g^{T+1}) \geq \Psi^{T+1}(p)$ for any $p \in \text{int}(\Omega)$, adding $\left\langle p, \sum_{t=1}^{T+1} u^t \right\rangle$ to both sides of the above inequality, we have

$$\sum_{t=1}^{T+1} \left\langle p - g^t, u^t \right\rangle + \left\langle g^t - p^t, m^t \right\rangle \leq \frac{\psi(p)}{\eta} - \frac{1}{\eta} \sum_{t=1}^{T+1} w(\|p^t - g^t\|_{\nabla^2 \psi(p^t)}) - \frac{1}{\eta} \sum_{t=1}^{T+1} w(\|p^t - g^{t-1}\|_{\nabla^2 \psi(g^{t-1})}),$$

which finishes the induction steps.

Step 2: Prove $\|g^t - p^t\|_{\nabla^2 \psi(p^t)} \leq 2\eta \|u^t - m^t\|_{(\nabla^2 \psi(p^t))^{-1}}$. By the definition of Φ^t and Ψ^t, we have that

$$\Psi^t(p) = \Phi^t(p) + \eta \left\langle p, u^t - m^t \right\rangle.$$

Since $\nabla \Phi^t(p^t) = 0$ by the first-order optimality and the definition of p^t, the above relationship between Φ^t and Ψ^t gives that

$$\nabla \Psi^t(p^t) = \eta(u^t - m^t).$$

Since $g^t = \arg\min(-\Psi^t)$, and

$$\|\nabla \Psi^t(p^t)\|_{(\nabla^2 \Psi(p^t))^{-1}} = \|\nabla \Psi^t(p^t)\|_{(\nabla^2 \psi(p^t))^{-1}} = \eta \|u^t - m^t\|_{(\nabla^2 \psi(p^t))^{-1}} < \frac{1}{2}$$

by assumption, invoking Lemma A.3 gives

$$\begin{aligned} \|g^t - p^t\|_{\nabla^2 \psi(p^t)} &= \|p^t - \arg\min(-\Psi^t)\|_{\nabla^2 \psi(p^t)} \\ &\leq 2 \|\nabla \Psi^t(p^t)\|_{(\nabla^2 \psi(p^t))^{-1}} \\ &= 2\eta \|u^t - m^t\|_{(\nabla^2 \psi(p^t))^{-1}}. \end{aligned} \tag{A.11}$$

Step 3: Lower bound last two terms in (A.7). We first use the inequality in (A.4) to prove that

$$w(\|p^t - g^t\|_{\nabla^2 \psi(p^t)}) + w(\|p^t - g^{t-1}\|_{\nabla^2 \psi(g^{t-1})}) \geq \frac{1}{4} \left(\|p^t - g^t\|^2_{\nabla^2 \psi(p^t)} + \|p^t - g^{t-1}\|^2_{\nabla^2 \psi(g^{t-1})} \right). \tag{A.12}$$

It suffices to show that $\|p^t - g^t\|_{\nabla^2 \psi(p^t)} \leq 1$ and $\|p^t - g^{t-1}\|_{\nabla^2 \psi(g^{t-1})} \leq 1$.

By (A.11), we have that

$$\|p^t - g^t\|_{\nabla^2\psi(p^t)} \leq 2\eta\|u^t - m^t\|_{(\nabla^2\psi(p^t))^{-1}} \leq \frac{1}{4}, \tag{A.13}$$

where the last inequality is due to the assumption that $\eta\|u^t - m^t\|_{(\nabla^2\psi(p^t))^{-1}} \leq \frac{1}{8}$.

For term $\|p^t - g^{t-1}\|_{\nabla^2\psi(g^{t-1})}$, we need to invoke Lemma A.3 again. Notice that the relationship between $\Psi^t(p)$ and $\Phi^t(p)$ is as follows:

$$\Phi^t(p) = \Psi^{t-1}(p) + \eta\left\langle p, m^t\right\rangle,$$

which implies

$$\nabla\Phi^t = \nabla\Psi^{t-1} + \eta m^t.$$

Since g^{t-1} minimize Ψ^{t-1}, we have $\nabla\Psi^{t-1}(g^{t-1}) = 0$ by the first-order optimality. Observe that

$$\|\nabla\Phi^t(g^{t-1})\|_{(\nabla^2\Psi(g^{t-1}))^{-1}} = \|\eta m^t\|_{(\nabla^2\psi(g^{t-1}))^{-1}} \leq \frac{1}{2}$$

by assumption. Thus, by Lemma A.3, we have

$$\begin{aligned}\|p^t - g^{t-1}\|_{\nabla^2\psi(g^{t-1})} &= \|g^{t-1} - p^t\|_{\nabla^2\psi(g^{t-1})} \\ &\leq 2\|\nabla\Phi^t(g^{t-1})\|_{(\nabla^2\psi(g^{t-1}))^{-1}} \\ &= 2\eta\|m^t\|_{(\nabla^2\psi(g^{t-1}))^{-1}} \leq 1,\end{aligned} \tag{A.14}$$

where the last inequality is due to the assumption that $\eta\|m^t\|_{(\nabla^2\psi(g^{t-1}))^{-1}} \leq \frac{1}{2}$.

Now, we have proven (A.12). The remaining step is to prove that

$$\sum_{t=1}^{T}\|p^t - g^t\|^2_{\nabla^2\psi(p^t)} + \|p^t - g^{t-1}\|^2_{\nabla^2\psi(g^{t-1})} \geq \frac{1}{4}\sum_{t=1}^{T}\|p^t - p^{t-1}\|^2_{\nabla^2\psi(p^{t-1})}.$$

By (A.13) and our assumption, it follows that $\nabla^2\psi(p^{t-1}) \preceq 2\nabla^2\psi\left(g^{(t-1)}\right)$. Thus, we have

$$\|p^t - g^{t-1}\|^2_{\nabla^2\psi(p^{t-1})} \leq 2\|p^t - g^{t-1}\|^2_{\nabla^2\psi(g^{t-1})}.$$

By the triangle inequality for norms, we further have

$$\begin{aligned}\|p^t - p^{t-1}\|^2_{\nabla^2\psi(p^{t-1})} &\leq 2\|p^t - g^{t-1}\|^2_{\nabla^2\psi(p^{t-1})} + 2\|g^{t-1} - p^{t-1}\|^2_{\nabla^2\psi(p^{t-1})} \\ &\leq 4\|p^t - g^{t-1}\|^2_{\nabla^2\psi(g^{t-1})} + 4\|p^{t-1} - g^{t-1}\|^2_{\nabla^2\psi(p^{t-1})}.\end{aligned}$$

Summing over $t = 1, \ldots, T$ gives the desired results:

$$\begin{aligned}\frac{1}{4}\sum_{t=1}^{T}\|p^t - p^{t-1}\|^2_{\nabla^2\psi(p^{t-1})} &\le \sum_{t=1}^{T}\|p^t - g^{t-1}\|^2_{\nabla^2\psi(g^{t-1})} + \|p^{t-1} - g^{t-1}\|^2_{\nabla^2\psi(p^{t-1})} \\ &\le \sum_{t=1}^{T}\|p^t - g^{t-1}\|^2_{\nabla^2\psi(g^{t-1})} + \|p^t - g^t\|^2_{\nabla^2\psi(p^t)}.\end{aligned}$$

□

A.1.6 Proof of Lemma 2.9

Proof. We adapt the proof in [6] to the following steps.

Step 1: Reduce the problem. We convert our learning problem to the domain as

$$\Delta^\circ := \left\{ p \in \mathbb{R}^{d-1}_{\ge 0} : \sum_{a\in[d-1]} p(a) \le 1 \right\},$$

and prove that two problems have the equal regret.

For notational convenience, define the final coordinate as $p(d) := 1 - \sum_{a\in[d-1]} p(a)$, so that the full d-dimensional distribution is implicitly specified. The log-barrier regularizer is then given by:

$$\tilde{\psi}(p) := -\sum_{a\in[d-1]} \ln(p(a)) - \ln\left(1 - \sum_{a\in[d-1]} p(a)\right).$$

To work in a $(d-1)$-dimensional space, we first transform the d-dimensional reward vector into a $(d-1)$-dimensional one. For each $a \in [d-1]$, define $\tilde{u}^t(a) := u^t(a) - u^t(d)$, and similarly, $\tilde{m}^t(a) := m^t(a) - m^t(d)$. This transformation preserves the regret. Specifically, for any $p, p^t \in \text{int}(\Delta^\circ)$, we have

$$\begin{aligned}\sum_{a\in[d-1]} \tilde{u}^t(a)\left(p(a) - p^t(a)\right) &= \sum_{a\in[d-1]} u^t(a)\left(p(a) - p^t(a)\right) - u^t(d)\sum_{a\in[d-1]}\left(p(a) - p^t(a)\right) \\ &= \sum_{a\in[d]} u^t(a)\left(p(a) - p^t(a)\right),\end{aligned}$$

where the last equality uses the notation $p(d) := 1 - \sum_{a\in[d-1]} p(a)$, and similarly

for $\tilde{p}^t(d)$. Thus, the transformed reward vector yields the same regret expression as the original.

Now, invoking Lemma 2.8 (assumptions verified later in Step 4 for clarity of exposition) on $\tilde{u}^1, \ldots, \tilde{u}^T$ gives

$$\sum_{t=1}^{T} \langle p - p^t, u^t \rangle = \sum_{t=1}^{T} \langle p - p^t, \tilde{u}^t \rangle \leq \frac{\psi(p)}{\eta} + 2\eta \sum_{t=1}^{T} \left\| \tilde{u}^t - \tilde{m}^t \right\|^2_{(\nabla^2 \tilde{\psi}(p^t))^{-1}} - \frac{1}{4\eta} \sum_{t=1}^{T} \|p^t - p^{t-1}\|_{\nabla^2 \tilde{\psi}(p^{t-1})}.$$

The lemma follows by showing $\|\tilde{u}^t - \tilde{m}^t\|^2_{(\nabla^2 \tilde{\psi}(p^t))^{-1}} \leq \|u^t - m^t\|^2_{(\nabla^2 \psi(p^t))^{-1}}$ and $\|p^t - p^{t-1}\|^2_{\nabla^2 \tilde{\psi}(p^{t-1})} = \|p^t - p^{t-1}\|^2_{\nabla^2 \psi(p^{t-1})}$.

Step 2: Prove $\|p^t - p^{t-1}\|^2_{\nabla^2 \tilde{\psi}(p^{t-1})} = \|p^t - p^{t-1}\|^2_{\nabla^2 \psi(p^{t-1})}$. Denote by $\tilde{p} = p^t - p^{t-1}$. We first show that $\|\tilde{p}\|^2_{\nabla^2 \tilde{\psi}(p^{t-1})} = \|\tilde{p}\|^2_{\nabla^2 \psi(p^{t-1})}$. The Hessian of $\tilde{\psi}$ can be computed as follows:

$$\nabla^2 \tilde{\psi}(p^{t-1}) = \text{diag}\left(\frac{1}{p^{t-1}(1)^2}, \ldots, \frac{1}{p^{t-1}(d-1)^2} \right) + \frac{1}{p^{t-1}(d)^2} \mathbf{1}_{d-1} \mathbf{1}_{d-1}^\top. \tag{A.15}$$

The Hessian of $\tilde{\psi}$ can be computed as (A.15), and the Hessian of ψ is

$$\nabla^2 \psi(p^{t-1}) = \text{diag}\left(\frac{1}{p^{t-1}(1)^2}, \ldots, \frac{1}{p^{t-1}(d)^2} \right).$$

Then, by definition of local norms, we have

$$\|\tilde{p}\|^2_{\nabla^2 \tilde{\psi}(p^{t-1})} = \sum_{a \in [d-1]} \frac{\tilde{p}(a)^2}{p^{t-1}(a)^2} + \frac{\left(\sum_{a \in [d-1]} \tilde{p}(a) \right)^2}{p^{t-1}(d)^2} = \sum_{a \in [d]} \frac{\tilde{p}(a)^2}{p^{t-1}(a)^2} = \|\tilde{p}\|_{\nabla^2 \psi(p^{t-1})},$$

where the second equality is due to $\tilde{p}(d) = p^t(d) - p^{t-1}(d) = 1 - \sum_{a \in [d-1]} p^t(a) - 1 + \sum_{a \in [d-1]} p^{t-1}(a) = -\sum_{a \in [d-1]} \tilde{p}(a)$.

Step 3: Prove $\|\tilde{u}^t - \tilde{m}^t\|^2_{(\nabla^2 \tilde{\psi}(p^t))^{-1}} \leq \|u^t - m^t\|^2_{(\nabla^2 \psi(p^t))^{-1}}$. Denote by $\tilde{u} := \tilde{u}^t - \tilde{m}^t$ and $u := u^t - m^t$. Because of our definition $\tilde{u}^t[a] = u^t(a) - u^t(d)$ and $\tilde{m}^t[a] = m^t(a) - m^t(d)$ for any $a \in [d-1]$, we have the relationship between $\tilde{u}(a)$ and $u(a)$

for any $a \in [d-1]$ as follows.

$$\tilde{u}(a) = \tilde{u}^t(a) - \tilde{m}^t(a) = u^t(a) - m^t(a) - (u^t(d) - m^t(d)) = u(a) - u(d). \tag{A.16}$$

Now, we show that$\|\tilde{u}\|^2_{(\nabla^2\tilde{\psi}(p^t))^{-1}} \le \|u\|^2_{(\nabla^2\psi(p^t))^{-1}}$. Since by Sherman–Morrison formula (i.e., $(D + \alpha\mathbf{1}\mathbf{1}^\top)^{-1} = D^{-1} - \frac{D^{-1}\mathbf{1}\mathbf{1}^\top D^{-1}}{\frac{1}{\alpha}+\mathbf{1}^\top D^{-1}\mathbf{1}}$), we have

$$(\nabla^2\tilde{\psi}(p^{t-1}))^{-1} = \mathsf{diag}(p^{t-1}(1)^2, \dots, p^{t-1}(d-1)^2) - \frac{(p^{t-1}(p^{t-1})^\top)^2}{\sum_{a\in[d]} p(a)^2},$$

and

$$(\nabla^2\psi(p^{t-1}))^{-1} = \mathsf{diag}(p^{t-1}(1)^2, \dots, p^{t-1}(d)^2).$$

Then, by the definition of local norms and (A.16)

$$\begin{aligned}
\|\tilde{u}\|^2_{(\nabla^2\tilde{\psi}(p^t))^{-1}} &= \sum_{a\in[d-1]} (u(a)-u(d))^2 p^t(a)^2 - \frac{\left(\sum_{a\in[d-1]} p^t(a)^2(u(a)-u(d))\right)^2}{\sum_{a\in[d]} p^t(a)^2} \\
&= \sum_{a\in[d]} (u(a)-u(d))^2 p^t(a)^2 - \frac{\left(\sum_{a\in[d]} p^t(a)^2(u(a)-u(d))\right)^2}{\sum_{a\in[d]} p^t(a)^2} \\
&= \sum_{a\in[d]} (u(a)-u(d))^2 p^t(a)^2 - \frac{\left(\sum_{a\in[d]} p^t(a)^2 u(a) - u(d)\sum_{a\in[d]} p^t(a)^2\right)^2}{\sum_{a\in[d]} p^t(a)^2} \\
&= \sum_{a\in[d]} \left(u(a)^2 - 2u(a)u(d) + u(d)^2\right) p^t(a)^2 - \frac{\left(\sum_{a\in[d]} p^t(a)^2 u(a)\right)^2}{\sum_{a\in[d]} p^t(a)^2} + u(d)\sum_{a\in[d]} 2u(a)p^t(a)^2 - u(d)^2\sum_{a\in[d]} p^t(a)^2 \\
&= \sum_{a\in[d]} u(a)^2 p^t(a)^2 - \frac{\left(\sum_{a\in[d]} p^t(a)^2 u(a)\right)^2}{\sum_{a\in[d]} p^t(a)^2} \\
&= \sum_{a\in[d]} u(a)^2 p^t(a)^2 - 2\frac{\left(\sum_{a\in[d]} p^t(a)^2 u(a)\right)^2}{\sum_{a\in[d]} p^t(a)^2} + \frac{\left(\sum_{a\in[d]} p^t(a)^2 u(a)\right)^2}{\sum_{a\in[d]} p^t(a)^2} \\
&= \sum_{a\in[d]} p^t(a)^2 \left(u(a)^2 - 2u(a)\frac{\sum_{a\in[d]} p^t(a)^2 u(a)}{\sum_{a\in[d]} p^t(a)^2} + \frac{\left(\sum_{a\in[d]} p^t(a)^2 u(a)\right)^2}{\left(\sum_{a\in[d]} p^t(a)^2\right)^2}\right) \\
&= \sum_{a\in[d]} p^t(a)^2 \left(u(a) - \frac{\sum_{a\in[d]} p^t(a)^2 u(a)}{\sum_{a\in[d]} p^t(a)^2}\right)^2.
\end{aligned}$$

Let $c_\star = \frac{\sum_{a\in[d]} p(a)^2 u(a)}{\sum_{a\in[d]} p(a)^2}$, and it is the minimizer of the quadratic function in the last equation. Therefore, we obtain that

$$\|\tilde{u}\|^2_{(\nabla^2\tilde{\psi}(p^t))^{-1}} = \|u - c_\star\mathbf{1}\|^2_{(\nabla^2\psi(p^t))^{-1}} \le \|u\|^2_{(\nabla^2\psi(p^t))^{-1}}.$$

Step 4: Prove the assumptions of Lemma 2.8 are satisfied. We show below that $\tilde{\psi}$ satisfies the assumptions of Lemma 2.8, i.e., a) $\nabla^2\psi(p)$ is positive definite; b) $\tilde{\psi}(p) \geq 0, \forall p \in \text{int}(\Delta^\circ)$ and c) $\nabla^2\tilde{\psi}(\tilde{p}) \preceq 2\nabla^2\tilde{\psi}(p)$ for any $p, \tilde{p} \in \text{int}(\Delta^\circ)$ with $\|p - \tilde{p}\|_{\nabla^2\tilde{\psi}(\tilde{p})} \leq \frac{1}{4}$.

a) By (A.15), it is clear that $\nabla^2\psi(p)$ is positive definite.

b) Since $p \in \text{int}(\Delta^\circ)$, we have $\ln(p(a)) < 0, \forall a \in [d-1]$ and $1 - \sum_{a\in[d-1]} p(a) \in [0, 1]$, so that $\ln\left(1 - \sum_{a\in[d-1]} p(a)\right) \leq 0$. Thus, $\tilde{\psi}(p) \geq 0, \forall p \in \text{int}(\Delta^\circ)$.

c) Since $\|p - \tilde{p}\|_{\nabla^2\tilde{\psi}(\tilde{p})} \leq \frac{1}{4}$, by invoking Lemma A.1, we have that

$$\nabla^2\tilde{\psi}(\tilde{p}) \preceq \frac{1}{(1 - \|\tilde{p} - p\|_{\nabla^2\tilde{\psi}(p)})^2}\nabla^2\tilde{\psi}(p) \leq \frac{16}{9}\nabla^2\tilde{\psi}(p) \leq 2\nabla^2\tilde{\psi}(p).$$

□

A.2 Proofs for Chapter 3

As the proofs are for any fixed individual player n, without confusion, we drop the subscript n in some notations for brevity. For example, we use the subscript n in $p_n^t(a)$ to denote agent n, while in $q_{a,a'}^t$, the agent index n is implicit. In addition, $y_a^t := 1 - u_n^t(a)$ is the loss suffered by player n in round t. Recall that $\mathcal{G}_t$ is the σ-algebra generated by the history information of all players till round t, and let $\mathbf{E}_t[\cdot] := \mathbf{E}[\cdot|\mathcal{G}_t]$ be the expectation conditioned on the history information by the end of round t. Let $\hat{L}_a^T := \sum_{t=1}^{T} \sum_{a' \in \mathcal{A}_n} q_{a,a'}^t \hat{Y}_{a,a'}^t$, $\bar{L}_{a,F(a)}^T := \sum_{t=1}^{T} p_n^t(a) y_{F(a)}^t$ and $\tilde{L}_{a,F(a)}^T := \sum_{t=1}^{T} \mathbf{1}[a_n^t = a] y_{F(a)}^t$.

A.2.1 Useful Facts

Definition A.2 (Supermartingale)**.** Let $\{X_t\}_{t\geq 0}$ be a sequence of integrable random variables adapted to a filtration $\{\mathcal{F}_t\}_{t\geq 0}$. We say that $\{X_t\}$ is a *supermartingale* with respect to $\{\mathcal{F}_t\}$ if for all $t \geq 0$,

$$\mathbf{E}[X_{t+1} \mid \mathcal{F}_t] \leq X_t \quad \text{almost surely}.$$

If $\{X_t\}_{t\geq 0}$ is a supermartingale with respect to $\{\mathcal{F}_t\}$, we have the following inequality held:

$$\mathbf{E}[X_{t+1}] \leq \mathbf{E}[X_t].$$

Lemma A.4 (Markov Inequality)**.** Let X be a non-negative random variable (i.e., $X \geq 0$ almost surely), and let $a > 0$. Then,

$$\Pr(X \geq a) \leq \frac{\mathbf{E}[X]}{a}.$$

Lemma A.5 (Azuma–Hoeffding inequality for (super)martingales)**.** If $\{X_t\}$ is a supermartingale with bounded differences (i.e., $|X_t - X_{t-1}| \leq c_t$), then with high probability:

$$\Pr[X_T - X_0 \geq \epsilon] \leq \exp\left(-\frac{\epsilon^2}{2\sum_{t=1}^{T} c_t^2}\right).$$

A.2.2 Proof of Lemma 3.1

Proof. By Markov inequality (Lemma A.4), we have that for any $\epsilon > 0$:

$$\begin{aligned}
&\Pr\left(\sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta_{a,a'}^t\left(\hat{Y}_{a,a'}^t - p_n^t(a)y_{a'}^t\right) > \epsilon\right)\\
=&\Pr\left(\exp\left\{\sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta_{a,a'}^t\left(\hat{Y}_{a,a'}^t - p_n^t(a)y_{a'}^t\right)\right\} > \exp\{\epsilon\}\right) \qquad \text{(A.17)}\\
\leq&\mathbf{E}\left[\exp\left\{\sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta_{a,a'}^t\left(\hat{Y}_{a,a'}^t - p_n^t(a)y_{a'}^t\right)\right\}\right]\exp\{-\epsilon\}.
\end{aligned}$$

In the following, we will construct a supermartingale sequence to upper bound the expectation term on the right-hand side (RHS) of (A.17). We first proving that the process $\{Z_t\}_{t\geq 0}$, where $Z_t := \exp\left\{\sum_{s=1}^{t}\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta_{a,a'}^s\left(\hat{Y}_{a,a'}^s - p_n^s(a)y_{a'}^s\right)\right\}$ for $t > 0$ and $Z_0 = 1$, is a supermartingale with respect to filtration $\{\mathcal{G}_t\}_{t\geq 0}$ for all $a \in \mathcal{A}_n$, i.e., $\mathbf{E}\left[Z_t|\mathcal{G}_{t-1}\right] \leq Z_{t-1}$. Denote by a_{-n}^t the actions of all players except player n in round t, and denote by $\mathbf{E}_{n,t-1}[\cdot] := \mathbf{E}_{t-1}\left[\cdot \mid a_{-n}^t\right]$. Then, we have that

$$\begin{aligned}
&\mathbf{E}_{t-1}\left[\exp\left\{\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta_{a,a'}^t\left(\hat{Y}_{a,a'}^t - p_n^t(a)y_{a'}^t\right)\right\}\right]\\
=&\mathbf{E}_{t-1}\left[\frac{\exp\left\{\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta_{a,a'}^t\hat{Y}_{a,a'}^t\right\}}{\exp\left\{\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta_{a,a'}^t p_n^t(a)y_{a'}^t\right\}}\right]\\
=&\mathbf{E}_{t-1}\left[\mathbf{E}_{n,t-1}\left[\frac{\exp\left\{\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta_{a,a'}^t\hat{Y}_{a,a'}^t\right\}}{\exp\left\{\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta_{a,a'}^t p_n^t(a)y_{a'}^t\right\}}\right]\right] \qquad \text{(A.18)}\\
=&\mathbf{E}_{t-1}\left[\frac{\mathbf{E}_{n,t-1}\left[\exp\left\{\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta_{a,a'}^t\hat{Y}_{a,a'}^t\right\}\right]}{\exp\left\{\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta_{a,a'}^t p_n^t(a)y_{a'}^t\right\}}\right],
\end{aligned}$$

where the second equality is due to the law of total expectation, and the last inequality is due to that $y_{a'}^t$ is determined given a_{-n}^t and $\beta_{a,a'}^t$ is $\mathcal{G}_{t-1}$-measurable.

Then, we want to prove that

$$\mathbf{E}_{n,t-1}\left[\exp\left\{\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta^t_{a,a'}\hat{Y}^t_{a,a'}\right\}\right] \leq \exp\left\{\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta^t_{a,a'}p^t_n(a)y^t_{a'}\right\}.$$

First, we can upper bound $\exp\left\{\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta^t_{a,a'}\hat{Y}^t_{a,a'}\right\}$ as follows.

$$\begin{aligned}
&\exp\left\{\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta^t_{a,a'}\hat{Y}^t_{a,a'}\right\}\\
&= \exp\left\{\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta^t_{a,a'}\frac{p^t_n(a)\mathbf{1}[a^t_n=a']q^t_{a,a'}y^t_{a'}}{p^t_n(a')(q^t_{a,a'}+\gamma_t)}\right\}\\
&= \prod_{a'\in\mathcal{A}_n}\exp\left\{\sum_{a\in\mathcal{A}_n}\beta^t_{a,a'}\frac{p^t_n(a)\mathbf{1}[a^t_n=a']q^t_{a,a'}y^t_{a'}}{p^t_n(a')(q^t_{a,a'}+\gamma_t)}\right\}\\
&\leq \prod_{a'\in\mathcal{A}_n}\sum_{a\in\mathcal{A}_n}\frac{p^t_n(a)q^t_{a,a'}}{p^t_n(a')}\exp\left\{\beta^t_{a,a'}\frac{\mathbf{1}[a^t_n=a']y^t_{a'}}{q^t_{a,a'}+\gamma_t}\right\}\\
&\leq \prod_{a'\in\mathcal{A}_n}\sum_{a\in\mathcal{A}_n}\frac{p^t_n(a)q^t_{a,a'}}{p^t_n(a')}\exp\left\{\frac{\beta^t_{a,a'}}{2\gamma_t}\frac{2\gamma_t\mathbf{1}[a^t_n=a']y^t_{a'}}{q^t_{a,a'}+\gamma_t\mathbf{1}[a^t_n=a']y^t_{a'}}\right\}\\
&= \prod_{a'\in\mathcal{A}_n}\sum_{a\in\mathcal{A}_n}\frac{p^t_n(a)q^t_{a,a'}}{p^t_n(a')}\exp\left\{\frac{\beta^t_{a,a'}}{2\gamma_t}\frac{2\gamma_t\mathbf{1}[a^t_n=a']y^t_{a'}/q^t_{a,a'}}{1+\gamma_t\mathbf{1}[a^t_n=a']y^t_{a'}/q^t_{a,a'}}\right\}\\
&\leq \prod_{a'\in\mathcal{A}_n}\sum_{a\in\mathcal{A}_n}\frac{p^t_n(a)q^t_{a,a'}}{p^t_n(a')}\exp\left\{\ln(1+\beta^t_{a,a'}\mathbf{1}[a^t_n=a']y^t_{a'}/q^t_{a,a'})\right\}\\
&= \prod_{a'\in\mathcal{A}_n}\sum_{a\in\mathcal{A}_n}\frac{p^t_n(a)q^t_{a,a'}}{p^t_n(a')}(1+\beta^t_{a,a'}\mathbf{1}[a^t_n=a']y^t_{a'}/q^t_{a,a'}).
\end{aligned}$$

where the first inequality is due to Jensen's inequality and the fact that $\forall a' \in \mathcal{A}_n$: $\sum_{a\in\mathcal{A}_n} p^t_n(a)q^t_{a,a'} = p^t_n(a')$, the second inequality is due to that $0 \leq \mathbf{1}[a^t_n = a']y^t_{a'} \leq 1$, the third inequality is because $\frac{z}{1+z/2} \leq \ln(1+z)$ for all $z > 0$, and $x\ln(1+y) \leq \ln(1+xy)$ for all $y > -1$ and $x \in [0,1]$. Then, taking the expectations on both sides,

we have that:

$$
\begin{aligned}
&\mathbf{E}_{n,t-1}\left[\exp\left\{\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta^t_{a,a'}\hat{Y}^t_{a,a'}\right\}\right] \\
&\leq \mathbf{E}_{n,t-1}\left[\prod_{a'\in\mathcal{A}_n}\sum_{a\in\mathcal{A}_n}\frac{p^t_n(a)q^t_{a,a'}}{p^t_n(a')}(1+\beta^t_{a,a'}\mathbf{1}[a^t_n=a']y^t_{a'}/q^t_{a,a'})\right] \\
&= \mathbf{E}_{n,t-1}\left[\prod_{a'\in\mathcal{A}_n}\left(1+\sum_{a\in\mathcal{A}_n}\frac{p^t_n(a)}{p^t_n(a')}\beta^t_{a,a'}\mathbf{1}[a^t_n=a']y^t_{a'}\right)\right] \\
&= \mathbf{E}_{n,t-1}\left[1+\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\frac{p^t_n(a)}{p^t_n(a')}\beta^t_{a,a'}\mathbf{1}[a^t_n=a']y^t_{a'}\right] \\
&= 1+\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}p^t_n(a)\beta^t_{a,a'}y^t_{a'} \leq \exp\left\{\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta^t_{a,a'}p^t_n(a)y^t_{a'}\right\},
\end{aligned}
$$

where the second equality is due to the fact that $\mathbf{1}[a^t_n=a']\cdot\mathbf{1}[a^t_n=a'']=0$ for any two distinct $a',a''\in\mathcal{A}_n$, and the last inequality is due to $1+x\leq\exp\{x\}$ for any $x\in\mathbb{R}$. Therefore, we have shown that (A.18) is bounded by 1. Thus,

$$
\mathbf{E}_{t-1}\left[Z_t\right]=\mathbf{E}_{t-1}\left[\exp\left\{\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta^t_{a,a'}\left(\hat{Y}^t_{a,a'}-p^t_n(a)y^t_{a'}\right)\right\}\right]\cdot Z_{t-1}\leq Z_{t-1},
$$

which shows that $\{Z_t\}_{t\geq 0}$ is a supermartingale with respect to filtration $\{\mathcal{G}_t\}_{t\geq 0}$. Thus, we have $\mathbf{E}\left[Z_T\right]\leq\mathbf{E}\left[Z_{T-1}\right]\leq\cdots\leq\mathbf{E}\left[Z_0\right]=1$, and substituting this fact to (A.17) gives

$$
\Pr\left(\sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\beta^t_{a,a'}\left(\hat{Y}^t_{a,a'}-\tilde{Y}^t_{a,a'}\right)\geq\epsilon\right)\leq 1\cdot\exp\{-\epsilon\}.
$$

The lemma follows by letting $\exp\{-\epsilon\}=\delta$ and solving ϵ for δ.

□

A.2.3 Proof of Theorem 3.4

Proof. By the relationship between p_n^t and q_a^t, we have the following equation held:

$$\begin{aligned}\sum_{a\in A_n} L_a^T &= \sum_{a\in\mathcal{A}_n}\sum_{t=1}^{T}\sum_{a'\in\mathcal{A}_n} Y_{a,a'}^t = \sum_{t=1}^{T}\sum_{a'\in\mathcal{A}_n}\sum_{a\in\mathcal{A}_n}\frac{\mathbf{1}[a_n^t=a']p_n^t(a)q_{a,a'}^t}{p_n^t(a')}y_{a'}^t \\ &= \sum_{t=1}^{T}\sum_{a'\in\mathcal{A}_n}\mathbf{1}[a_n^t=a']y_{a'}^t = \sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\mathbf{1}[a_n^t=a]y_a^t.\end{aligned} \tag{A.19}$$

Then, the regret defined in (3.1) can be rewritten in the loss form and can be decomposed as follows:

$$\begin{aligned}R_n^{\text{swa}}(T,\mathcal{F}_n) &= \max_{F\in\mathcal{F}_n}\sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\mathbf{1}[a_n^t=a]y_a^t - \sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\mathbf{1}[a_n^t=a]y_{F(a)}^t \\ &= \max_{F\in\mathcal{F}_n}\sum_{a\in\mathcal{A}_n}L_a^T - \sum_{a\in\mathcal{A}_n}\tilde{L}_{a,F(a)}^T \\ &= \max_{F\in\mathcal{F}_n}\underbrace{\sum_{a\in\mathcal{A}_n}(L_a^T-\hat{L}_a^T)}_{=:\text{(a)}} + \underbrace{\sum_{a\in\mathcal{A}_n}(\hat{L}_a^T-\hat{L}_{a,F(a)}^T)}_{=:\text{(b)}} \\ &\quad + \underbrace{\sum_{a\in\mathcal{A}_n}(\hat{L}_{a,F(a)}^T-\bar{L}_{a,F(a)}^T)}_{=:\text{(c)}} + \underbrace{\sum_{a\in\mathcal{A}_n}(\bar{L}_{a,F(a)}^T-\tilde{L}_{a,F(a)}^T)}_{=:\text{(d)}},\end{aligned} \tag{A.20}$$

where the second equality is due to (A.19), and in the third equality, we have $\hat{L}_a^T := \sum_{t=1}^{T}\sum_{a'\in\mathcal{A}_n} q_{a,a'}^t\hat{Y}_{a,a'}^t$, $\hat{L}_{a,F(a)}^t := \sum_{t=1}^{T}\hat{Y}_{a,F(a)}^t$, $\bar{L}_{a,F(a)}^T := \sum_{t=1}^{T}p_n^t(a)y_{F(a)}^t$, and $\tilde{L}_{a,F(a)}^T := \sum_{t=1}^{T}\mathbf{1}[a_n^t=a]y_{F(a)}^t$. In the following steps, we will give an upper bound to (a), (b), (c), and (d) for any $F\in\mathcal{F}_n$.

Step 1: Bound (a). By definition of L_a^T and $\hat{L}_a^T$, we have that

$$\begin{aligned}L_a^T - \hat{L}_a^T &= \sum_{t=1}^{T}\sum_{a'\in\mathcal{A}_n}Y_{a,a'}^t - \sum_{t=1}^{T}\sum_{a'\in\mathcal{A}_n}q_{a,a'}^t\hat{Y}_{a,a'}^t \\ &= \sum_{t=1}^{T}\sum_{a'\in\mathcal{A}_n}Y_{a,a'}^t\left(1-\frac{q_{a,a'}^t}{q_{a,a'}^t+\gamma_t}\right) = \sum_{t=1}^{T}\gamma_t\sum_{a'\in\mathcal{A}_n}\hat{Y}_{a,a'}^t.\end{aligned} \tag{A.21}$$

Thus, (a) is bounded by $\sum_{t=1}^{T} \gamma_t \sum_{a\in\mathcal{A}_n} \sum_{a'\in\mathcal{A}_n} \hat{Y}^t_{a,a'}$. For simplicity, we will analyze a fixed value γ_t, which we refer to as γ throughout the following analysis.

Invoking (3.6) of Lemma 3.2, we have with probability at least $1-\frac{\delta}{4}$:

$$\text{(a)} \le \gamma \sum_{t=1}^{T} \sum_{a'\in\mathcal{A}_n} p^t_n(a) y^t_{a'} + \ln\left(\frac{4}{\delta}\right).$$

Step 2: Bound (b). By the definition of $\hat{L}^T_a$ and $\hat{L}^T_{a,F(a)}$, we have that

$$\begin{aligned}\text{(b)} &= \sum_{t=1}^{T} \sum_{a\in\mathcal{A}_n} \sum_{a'}^{t} q^t_{a,a'} \hat{Y}^t_{a,a'} - \sum_{t=1}^{T} \sum_{a\in\mathcal{A}_n} \hat{Y}^t_{a,F(a)} \\ &\le \sum_{a\in\mathcal{A}_n} \max_{Q_a\in\Delta(\mathcal{A}_n)} \underbrace{\sum_{t=1}^{T} \left(\sum_{a'\in\mathcal{A}_n} q^t_{a,a'} \hat{Y}^t_{a,a'} - \sum_{a'\in\mathcal{A}_n} Q_a(a') \hat{Y}^t_{a,a'} \right)}_{=:\mathcal{R}_a}.\end{aligned}$$

Notice that $\mathcal{R}_a$ can be directly bounded by Lemma 2.6

$$\mathcal{R}_a \le \frac{\ln A_n}{\eta} + \frac{\eta}{2} \sum_{t=1}^{T} \sum_{a'\in\mathcal{A}_n} q^t_{a,a'} (\hat{Y}^t_{a,a'})^2 \le \frac{\ln A_n}{\eta} + \frac{\eta}{2} \sum_{t=1}^{T} \sum_{a'\in\mathcal{A}_n} \hat{Y}^t_{a,a'},$$

where the last inequality is due to

$$q^t_{a,a'} \hat{Y}^t_{a,a'} = q^t_{a,a'} \cdot \frac{Y^t_{a,a'}}{q^t_{a,a'} + \gamma_t} \le Y^t_{a,a'} = \frac{p^t_n(a) q^t_{a,a'} y^t_{a'}}{p^t_n(a')} \le 1.$$

The last inequality in the above equation is because $p^t_n(a') = \sum_{a\in\mathcal{A}_n} p^t_n(a) q^t_{a,a'}$. Then, summing over $a \in \mathcal{A}_n$ and invoking (3.6) of Lemma 3.2, we have with probability at least $1-\frac{\delta}{4}$:

$$\begin{aligned}\text{(b)} &\le A_n \frac{\ln A_n}{\eta} + \frac{\eta}{2} \sum_{t=1}^{T} \sum_{a\in\mathcal{A}_n} \sum_{a'\in\mathcal{A}_n} \hat{Y}^t_{a,a'} \\ &\le \frac{A_n \ln A_n}{\eta} + \frac{\eta}{2} \sum_{t=1}^{T} \sum_{a\in\mathcal{A}_n} \sum_{a'\in\mathcal{A}_n} p^t_n(a) y^t_{a'} + \frac{\eta}{2\gamma} \ln\left(\frac{4}{\delta}\right).\end{aligned}$$

Step 3: Bound (c). By using (3.8) of Lemma 3.2, we have (c) bounded simultaneously for all $F \in \mathcal{F}_n$ with probability at least $1 - \frac{\delta}{4}$, $(\mathrm{c}) \leq \frac{A_n}{\gamma} \ln\left(\frac{4A_n}{\delta}\right)$.

Step 4: Bound (d). Invoking Lemma 3.3, with probability at least $1 - \frac{\delta}{4}$, the following inequality holds simultaneously for all $F \in \mathcal{F}_n$, $(\mathrm{d}) \leq \sqrt{2A_n T \ln\left(\frac{4A_n}{\delta}\right)}$.

Step 5: Bound (a) + (b) + (c) + (d). Combining the above terms with union bound, for all $F \in \mathcal{F}_n$ with probability at least $1 - \delta$, we have

$$\begin{aligned} R_n^{\text{swa}}(T, \mathcal{F}_n) \leq & \frac{A_n \ln A_n}{\eta} + \left(\gamma + \frac{\eta}{2}\right) \sum_{t=1}^{T} \sum_{a \in \mathcal{A}_n} \sum_{a' \in \mathcal{A}_n} p_n^t(a) y_{a'}^t + \left(1 + \frac{\eta}{2\gamma}\right) \ln\left(\frac{4}{\delta}\right) + \frac{A_n}{\gamma} \ln\left(\frac{4A_n}{\delta}\right) + \sqrt{2TA_n \ln \frac{4A_n}{\delta}} \\ = & \frac{A_n \ln A_n}{\eta} + \eta T A_n + 2 \ln\left(\frac{4}{\delta}\right) + \frac{2A_n}{\eta} \ln\left(\frac{4A_n}{\delta}\right) + \sqrt{2TA_n \ln \frac{4A_n}{\delta}}, \end{aligned}$$

where the last equality is because we let $\gamma = \frac{\eta}{2}$. Then, let $\eta = \sqrt{\frac{\ln \frac{4A_n}{\delta}}{T}}$, we have

$$R_n^{\text{swa}}(T, \mathcal{F}_n) \leq 4A_n \sqrt{T \ln \frac{4A_n}{\delta}} + \sqrt{2TA_n \ln \frac{4A_n}{\delta}} + 2 \ln\left(\frac{4}{\delta}\right).$$

□

A.2.4 Proof of Theorem 3.5

Proof. By the relationship between p_n^t and q_a^t, we have the following equation held:

$$\begin{aligned} \sum_{a \in \mathcal{A}_n} L_a^T &= \sum_{a \in \mathcal{A}_n} \sum_{t=1}^{T} \sum_{a' \in \mathcal{A}_n} Y_{a,a'}^t = \sum_{t=1}^{T} \sum_{a' \in \mathcal{A}_n} \sum_{a \in \mathcal{A}_n} \frac{\mathbf{1}[a_n^t = a'] p_n^t(a) q_{a,a'}^t}{p_n^t(a')} y_{a'}^t \\ &= \sum_{t=1}^{T} \sum_{a' \in \mathcal{A}_n} \mathbf{1}[a_n^t = a'] y_{a'}^t = \sum_{t=1}^{T} \sum_{a \in \mathcal{A}_n} \mathbf{1}[a_n^t = a] y_a^t. \end{aligned} \tag{A.22}$$

Then, the regret defined in (4.1) can be rewritten in the loss form and can be

decomposed as follows:

$$
\begin{aligned}
R_n^{\text{swa}}(T, \mathcal{F}_n) &= \max_{F \in \mathcal{F}_n} \sum_{t=1}^{T} \sum_{a \in \mathcal{A}_n} \mathbf{1}[a_n^t = a] y_a^t - \sum_{t=1}^{T} \sum_{a \in \mathcal{A}_n} \mathbf{1}[a_n^t = a] y_{F(a)}^t \\
&= \max_{F \in \mathcal{F}_n} \sum_{a \in \mathcal{A}_n} L_a^T - \sum_{a \in \mathcal{A}_n} \tilde{L}_{a,F(a)}^T \\
&= \max_{F \in \mathcal{F}_n} \underbrace{\sum_{a \in \mathcal{A}_n} (L_a^T - \hat{L}_a^T)}_{=:\text{(a)}} + \underbrace{\sum_{a \in \mathcal{A}_n} (\hat{L}_a^T - \hat{L}_{a,F(a)}^T)}_{=:\text{(b)}} \\
&\quad + \underbrace{\sum_{a \in \mathcal{A}_n} (\hat{L}_{a,F(a)}^T - \bar{L}_{a,F(a)}^T)}_{=:\text{(c)}} + \underbrace{\sum_{a \in \mathcal{A}_n} (\bar{L}_{a,F(a)}^T - \tilde{L}_{a,F(a)}^T)}_{=:\text{(d)}},
\end{aligned}
\tag{A.23}
$$

where the second equality is due to (A.22), and in the third equality, we have $\hat{L}_a^T := \sum_{t=1}^{T} \sum_{a' \in \mathcal{A}_n} q_{a,a'}^t \hat{Y}_{a,a'}^t$, $\hat{L}_{a,F(a)}^t := \sum_{t=1}^{T} \hat{Y}_{a,F(a)}^t$, $\bar{L}_{a,F(a)}^T := \sum_{t=1}^{T} p_n^t(a) y_{F(a)}^t$ and $\tilde{L}_{a,F(a)}^T := \sum_{t=1}^{T} \mathbf{1}[a_n^t = a] y_{F(a)}^t$. In the following steps, we will give an upper bound to (a), (b), (c), and (d) for any $F \in \mathcal{F}_n$.

Step 1: Bound (a). By definition of L_a^T and $\hat{L}_a^T$, we have that

$$
\begin{aligned}
L_a^T - \hat{L}_a^T &= \sum_{t=1}^{T} \sum_{a' \in \mathcal{A}_n} Y_{a,a'}^t - \sum_{t=1}^{T} \sum_{a' \in \mathcal{A}_n} q_{a,a'}^t \hat{Y}_{a,a'}^t \\
&= \sum_{t=1}^{T} \sum_{a' \in \mathcal{A}_n} Y_{a,a'}^t \left(1 - \frac{q_{a,a'}^t}{q_{a,a'}^t + \gamma_t}\right) = \sum_{t=1}^{T} \gamma_t \sum_{a' \in \mathcal{A}_n} \hat{Y}_{a,a'}^t.
\end{aligned}
\tag{A.24}
$$

Thus, (a) is bounded by $\sum_{t=1}^{T} \gamma_t \sum_{a \in \mathcal{A}_n} \sum_{a' \in \mathcal{A}_n} \hat{Y}_{a,a'}^t$. For simplicity, we will analyze a fixed value γ_t, which we refer to as γ throughout the following analysis.

Now, invoking (3.6) in Lemma 3.2, we have that with probability $1 - \frac{\delta}{4}$:

$$
\begin{aligned}
\text{(a)} &\leq \gamma \sum_{t=1}^{T} \sum_{a \in \mathcal{A}_n} \sum_{a' \in \mathcal{A}_n} p_n^t(a) y_{a'}^t + \ln\left(\frac{4}{\delta}\right) \\
&\leq \gamma T A_n + \ln\left(\frac{4}{\delta}\right),
\end{aligned}
$$

Step 2: Bound (b). By the definition of $\hat{L}_a^T$ and $\hat{L}_{a,F(a)}^T$, we have that

$$\begin{aligned}(\mathrm{b}) &= \sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\sum_{a'}^{t} q_{a,a'}^t \hat{Y}_{a,a'}^t - \sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\hat{Y}_{a,F(a)}^t \\ &\le \sum_{a\in\mathcal{A}_n}\max_{Q_a\in\Delta(\mathcal{A}_n)} \underbrace{\sum_{t=1}^{T}\left(\sum_{a'\in\mathcal{A}_n} q_{a,a'}^t \hat{Y}_{a,a'}^t - \sum_{a'\in\mathcal{A}_n} Q_a(a')\hat{Y}_{a,a'}^t\right)}_{=:\mathcal{R}_a}.\end{aligned}$$

Notice that $\mathcal{R}_a$ can be directly bounded by Lemma 2.7

$$\begin{aligned}\mathcal{R}_a &\le \frac{(A_n)^{1-\beta}-1}{(1-\beta)\eta} + \frac{\eta}{\beta}\sum_{t=1}^{T}\sum_{a'\in\mathcal{A}_n}(q_{a,a'}^t)^{2-\beta}(\hat{Y}_{a,a'}^t)^2 \\ &\le \frac{(A_n)^{1-\beta}-1}{(1-\beta)\eta} + \frac{\eta}{\beta}\sum_{t=1}^{T}\sum_{a'\in\mathcal{A}_n}(q_{a,a'}^t)^{1-\beta}\hat{Y}_{a,a'}^t,\end{aligned}$$

where the last inequality is due to

$$q_{a,a'}^t\hat{Y}_{a,a'}^t = q_{a,a'}^t\cdot\frac{Y_{a,a'}^t}{q_{a,a'}^t+\gamma_t} \le 1.$$

Then, summing over $a\in\mathcal{A}_n$ gives

$$\begin{aligned}(\mathrm{b}) &\le A_n\frac{(A_n)^{1-\beta}-1}{(1-\beta)\eta} + \frac{\eta}{\beta}\sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}(q_{a,a'}^t)^{1-\beta}\hat{Y}_{a,a'}^t \\ &\le \frac{2A_n\sqrt{A_n}-2A_n}{\eta} + 2\eta\sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}(q_{a,a'}^t)^{0.5}\hat{Y}_{a,a'}^t\end{aligned}$$

where the second inequality is due to the fact $q_{a,a'}^t\hat{Y}_{a,a'}^t\le 1$, and the last inequality is due to we set $\beta=0.5$.

Then, invoking (3.7), we have with probability $1-\frac{\delta}{4}$ that

$$\begin{aligned}(b) &\le \frac{2A_n\sqrt{A_n}-2A_n}{\eta} + 2\eta\sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}(q_{a,a'}^t)^{0.5}p_n^t(a)y_{a'}^t + \frac{2\eta}{\gamma}\ln\left(\frac{4}{\delta}\right) \\ &\le \frac{2A_n\sqrt{A_n}-2A_n}{\eta} + 2\eta T\sqrt{A_n} + \frac{2\eta}{\gamma}\ln\left(\frac{4}{\delta}\right),\end{aligned}$$

where the second inequality is due to the Hölder's inequality:

$$\sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}(q_{a,a'}^t)^{0.5}p_n^t(a)y_{a'}^t = \sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}p_n^t(a)\left(\sum_{a'\in\mathcal{A}_n}\left((q_{a,a'}^t)^{0.5}\right)^2\right)^{0.5}\left(\sum_{a'\in\mathcal{A}_n}1^2\right)^{0.5}$$
$$= \sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}p_n^t(a)\sqrt{A_n} = T\sqrt{A_n}.$$

Step 3: Bound (c). By using (3.8) of Lemma 3.2, we have (c) bounded simultaneously for all $F\in\mathcal{F}_n$ with probability at least $1-\frac{\delta}{4}$, $(\mathrm{c})\le\frac{A_n}{\gamma}\ln\left(\frac{4A_n}{\delta}\right)$.

Step 4: Bound (d). Invoking Lemma 3.3, with probability at least $1-\frac{\delta}{4}$, the following inequality holds simultaneously for all $F\in\mathcal{F}_n$, $(\mathrm{d})\le\sqrt{2A_nT\ln\left(\frac{4A_n}{\delta}\right)}$.

Step 5: Bound (a) + (b) + (c) + (d). With union bound and with probability at least $1-\delta$, we have

$$R_n^{\mathsf{swa}}(T,\mathcal{F}_n)\le\gamma TA_n+\frac{2A_n\sqrt{A_n}-2A_n}{\eta}+2\eta T\sqrt{A_n}+\left(1+\frac{2\eta}{\gamma}\right)\ln\left(\frac{4}{\delta}\right)$$
$$+\frac{A_n}{\gamma}\ln(\frac{4A_n}{\delta})+\sqrt{2A_nT\ln\left(\frac{4A_n}{\delta}\right)}.$$

Now, let $\gamma=\sqrt{\frac{\ln(4A_n/\delta)}{T}}$ and $\eta=\sqrt{\frac{A_n}{T}}$, we have that

$$R_n^{\mathsf{swa}}(T,\mathcal{F}_n)\le A_n\sqrt{T\ln\left(\frac{4A_n}{\delta}\right)}+2A_n\sqrt{T}-2\sqrt{A_nT}+\ln\left(\frac{4}{\delta}\right)+2\sqrt{A_n}$$
$$+A_n\sqrt{T\ln\left(\frac{4A_n}{\delta}\right)}+\sqrt{2A_nT\ln\left(\frac{4A_n}{\delta}\right)}$$
$$\le 2A_n\sqrt{T\ln\left(\frac{4A_n}{\delta}\right)}+2A_n\sqrt{T}+\ln\left(\frac{4}{\delta}\right)+\sqrt{2A_nT\ln\left(\frac{4A_n}{\delta}\right)}.$$

□

A.3 Proofs for Chapter 4

A.3.1 Useful Facts

Definition A.3 (Directed Tree)**.** A directed graph $\mathcal{T}_a = (V, E)$ is a directed tree rooted at node a if it satisfies the following conditions:

1. it contains no (directed) cycles,
2. every node in $V \setminus a$ has exactly one outgoing edge,
3. the root node a does not have any outgoing edges.

Then, let $\mathbb{T}_a$ denote the set of all $\mathcal{T}_a$, and we are ready to state the Markov chain tree theorem as follows.

Theorem A.6 (Markov Chain Tree Theorem [8])**.** Let $p \in \Delta^m$ be a stationary distribution for an ergodic (i.e., aperiodic and irreducible) Markov chain with the transition matrix described by Q, which can be calculated as follows:

$$p(a) = \frac{\Sigma_a}{\Sigma}, \tag{A.25}$$

where $\Sigma_a := \sum_{\mathcal{T} \in \mathbb{T}_a} \prod_{(u,v) \in E(\mathcal{T})} Q(u, v)$, and $\sum := \sum_a \Sigma_a$.

A.3.2 Proof of Lemma 4.1

Proof. The proof follows the two steps below.

1. We claim that for any $t \in [T]$:

$$\begin{aligned} &\sum_{a \in \mathcal{A}_n} \left\| p_n^t(a) \hat{u}_n^t - p_n^{t-1}(a) m_n^t \right\|_{*, q_a^t}^2 \\ &\leq 2 \sum_{a \in \mathcal{A}_n} \left| u_n^t(a) - m_n^t(a) \right| \mathbf{1}[a_n^t = a] + 2 \left\| p_n^t - p_n^{t-1} \right\|_1 . \end{aligned} \tag{A.26}$$

2. Next, we complete the proof by demonstrating that when summed over $t \in [T]$, we have

$$\sum_{t=1}^{T} \sum_{a \in \mathcal{A}_n} \left| u_n^t(a) - m_n^t(a) \right| \mathbf{1}[a_n^t = a] = \sum_{t=1} \| u_n^t - u_n^{t-1} \|_1.$$

Step 1. Now, we prove our claim in Step 1 as follows. Recall that

$$\|x\|_{*,q_a^t} := \sqrt{\sum_{a' \in \mathcal{A}_n} (x(a')q_a^t(a'))^2}.$$

Then, we have

$$\begin{aligned}
&\left\|p_n^t(a)\hat{u}_n^t - p_n^{t-1}(a)m_n^t\right\|_{*,q_a^t}^2 \\
&= \sum_{a' \in \mathcal{A}_n} (q_a^t(a'))^2 \left(p_n^t(a)\hat{u}_n^t - p_n^t(a)m_n^t + p_n^t(a)m_n^t - p_n^{t-1}(a)m_n^t\right)^2 \\
&\leq 2 \sum_{a' \in \mathcal{A}_n} (q_a^t(a'))^2 \left(\left(p_n^t(a)\hat{u}_n^t - p_n^t(a)m_n^t\right)^2 + \left(p_n^t(a)m_n^t - p_n^{t-1}(a)m_n^t\right)^2\right) \\
&= 2\left\|p_n^t(a)\hat{u}_n^t - p_n^t(a)m_n^t\right\|_{*,q_a^t}^2 + 2\left\|p_n^t(a)m_n^t - p_n^{t-1}(a)m_n^t\right\|_{*,q_a^t}^2,
\end{aligned} \tag{A.27}$$

where the inequality is due to the Cauchy-Schwarz inequality.

Summing over $a \in \mathcal{A}_n$, the first term on the RHS of (A.27) can be further bounded as follows.

$$\begin{aligned}
&2 \sum_{a \in \mathcal{A}_n} \left\|p_n^t(a)\hat{u}_n^t - p_n^t(a)m_n^t\right\|_{*,q_a^t}^2 \\
&= 2 \sum_{a \in \mathcal{A}_n} \sum_{a' \in \mathcal{A}_n} (q_a^t(a')^2 \left(\frac{p_n^t(a)(u_n^t(a') - m_n^t(a'))}{p_n^t(a')}\right)^2 \mathbf{1}[a_n^t = a'] \\
&\leq 2 \sum_{a \in \mathcal{A}_n} \sum_{a' \in \mathcal{A}_n} \frac{p_n^t(a)q_a^t(a')}{p_n^t(a')} |u_n^t(a') - m_n^t(a')| \mathbf{1}[a_n^t = a'] \\
&= 2 \sum_{a' \in \mathcal{A}_n} |u_n^t(a') - m_n^t(a')| \mathbf{1}[a_n^t = a'],
\end{aligned}$$

where the first inequality is due to $\frac{p_n^t(a)q_a^t(a')}{p_n^t(a')} \leq 1$ and $|u_n^t(a') - m_n^t(a')| \leq 1$. The last equality is due to the definition of p_n^t such that $p_n^t(a') = \sum_{a \in \mathcal{A}_n} p_n^t(a)q_a^t(a')$.

Regarding the second term on the RHS of (A.27), we have that

$$
\begin{aligned}
&2 \sum_{a \in \mathcal{A}_n} \left\| p_n^t(a) m_n^t - p_n^{t-1}(a) m_n^t \right\|_{*, q_a^t}^2 \\
&= 2 \sum_{a \in \mathcal{A}_n} \sum_{a' \in \mathcal{A}_n} (q_a^t(a'))^2 \left(p_n^t(a) m_n^t(a') - p_n^{t-1}(a) m_n^t(a') \right)^2 \\
&\leq 2 \sum_{a \in \mathcal{A}_n} \sum_{a' \in \mathcal{A}_n} q_a^t(a') |p_n^t(a) - p_n^{t-1}(a)| \\
&= 2 \| p_n^t - p_n^{t-1} \|_1,
\end{aligned}
$$

where the inequality is due to that $m_n^t(a') \leq 1$ and $q_a^t(a') \leq 1$ for all $a' \in \mathcal{A}_n$, and the last equality is due to that $\sum_{a' \in \mathcal{A}_n} q_a^t(a') = 1$.

Step 2. Then, Step 2 is obtained by summing over $t \in [T]$ as follows.

$$
\begin{aligned}
&\sum_{t=1}^{T} \sum_{a \in \mathcal{A}_n} \left| u_n^t(a) - m_n^t(a) \right| \mathbf{1}[a_n^t = a] = \sum_{t=1}^{T} \sum_{a \in \mathcal{A}_n} \left| u_n^t(a) - u_n^{\tau_t(a)}(a) \right| \mathbf{1}[a_n^t = a] \\
&\leq \sum_{a \in \mathcal{A}_n} \sum_{t=1}^{T} \mathbf{1}[a_n^t = a] \sum_{s=\tau_t(a)+1}^{t} |u_n^s(a) - u_n^{s-1}(a)| \; = \sum_{a \in \mathcal{A}_n} \sum_{t=1}^{T} |u_n^t(a) - u_n^{t-1}(a)|.
\end{aligned}
$$

□

A.3.3 Proof of Lemma A.7

Lemma A.7. When $\eta \leq \frac{1}{16}$, we have that

$$
\sum_{t=1}^{T} \sum_{a \in \mathcal{A}_n} \left\| q_a^t - q_a^{t-1} \right\|_{q_a^{t-1}}^2 \geq \frac{1}{64 A_n} \sum_{t=1}^{T} \left\| p_n^t - p_n^{t-1} \right\|_1^2 .
$$

Before we prove Lemma A.7, we need the following lemma.

Lemma A.8 (Multiplicative Stability for Bandits)**.** For $\eta \leq \frac{1}{16}$, $\|u_n^t\|_\infty \leq 1$ and $\|p_n^{t-1}(a) m_n^t\|_\infty \leq 1$ for all $t \in [T]$ and $a \in \mathcal{A}_n$, we have that with the log-barrier regularizer:

$$
\sum_{a \in \mathcal{A}_n} \| q_a^t - q_a^{t-1} \|_{q_a^{t-1}} \leq \frac{1}{2}.
$$

Proof of Lemma A.8. By the triangle inequality, we have

$$\sum_{a\in\mathcal{A}_n}\|q_a^t-q_a^{t-1}\|_{q_a^{t-1}}\le\sum_{a\in\mathcal{A}_n}\|q_a^t-g_a^{t-1}\|_{q_a^{t-1}}+\sum_{a\in\mathcal{A}_n}\|g_a^{t-1}-q_a^{t-1}\|_{q_a^{t-1}}.$$

Then, we need to show that both terms on the RHS of the above inequality are no larger than $\frac{1}{4}$ to prove the lemma. Let $\Psi_a^t(g):=\eta\left\langle\sum_{s=1}^t p_n^s(a)\hat{u}_n^s,g\right\rangle-\psi(g)$, and denote by $g_a^t:=\operatorname*{arg\,max}_{g\in\text{relint}(\Delta(\mathcal{A}_n))}\Psi^t(g)$ the maximizer of $\Psi_a^t(g)$. Similarly, let $\Phi_a^t(q):=\eta\left\langle p_n^{t-1}(a)m_n^t+\sum_{s=1}^{t-1}p_n^s(a)\hat{u}_n^s,q\right\rangle-\psi(q)$, and denote by $q_a^t:=\operatorname*{arg\,max}_{q\in\text{relint}(\Delta(\mathcal{A}_n))}\Phi_a^t(q)$ be the maximizer of $\Phi_a^t(q)$.

Step 1: Bound $\sum_{a\in\mathcal{A}_n}\|q_a^t-g_a^{t-1}\|_{q_a^{t-1}}$. We can apply the same reasoning as in (A.14). Notice that

$$\Phi_a^t(g)=\Psi_a^{t-1}(g)+\eta\left\langle p_n^{t-1}(a)m_n^t,g\right\rangle,$$

which implies

$$\nabla\Phi_a^t=\nabla\Psi_a^{t-1}+\eta p_n^{t-1}(a)m_n^t.$$

Since g_a^{t-1} minimizes Ψ_a^{t-1}, we have $\nabla\Psi_a^{t-1}(g_a^{t-1})=0$ by the first-order optimality. Thus, we have

$$\left\|\nabla\Phi_a^t\left(g_a^{t-1}\right)\right\|_{*,g_a^{t-1}}=\eta\left\|p_n^{t-1}(a)m_n^t\right\|_{*,g_a^{t-1}}\le\eta\le\frac{1}{2},$$

where the first inequality is due to $g_a^{t-1}\in\text{relint}(\Delta(\mathcal{A}_n))$, $p_n^{t-1}(a)\le 1$ and $m_n^t\le 1$. Therefore, we can invoke Lemma A.3 to have

$$\begin{aligned}\sum_{a\in\mathcal{A}_n}\|q_a^t-g_a^{t-1}\|_{q_a^{t-1}}&=\sum_{a\in\mathcal{A}_n}\|g_a^{t-1}-\operatorname*{arg\,min}_{q\in\text{relint}(\Delta(\mathcal{A}_n))}(-\Phi^t(q))\|_{q_a^{t-1}}\\&\le 2\sum_{a\in\mathcal{A}_n}\|\nabla\Phi_a^t\left(g_a^{t-1}\right)\|_{*,g_a^{t-1}}\\&=2\eta\sum_{a\in\mathcal{A}_n}\|p_n^{t-1}(a)m_n^t\|_{*,g_a^{t-1}}\\&\le 2\eta\|m_n^t\|_{*,g_a^{t-1}}\le\frac{1}{8}<\frac{1}{4},\end{aligned}$$

where the second inequality is by the assumption that $\eta\le\frac{1}{16}$.

Step 2: Bound $\sum_{a\in\mathcal{A}_n}\|g_a^{t-1}-q_a^{t-1}\|_{q_a^{t-1}}$. Similar to (A.11), we first observe that

$$\Psi_a^{t-1}(q)=\Phi_a^{t-1}(q)+\eta\left\langle q,p_n^{t-1}(a)\hat{u}_n^{t-1}-p_n^{t-2}(a)m_n^{t-1}\right\rangle,$$

which implies

$$\nabla\Psi_a^{t-1}=\nabla\Phi_a^{t-1}+\eta\left(p_n^{t-1}(a)\hat{u}_n^{t-1}-p_n^{t-2}(a)m_n^{t-1}\right).$$

Since q_a^{t-1} minimizes Φ_a^{t-1}, we have $\nabla\Phi_a^{t-1}(q_a^{t-1})=0$ by the first-order optimality. Thus, we have

$$\nabla\Psi_a^{t-1}(q_a^{t-1})=\eta\left(p_n^{t-1}(a)\hat{u}_n^{t-1}-p_n^{t-2}(a)m_n^{t-1}\right).$$

Then, we want to invoke Lemma A.3 again, and we need to show $\|\nabla\Psi_a^{t-1}(q_a^{t-1})\|_{*,q_a^{t-1}}\leq\frac{1}{2}$. By the triangle inequality, we have that

$$\begin{aligned}\left\|p_n^{t-1}(a)\hat{u}_n^{t-1}-p_n^{t-2}(a)m_n^{t-1}\right\|_{*,q_a^{t-1}}\leq&\left\|p_n^{t-1}(a)\hat{u}_n^{t-1}-p_n^{t-1}(a)m_n^{t-1}\right\|_{*,q_a^{t-1}}\\&+\left\|p_n^{t-1}(a)m_n^{t-1}-p_n^{t-2}(a)m_n^{t-1}\right\|_{*,q_a^{t-1}}.\end{aligned}\tag{A.28}$$

Then, the first term on the RHS of the above inequality can be bounded as follows

$$\begin{aligned}&\|p_n^{t-1}(a)\hat{u}_n^{t-1}-p_n^{t-1}(a)m_n^{t-1}\|_{*,q_a^{t-1}}\\&\leq\sqrt{\sum_{a'\in\mathcal{A}_n}(q_a^{t-1}(a'))^2\left(\frac{p_n^{t-1}(a)\mathbf{1}[a_n^{t-1}=a'](u_n^t(a')-m_n^t(a'))}{p_n^{t-1}(a')}\right)^2}\\&\leq\sqrt{\sum_{a'\in\mathcal{A}_n}\left(\frac{p_n^{t-1}(a)q_a^{t-1}(a')}{p_n^{t-1}(a')}\right)^2\mathbf{1}[a_n^{t-1}=a']}=\frac{p_n^{t-1}(a)q_a^{t-1}(a_n^{t-1})}{p_n^{t-1}(a_n^{t-1})},\end{aligned}\tag{A.29}$$

where the second inequality is due to that $|u_n^t(a) - m_n^t(a)| \leq 1$. The second term on the RHS of (A.28) can be bounded as follows

$$\begin{aligned}
&\|p_n^{t-1}(a)m_n^{t-1} - p_n^{t-2}(a)m_n^{t-1}\|_{*,q_a^{t-1}} \\
&= \sqrt{\sum_{a' \in \mathcal{A}_n} (q_a^{t-1}(a'))^2 \left(p_n^{t-1}(a)m_n^{t-1}(a') - p_n^{t-2}(a)m_n^{t-1}(a')\right)^2} \\
&\leq \sqrt{\sum_{a' \in \mathcal{A}_n} q_a^{t-1}(a') \left(p_n^{t-1}(a) - p_n^{t-2}(a)\right)^2} \\
&= p_n^{t-1}(a) - p_n^{t-2}(a).
\end{aligned} \tag{A.30}$$

Then, substituting (A.29) and (A.30) into (A.28) gives

$$\begin{aligned}
\|\nabla \Psi^{t-1}(q_a^{t-1})\|_{*,q_a^{t-1}} &= \eta \left\|p_n^{t-1}(a)\hat{u}_n^{t-1} - p_n^{t-2}(a)m_n^{t-1}\right\|_{*,q_a^{t-1}} \\
&\leq \eta \frac{p_n^{t-1}(a)q_a^{t-1}(a_n^{t-1})}{p_n^{t-1}(a_n^{t-1})} + \eta(p_n^{t-1}(a) - p_n^{t-2}(a)) \\
&\leq 2\eta \leq \frac{1}{8} < \frac{1}{2},
\end{aligned} \tag{A.31}$$

where the first inequality is due to that $p_n^{t-1}(a') = \sum_{a \in \mathcal{A}_n} p_n^{t-1}(a)q_a^{t-1}(a')$, and the last inequality is due to the assumption that $\eta \leq \frac{1}{16}$. Then, by invoking Lemma A.3, we have that

$$\begin{aligned}
\sum_{a \in \mathcal{A}_n} \|g_a^{t-1} - q_a^{t-1}\|_{q_a^{t-1}} &= \sum_{a \in \mathcal{A}_n} \|q_a^{t-1} - \operatorname*{arg\,min}_{g \in \text{relint}(\Delta(\mathcal{A}_n))} (-\Psi^{t-1}(g))\|_{q_a^{t-1}} \\
&\leq 2 \sum_{a \in \mathcal{A}_n} \|\nabla \Psi^{t-1}(q_a^{t-1})\|_{*,q_a^{t-1}} \\
&\leq 2 \sum_{a \in \mathcal{A}_n} \eta \|p_n^{t-1}(a)\hat{u}^{t-1} - p_n^{t-1}(a)m_n^{t-2}\|_{*,q_a^{t-1}} \\
&\leq 2 \sum_{a \in \mathcal{A}_n} \eta \frac{p_n^{t-1}(a)q_a^{t-1}(a_n^{t-1})}{p_n^{t-1}(a_n^{t-1})} + 2\eta \sum_{a \in \mathcal{A}_n} (p_n^{t-1}(a) - p_n^{t-2}(a)) \\
&\leq 2\eta + 0 \leq \frac{1}{8} < \frac{1}{4}.
\end{aligned}$$

□

Then, we are ready to give the proof of Lemma A.7 as follows.

Proof of Lemma A.7. Let $\mu_a^t := \max_{a' \in \mathcal{A}_n} \left| 1 - \frac{q_a^t(a')}{q_a^{t-1}(a')} \right|$. Then, it holds that

$$(\mu_a^t)^2 \leq \sum_{a' \in \mathcal{A}_n} \left(1 - \frac{q_a^t(a')}{q_a^{t-1}(a')} \right)^2 = \|q_a^t - q_a^{t-1}\|_{q_a^{t-1}}^2. \tag{A.32}$$

According to Lemma A.8, we have that $\sum_{a \in \mathcal{A}_n} \mu_a^t \leq 0.5$. By the definition of μ_a^t, we have that for any $a, a' \in \mathcal{A}_n$:

$$(1 - \mu_a^t) q_a^{t-1}(a') \leq q_a^t(a') \leq (1 + \mu_a^t) q_a^{t-1}(a').$$

By the definition of $\mathcal{T}_a$ in Definition A.3, we have that $\prod_{(u,v) \in E(\mathcal{T}_a)} Q_n^t(u,v) = \prod_{(u,v) \in E(\mathcal{T}_a)} q_u^t(v)$, and the above inequality implies that

$$\begin{aligned} &\prod_{u \in \mathcal{A}_n} (1 - \mu_u^t) \prod_{(u,v) \in E(\mathcal{T}_a)} q_u^{t-1}(v) \leq \prod_{(u,v) \in E(\mathcal{T}_a)} q_u^t(v) \\ &\leq \prod_{u \in \mathcal{A}_n} (1 + \mu_u^t) \prod_{(u,v) \in E(\mathcal{T}_a)} q_u^{t-1}(v). \end{aligned} \tag{A.33}$$

Since $\Sigma_a^t = \sum_{\mathcal{T} \in \mathbb{T}_a} \prod_{(u,v) \in E(\mathcal{T})} Q_n^t(u,v)$, summing up (A.33) over all $\mathcal{T} \in \mathbb{T}_a$ gives that

$$\Sigma_a^t \leq \Sigma_a^{t-1} \prod_{a' \in \mathcal{A}_n} (1 + \mu_a^t) \leq \Sigma_a^{t-1} \exp\left\{ \sum_{a' \in \mathcal{A}_n} \mu_{a'}^t \right\},$$

and that

$$\Sigma_a^t \geq \Sigma_a^{t-1} \prod_{a' \in \mathcal{A}_n} (1 - \mu_a^t) \geq \Sigma_a^{t-1} \exp\left\{ -2 \sum_{a' \in \mathcal{A}_n} \mu_{a'}^t \right\},$$

where the last inequality is due to $1 - x \geq e^{-2x}$ for any $x \in [0, \frac{1}{2}]$ and $\sum_{a \in \mathcal{A}_n} \mu_a^t \leq \frac{1}{2}$ due to Lemma A.8.

Since $\Sigma^t = \sum_{a \in \mathcal{A}_n} \Sigma_a^t$, we have the lower and upper bounds for Σ^t as follows:

$$\Sigma^{t-1} \exp\left\{ -2 \sum_{a' \in \mathcal{A}_n} \mu_{a'}^t \right\} \leq \Sigma^t \leq \Sigma^{t-1} \exp\left\{ \sum_{a' \in \mathcal{A}_n} \mu_{a'}^t \right\}.$$

Then, we have that

$$\begin{aligned} p_n^t(a) - p_n^{t-1}(a) &= \frac{\Sigma_a^t}{\Sigma^t} - \frac{\Sigma_a^{t-1}}{\Sigma^{t-1}} \\ &\leq \frac{\exp\left\{\sum_{a' \in \mathcal{A}_n} \mu_{a'}^t\right\} \Sigma_a^{t-1}}{\exp\left\{-2 \sum_{a' \in \mathcal{A}_n} \mu_{a'}^t\right\} \Sigma^{t-1}} - \frac{\Sigma_a^{t-1}}{\Sigma^{t-1}} \\ &= \frac{\Sigma_a^{t-1}}{\Sigma^{t-1}} \left(\exp\left\{3 \sum_{a' \in \mathcal{A}_n} \mu_{a'}^t\right\} - 1 \right) \\ &\leq \frac{\Sigma_a^{t-1}}{\Sigma^{t-1}} \left(8 \sum_{a' \in \mathcal{A}_n} \mu_{a'}^t \right), \end{aligned}$$

where the last inequality is due to $e^x - 1 \leq \frac{8}{3}x$ for all $x \in [0, \frac{3}{2}]$. Similarly, we have that

$$\begin{aligned} p_n^{t-1}(a) - p_n^t(a) &= \frac{\Sigma_a^{t-1}}{\Sigma^{t-1}} - \frac{\Sigma_a^t}{\Sigma^t} \\ &\leq \frac{\Sigma_a^{t-1}}{\Sigma^{t-1}} - \frac{\exp\left\{-2 \sum_{a' \in \mathcal{A}_n} \mu_{a'}^t\right\} \Sigma_a^{t-1}}{\exp\left\{\sum_{a' \in \mathcal{A}_n} \mu_{a'}^t\right\} \Sigma^{t-1}} \\ &= \frac{\Sigma_a^{t-1}}{\Sigma^{t-1}} \left(1 - \exp\left\{-3 \sum_{a' \in \mathcal{A}_n} \mu_{a'}^t\right\} \right) \\ &\leq \frac{\Sigma_a^{t-1}}{\Sigma^{t-1}} \left(3 \sum_{a' \in \mathcal{A}_n} \mu_{a'}^t \right), \end{aligned}$$

where the last inequality is due to $1 - x \leq e^{-x}$. Thus, by the fact that $p_n^{t-1} = \frac{\Sigma_a^{t-1}}{\Sigma^{t-1}}$, we obtain that

$$|p_n^t(a) - p_n^{t-1}(a)| \leq p_n^{t-1}(a) \left(8 \sum_{a' \in \mathcal{A}_n} \mu_{a'}^t \right).$$

Summing over $a \in \mathcal{A}_n$ gives $\|p_n^t - p_n^{t-1}\|_1 \leq 8 \sum_{a' \in \mathcal{A}_n} \mu_{a'}^t$.

Thus, by Cauchy-Schwarz inequality and (A.32), we have that

$$\begin{aligned} \|p_n^t - p_n^{t-1}\|_1^2 &\leq 64 \left(\sum_{a \in \mathcal{A}_n} \mu_a^t \right)^2 \leq 64 A_n \sum_{a \in \mathcal{A}_n} (\mu_a^t)^2 \\ &\leq 64 A_n \sum_{a \in \mathcal{A}_n} \|q_a^t - q_a^{t-1}\|_{q_a^{t-1}}^2. \end{aligned} \tag{A.34}$$

□

A.3.4 Proof of Theorem 4.2

Proof. **Step 1:** We express the swap regret bound in terms of the estimated reward as follows. Notice that

$$\begin{aligned}
&\mathbf{E}\left[\sum_{a\in\mathcal{A}_n}\left\langle q_a^t, p_n^t(a)\hat{u}_n^t\right\rangle \mid \mathcal{F}_{t-1}\right] = \sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n} p_n^t(a)q_a^t(a')m_n^t(a') \\
&- \sum_{a\in\mathcal{A}_n}\sum_{a'\in\mathcal{A}_n}\mathbf{E}\left[\frac{p_n^t(a)q_a^t(a')\mathbf{1}[a_n^t=a'](m_n^t(a')-u_n^t(a'))}{p_n^t(a')} \mid \mathcal{F}_{t-1}\right] \\
&= \sum_{a'\in\mathcal{A}_n} p_n^t(a')m_n^t(a') - \sum_{a'\in\mathcal{A}_n} p_n^t(a')(m_n^t(a')-u_n^t(a')) \\
&= \sum_{a\in\mathcal{A}_n} p_n^t(a)u_n^t(a),
\end{aligned}$$

and that

$$\begin{aligned}
&\mathbf{E}\left[\sum_{a\in\mathcal{A}_n} p_n^t(a)\hat{u}_n^t(F(a)) \mid \mathcal{F}_{t-1}\right] = \sum_{a\in\mathcal{A}_n} p_n^t(a)m_n^t(F(a)) \\
&- \sum_{a\in\mathcal{A}_n} p_n^t(a)(m_n^t(F(a))-u_n^t(F(a))) = \sum_{a\in\mathcal{A}_n} p_n^t(a)u_n^t(F(a)).
\end{aligned}$$

Then, the regret defined in (4.1) can be converted as follows:

$$\begin{aligned}
R_n^{\text{swa}}(T) &= \max_{F\in\mathcal{F}_n}\mathbf{E}\left[\sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\mathbf{1}[a_n^t=a]\left(u_n^t(F(a))-u_n^t(a)\right)\right] \\
&= \max_{F\in\mathcal{F}_n}\mathbf{E}\left[\sum_{t=1}^{T}\mathbf{E}\left[\sum_{a\in\mathcal{A}_n} p_n^t(a)\left(u_n^t(F(a))-u_n^t(a)\right] \mid \mathcal{F}_{t-1}\right]\right] \qquad \text{(A.35)} \\
&= \max_{F\in\mathcal{F}_n}\sum_{t=1}^{T}\sum_{a\in\mathcal{A}_n}\mathbf{E}\left[p_n^t(a)\hat{u}_n^t(F(a)) - \left\langle q_a^t, p_n^t(a)\hat{u}_n^t\right\rangle\right].
\end{aligned}$$

Step 2: Notice that for any $F \in \mathcal{F}_n$:

$$\mathbf{E}\left[\sum_{t=1}^{T}\left(p_n^t(a)\hat{u}_n^t(F(a)) - \left\langle q_a^t, p_n^t(a)\hat{u}_n^t\right\rangle\right)\right] \leq \max_{q\in\Delta(\mathcal{A}_n)} \mathbf{E}\left[\underbrace{\sum_{t=1}^{T}\left\langle q - q_a^t, p_n^t(a)\hat{u}_n^t\right\rangle}_{=:R_a^T(q)}\right].$$

Thus, we reduce the problem of bounding the swap regret to bounding external regret $R_a^T(q)$ for each subroutine $a \in \mathcal{A}_n$. This enables us to refine the existing full-information feedback analytical results to apply to our bandit feedback setting.

Since Lemma 2.9 requires that $q \in \text{relint}(\Delta(\mathcal{A}_n))$, but $q_a^\star \in \Delta(\mathcal{A}_n)$. Let $q_a^\star := \arg\max_{q\in\Delta_{\mathcal{A}_n}} \mathbf{E}[R_a^T(q)]$. We get around with this by letting $\tilde{q}_a := \left(1 - \frac{1}{T}\right) q_a^\star + \frac{1}{T} q_a^c$, where $q_a^c(a') = \frac{1}{A_n}, \forall a' \in \mathcal{A}_n$, and thus $\tilde{q}_a \in \text{relint}(\Delta(\mathcal{A}_n))$. Now we can write R_a^T as follows.

$$\begin{aligned}\mathbf{E}\left[R_a^T(q_a^*)\right] &= \mathbf{E}\left[\sum_{t=1}^{T}\left\langle q_a^\star - \tilde{q}_a, p_n^t(a)\hat{u}_n^t\right\rangle + \sum_{t=1}^{T}\left\langle \tilde{q}_a - q_a^t, p_n^t(a)\hat{u}_n^t\right\rangle\right] \\ &= \mathbf{E}\left[\sum_{t=1}^{T}\frac{1}{T}\left\langle q_a^\star - q_a^c, p_n^t(a)\hat{u}_n^t\right\rangle\right] + \mathbf{E}\left[R_a^T(\tilde{q}_a)\right].\end{aligned} \tag{A.36}$$

We can bound the first term of the RHS in the above equation as follows.

$$\begin{aligned}&\mathbf{E}\left[\sum_{t=1}^{T}\frac{1}{T}\left\langle q_a^\star - q_a^c, p_n^t(a)\hat{u}_n^t\right\rangle\right] = \mathbf{E}\left[\sum_{t=1}^{T}\frac{1}{T}\left\langle q_a^\star - q_a^c, p_n^t(a)u_n^t\right\rangle\right] \\ &\leq \mathbf{E}\left[\sum_{t=1}^{T}\frac{1}{T}\|q_a^\star - q_a^c\|_1\|p_n^t(a)u_n^t\|_\infty\right] \leq \mathbf{E}\left[\sum_{t=1}^{T}\frac{2}{T}p_n^t(a)\right],\end{aligned}$$

where the last inequality is due to that $\|q_a^\star - q_a^c\|_1 \leq \|q_a^\star\|_1 + \|q_a^c\|_1 \leq 2$ and $u_n^t \leq 1$.

Next, we can invoke Lemma 2.9 to bound $R_a^T(\tilde{q}_a)$, which requires $\eta\|p_n^t(a)\hat{u}_n^t - p_n^{t-1}(a)m_n^t\|_{*,q_a^t} \leq \frac{1}{8}$ and $\eta\|p_n^{t-1}(a)m_n^t\|_{*,g_a^{t-1}} \leq \frac{1}{2}$, where $g_a^t \in \text{relint}(\Delta(\mathcal{A}_n))$ is an auxiliary probability distribution defined as the solution when the predictor is exactly the estimated reward for the t-th round for subroutine $a \in \mathcal{A}_n$.

In the following, we show that the requirement for Lemma 2.9 is satisfied when

$\eta \le \frac{1}{16}$. According to (A.28) to (A.31), we have that

$$\begin{aligned}
&\eta \left\| p_n^t(a)\hat{u}_n^t - p_n^{t-1}(a)m_n^t \right\|_{*,q_a^t} \\
&\le \eta \left(\left\| p_n^t(a)\hat{u}_n^t - p_n^t(a)m_n^t \right\|_{*,q_a^t} + \left\| p_n^t(a)m_n^t - p_n^{t-1}(a)m_n^t \right\|_{*,q_a^t} \right) \\
&\le \eta \left(\frac{p_n^t(a)q_a^t(a_n^t)}{p_n^t(a_n^t)} + p_n^t(a) - p_n^{t-1}(a) \right) \le 2\eta \le \frac{1}{8},
\end{aligned} \tag{A.37}$$

where the third inequality is due to $\frac{p_n^t(a)q_a^t(a_n^t)}{p_n^t(a_n^t)} \le 1$ and $p_n^t(a) - p_n^{t-1}(a) \le 1$, and the last inequality is due to $\eta \le \frac{1}{16}$.

On the other hand, for any $q \in \Delta(\mathcal{A}_n)$, we have that

$$\eta \left\| p_n^{t-1}(a)m_n^t \right\|_{*,q} \le \eta \left\| p_n^{t-1}(a)m_n^t \right\|_{\infty} \le \eta \le \frac{1}{2}.$$

Then, we can invoke Lemma 2.9 to bound $R_a^T(\tilde{q}_a)$ as follows:

$$R_a^T(\tilde{q}_a) \le \frac{\psi(\tilde{q}_a)}{\eta} + 2\eta \sum_{t=1}^{T} \left\| p_n^t(a)\hat{u}_n^t - p_n^{t-1}(a)m_n^t \right\|_{*,q_a^t}^2 - \frac{1}{16\eta} \sum_{t=1}^{T} \left\| q_a^t - q_a^{t-1} \right\|_{q_a^{t-1}}^2. \tag{A.38}$$

Now, we give a bound of $\psi(\tilde{q}_a)$ by invoking Lemma 2.4 as follows.

$$\psi(\tilde{q}_a) \le \psi(q_a^c) + A_n \ln T = A_n \ln A_n + A_n \ln T \le 2A_n \ln T, \tag{A.39}$$

where we use the fact that $\pi(\tilde{q}_a; q_a^c) \le 1 - \frac{1}{T}$ by the definition of $\tilde{q}_a$ (see definition of $\pi(\cdot;\cdot)$ in (2.5)).

Now, substituting (A.38) and (A.39) into (A.36) gives:

$$\begin{aligned}
\mathbf{E}[R_a^t(q^\star)] \le \mathbf{E} \Bigg[&\frac{2}{T} \sum_{t=1}^{T} p_n^t(a) + \frac{2A_n \ln T}{\eta} + 2\eta \sum_{t=1}^{T} \left\| p_n^t(a)\hat{u}_n^t - p_n^{t-1}(a)m_n^t \right\|_{*,q_a^t}^2 \\
&- \frac{1}{16\eta} \sum_{t=1}^{T} \left\| q_a^t - q_a^{t-1} \right\|_{q_a^{t-1}}^2 \Bigg].
\end{aligned}$$

Step 3: Summing over $a \in \mathcal{A}_n$ and invoking Lemmas 4.1 and A.7, we have the

swap regret bounded as follows:

$$\sum_{a\in\mathcal{A}_n}\mathbf{E}[R_a^T(q_a^\star)] \le \mathbf{E}\left[2+\frac{2(A_n)^2\ln T}{\eta}+4\eta\sum_{t=1}^{T}\|u_n^t-u_n^{t-1}\|_1+4\eta\sum_{t=1}^{T}\|p_n^t-p_n^{t-1}\|_1 -\frac{1}{1024A_n\eta}\sum_{t=1}^{T}\left\|p_n^t-p_n^{t-1}\right\|_1^2\right].$$

Next, for any $t\in[T]$ and $a\in\mathcal{A}_n$, since

$$u_n^t(a)=\mathbf{E}_{a_{-n}\sim p_{-n}^t}[u_n(a;a_{-n}^t)]=\sum_{a_{-n}}p_{-n}^t(a_{-n})u_n(a;a_{-n}^t),$$

we have that

$$\begin{aligned}
&|u_n^t(a)-u_n^{t-1}(a)|\\
&=\left|\sum_{a_{-n}}p_{-n}^t(a_{-n})u_n(a;a_{-n})-\sum_{a_{-n}}p_{-n}^{t-1}(a_{-n})u_n(a;a_{-n})\right|\\
&\le\sum_{a_{-n}}|u_n(a;a_{-n})|\left|p_{-n}^t(a_{-n})-p_{-n}^{t-1}(a_{-n})\right|\\
&\le\sum_{a_{-n}}\left|\prod_{m\neq n}p_m^t[a_m]-\prod_{m\neq n}p_m^{t-1}[a_m]\right|\le\sum_{m\neq n}\left\|p_m^t-p_m^{t-1}\right\|_1,
\end{aligned}$$

where the last inequality is due to the total distance between two product distributions being bounded by the sum of the total variations of each marginal distribution [48]. Then, we have the swap regret bounded as follows:

$$R_n^{\mathsf{swa}}(T)\le\mathbf{E}\left[2+\frac{2(A_n)^2\ln T}{\eta}+4\eta\sum_{t=1}^{T}\sum_{m\in[N]}\left\|p_m^t-p_m^{t-1}\right\|_1 -\frac{1}{1024A_n\eta}\sum_{t=1}^{T}\left\|p_n^t-p_n^{t-1}\right\|_1^2\right].$$

□

Bibliography

[1] Soheil Abbasloo, Chen-Yu Yen, and H Jonathan Chao. Classic Meets Modern: A Pragmatic Learning-based Congestion Control for The Internet. In *Proc. ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication (SIGCOMM)*, pages 632–647, 2020.

[2] Jacob D Abernethy, Chansoo Lee, and Ambuj Tewari. Fighting Bandits with A New Kind of Smoothness. *Advances in Neural Information Processing Systems (NeurIPS)*, 28, 2015.

[3] Aditya Akella, Srinivasan Seshan, Richard Karp, Scott Shenker, and Christos Papadimitriou. Selfish Behavior and Stability of the Internet: A Game-Theoretic Analysis of TCP. *ACM SIGCOMM Computer Communication Review*, 32(4):117–130, 2002.

[4] Tansu Alpcan and Tamer Basar. A Game-Theoretic Framework for Congestion Control in General Topology Networks. In *Proc. IEEE Conference on Decision and Control (CDC)*, volume 2, pages 1218–1224. IEEE, 2002.

[5] Ioannis Anagnostides, Constantinos Daskalakis, Gabriele Farina, Maxwell Fishelson, Noah Golowich, and Tuomas Sandholm. Near-Optimal No-Regret Learning for Correlated Equilibria in Multi-Player General-Sum Games. In *Proc. Annual ACM Symposium on Theory of Computing (STOC)*, pages 736–749, 2022.

[6] Ioannis Anagnostides, Gabriele Farina, Christian Kroer, Chung-Wei Lee, Haipeng Luo, and Tuomas Sandholm. Uncoupled Learning Dynamics with $O(\log T)$ Swap Regret in Multiplayer Games. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 3292–3304, 2022.

[7] Grigorios G. Anagnostopoulos. Bayesian Games on A Maxmin Network Router. In *Proc. Mini-Conference on Applied Theoretical Computer Science (MATCOS)*, pages 29–34, 2010.

[8] Venkat Anantharam and Pantelis Tsoucas. A Proof of the Markov Chain Tree Theorem. *Statistics & Probability Letters*, 8(2):189–192, 1989.

[9] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The Non-Stochastic Multiarmed Bandit Problem. *SIAM Journal on Computing*, 32(1):48–77, 2002.

[10] Robert J Aumann. Subjectivity and Correlation in Randomized Strategies. *Journal of Mathematical Economics*, 1(1):67–96, 1974.

[11] Jakub Bielawski, Thiparat Chotibut, Fryderyk Falniowski, Grzegorz Kosiorowski, Michał Misiurewicz, and Georgios Piliouras. Follow-the-Regularized-Leader Routes to Chaos in Routing Games. In *Proc. International Conference on Machine Learning (ICML)*, volume 139, pages 925–935. PMLR, 18–24 Jul 2021.

[12] Avrim Blum and Yishay Mansour. From External to Internal Regret. *Journal of Machine Learning Research (JMLR)*, 8(6), 2007.

[13] Tomer Boyarski, Amir Leshem, and Vikram Krishnamurthy. Distributed Learning in Congested Environments with Partial Information, 2021.

[14] George W Brown. Some Notes on Computation of Games Solutions. Technical report, RAND Corp Santa Monica CA, 1949.

[15] Sébastien Bubeck, Yuanzhi Li, Yuval Peres, and Mark Sellke. Non-stochastic Multi-player Multi-Armed Bandits: Optimal Rate with Collision Information, Sublinear Without. In *Proc. Conference on Learning Theory (COLT)*, pages 961–987. PMLR, 2020.

[16] Swapna Buccapatnam, Jian Tan, and Li Zhang. Information Sharing in Distributed Stochastic Bandits. In *Proc. IEEE Conference on Computer Communications (INFOCOM)*, pages 2605–2613. IEEE, 2015.

[17] Yang Cai, Gabriele Farina, Julien Grand-Clément, Christian Kroer, Chung-Wei Lee, Haipeng Luo, and Weiqiang Zheng. Fast Last-Iterate Convergence

of Learning in Games Requires Forgetful Algorithms. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

[18] Yang Cai, Argyris Oikonomou, and Weiqiang Zheng. Finite-Time Last-Iterate Convergence for Learning in Multi-Player Games. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pages 33904–33919, 2022.

[19] Neal Cardwell, Yuchung Cheng, Soheil Hassas Yeganeh, Priyaranjan Jha, Yousuk Seung, Ian Swett, Victor Vasiliev, Bin Wu, and Matt Mathis Van Jacobson. BBR v2: A Model-based Congestion Control. Technical report, Montreal, 2019.

[20] Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

[21] Mithun Chakraborty, Kai Yee Phoebe Chua, Sanmay Das, and Brendan Juba. Coordinated versus Decentralized Exploration in Multi-Agent Multi-Armed Bandits. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, pages 164–170, 2017.

[22] Gong Chen and Marc Teboulle. Convergence Analysis of a Proximal-Like Minimization Algorithm using Bregman Functions. *SIAM Journal on Optimization*, 3(3):538–543, 1993.

[23] Xi Chen and Binghui Peng. Hedging in Games: Faster Convergence of External and Swap Regrets. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[24] Mung Chiang, S.H. Low, D. Wei, Ao Tang, Mung Chiang, S.H. Low, D. Wei, and Ao Tang. Heterogeneous Congestion Control: Efficiency, Fairness and Design. In *Proc. IEEE International Conference on Network Protocols (ICNP)*, pages 127–136, 2006.

[25] Christine Chung and Evangelia Pyrga. Stochastic Stability in Internet Router Congestion Games. In *Proc. International Symposium on Algorithmic Game Theory (SAGT)*, pages 183–195. Springer, 2009.

[26] Johanne Cohen, Amélie Héliou, and Panayotis Mertikopoulos. Learning with Bandit Feedback in Potential Games. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, pages 6372–6381, 2017.

[27] Michael B Cohen, Jonathan Kelner, John Peebles, Richard Peng, Anup B Rao, Aaron Sidford, and Adrian Vladu. Almost-Linear-Time Algorithms for Markov Chains and New Spectral Primitives for Directed Graphs. In *Proc. Annual ACM Symposium on Theory of Computing (STOC)*, pages 410–419, 2017.

[28] Pierre Coucheney, Bruno Gaujal, and Panayotis Mertikopoulos. Penalty-Regulated Dynamics and Robust Learning Procedures in Games. *Mathematics of Operations Research*, 40(3):611–633, 2015.

[29] Qiwen Cui, Zhihan Xiong, Maryam Fazel, and Simon S Du. Learning in Congestion Games with Bandit Feedback. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, pages 523–532, 2022.

[30] Yuval Dagan, Constantinos Daskalakis, Maxwell Fishelson, and Noah Golowich. From External to Swap Regret 2.0: An Efficient Reduction for Large Action Spaces. In *Proc. Annual ACM Symposium on Theory of Computing (STOC)*, pages 1216–1222, 2024.

[31] Constantinos Daskalakis, Maxwell Fishelson, and Noah Golowich. Near-Optimal No-Regret Learning in General Games. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, 2021.

[32] Michael Dinitz. Lecture 7: No-regret and equilibria. Lecture Notes for 601.436/636 Algorithmic Game Theory, Spring 2020, 2020. Accessed: 2025-08-27.

[33] Mo Dong, Tong Meng, Doron Zarchy, Engin Arslan, Yossi Gilad, Brighten Godfrey, and Michael Schapira. PCC Vivace: Online-Learning Congestion Control. In *Proc. USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, pages 343–356, 2018.

[34] Abhimanyu Dubey et al. Cooperative Multi-Agent Bandits with Heavy Tails. In *Proc. International Conference on Machine Learning (ICML)*, pages 2730–2739. PMLR, 2020.

[35] Pavlos S Efraimidis, Lazaros Tsavlidis, and George B Mertzios. Window-games between TCP Flows. *Theoretical Computer Science*, 411(31–33):2798–2817, 2010.

[36] Salma Emara, Baochun Li, and Yanjiao Chen. Eagle: Refining Congestion Control by Learning from The Experts. In *Proc. IEEE Conference on Computer Communications (INFOCOM)*, pages 676–685. IEEE, 2020.

[37] Salma Emara, Fei Wang, Baochun Li, and Timothy Zeyl. Pareto: Fair Congestion Control with Online Reinforcement Learning. *IEEE Transactions on Network Science and Engineering (TNSE)*, 2022.

[38] Eyal Even-Dar, Yishay Mansour, and Uri Nadav. On the Convergence of Regret Minimization Dynamics in Concave Games. In *Proc. Annual ACM symposium on Theory of Computing (STOC)*, pages 523–532, 2009.

[39] Gabriele Farina, Ioannis Anagnostides, Haipeng Luo, Chung-Wei Lee, Christian Kroer, and Tuomas Sandholm. Near-Optimal No-Regret Learning Dynamics for General Convex Games. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[40] Brion N Feinberg and Samuel S Chiu. A Method to Calculate Steady-State Distributions of Large Markov Chains by Aggregating States. *Operations Research*, 35(2):282–290, 1987.

[41] Xiaojie Gao, Kamal Jain, and Leonard J Schulman. Fair and Efficient Router Congestion Control. In *Proc. Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1050–1059, 2004.

[42] Rahul Garg, Abhinav Kamra, and Varun Khurana. A Game-Theoretic Approach Towards Congestion Control in Communication Networks. *ACM SIGCOMM Computer Communication Review*, 32(3):47–61, 2002.

[43] Sangtae Ha, Injong Rhee, and Lisong Xu. CUBIC: a New TCP-friendly High-speed TCP Variant. *ACM SIGOPS Operating Systems Review*, 42(5):64–74, 2008.

[44] Sergiu Hart and Andreu Mas-Colell. A Simple Adaptive Procedure Leading to Correlated Equilibrium. *Econometrica*, 68(5):1127–1150, 2000.

[45] Sergiu Hart and Andreu Mas-Colell. A Reinforcement Procedure Leading to Correlated Equilibrium. In *Economics Essays: A Festschrift for Werner Hildenbrand*, pages 181–200. Springer, 2001.

[46] David Hayes, David Ros, Lachlan L.H. Andrew, and Sally Floyd. Common TCP Evaluation Suite. Internet-Draft draft-irtf-iccrg-tcpeval-01, IETF, July 2014.

[47] Eshcar Hillel, Zohar S Karnin, Tomer Koren, Ronny Lempel, and Oren Somekh. Distributed Exploration in Multi-Armed Bandits. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 26, pages 854–862, 2013.

[48] Wassily Hoeffding and J Wolfowitz. Distinguishability of Sets of Distributions. *The Annals of Mathematical Statistics*, 29(3):700–718, 1958.

[49] Yu-Guan Hsieh, Kimon Antonakopoulos, and Panayotis Mertikopoulos. Adaptive Learning in Continuous Games: Optimal Regret Bounds and Convergence to Nash Equilibrium. In *Proc. Conference on Learning Theory (COLT)*, pages 2388–2422. PMLR, 2021.

[50] Zhiming Huang, Bingshan Hu, and Jianping Pan. Poster: Multi-agent Combinatorial Bandits with Moving Arms. In *Proc. International Conference on Distributed Computing Systems (ICDCS)*, pages 1140–1141. IEEE, 2021.

[51] Zhiming Huang, Bingshan Hu, and Jianping Pan. Gaussian Randomized Exploration for Semi-bandits with Sleeping Arms. In *NeurIPS 2024 Workshop on Bayesian Decision-making and Uncertainty*, 2024.

[52] Zhiming Huang and Jianping Pan. A Near-Optimal High-Probability Swap-Regret Upper Bound for Multi-Agent Bandits in Unknown General-Sum Games. In *Proc. Conference on Uncertainty in Artificial Intelligence (UAI)*, volume 216, pages 911–921. PMLR, 31 Jul–04 Aug 2023.

[53] Zhiming Huang and Jianping Pan. End-to-End Congestion Control as Learning for Unknown Games with Bandit Feedback. In *Proc. IEEE Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2023.

[54] Zhiming Huang and Jianping Pan. Distributed Learning of Unknown Games for HetNet Selection. *IEEE Transactions on Network Science and Engineering (TNSE)*, 2024.

[55] Zhiming Huang and Jianping Pan. Game-Theoretic Bandits for Network Optimization With High-Probability Swap-Regret Upper Bounds. *IEEE/ACM Transactions on Networking (TON)*, 2024.

[56] Zhiming Huang and Jianping Pan. Adversarial Semi-Bandits with Moving Arms. In *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, May 2025.

[57] Zhiming Huang and Jianping Pan. Faster Convergence for Unknown-Game Bandits. In *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, 2025.

[58] Shinji Ito. A Tight Lower Bound and Efficient Reduction for Swap Regret. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 18550–18559, 2020.

[59] Kevin Jamieson and Robert Nowak. Best-arm Identification Algorithms for Multi-Armed Bandits in the Fixed Confidence Setting. In *Proc. Annual Conference on Information Sciences and Systems (CISS)*, pages 1–6. IEEE, 2014.

[60] Nathan Jay, Noga Rotman, Brighten Godfrey, Michael Schapira, and Aviv Tamar. A Deep Reinforcement Learning Perspective on Internet Congestion Control. In *Proc. International Conference on Machine Learning (ICML)*, pages 3050–3059. PMLR, 2019.

[61] Chi Jin, Qinghua Liu, Yuanhao Wang, and Tiancheng Yu. V-Learning—A Simple, Efficient, Decentralized Algorithm for Multiagent RL. In *Proc. ICLR 2022 Workshop on Gamification and Multiagent Solutions*, 2022.

[62] Anna R Karlin and Yuval Peres. *Game Theory, Alive*, volume 101. American Mathematical Soc., 2017.

[63] Richard Karp, Elias Koutsoupias, Christos Papadimitriou, and Scott Shenker. Optimization Problems in Congestion Control. In *Proc. Annual Symposium on Foundations of Computer Science (FOCS)*, pages 66–74. IEEE, 2000.

[64] Tomáš Kocák, Gergely Neu, Michal Valko, and Rémi Munos. Efficient Learning by Implicit Exploration in Bandit Problems with Side Observations. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, pages 613–621, 2014.

[65] Ravi Kumar Kolla, Krishna Jagannathan, and Aditya Gopalan. Collaborative Learning of Stochastic Bandits over a Social Network. *IEEE/ACM Transactions on Networking (TON)*, 26(4):1782–1795, 2018.

[66] Walid Krichene, Benjamin Drighès, and Alexandre M Bayen. Online Learning of Nash Equilibria in Congestion Games. *SIAM Journal on Control and Optimization*, 53(2):1056–1081, 2015.

[67] James Kurose and Keith Ross. Computer Networks: A Top Down Approach Featuring the Internet, 2010.

[68] Bob Lantz, Brandon Heller, and Nick McKeown. A Network in A Laptop: Rapid Prototyping for Software-defined Networks. In *Proc. the ACM SIGCOMM Workshop on Hot Topics in Networks*, pages 1–6, 2010.

[69] Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms*. Cambridge University Press, July 2020.

[70] Renato Paes Leme, Georgios Piliouras, and Jon Schneider. Convergence of No-Swap-Regret Dynamics in Self-Play. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

[71] Wei Li, Fan Zhou, Kaushik Roy Chowdhury, and Waleed Meleis. QTCP: Adaptive Congestion Control with Reinforcement Learning. *IEEE Transactions on Network Science and Engineering (TNSE)*, 6(3):445–458, 2018.

[72] Xin Li, Qiuyuan Huang, and Dapeng Wu. A Repeated Stochastic Game Approach for Strategic Network Selection in Heterogeneous Networks. In *Proc. IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 88–93. IEEE, 2018.

[73] Michael L Littman. Markov Games as A Framework for Multi-Agent Reinforcement Learning. In *Proc. International Conference on Machine Learning (ICML)*, pages 157–163, 1994.

[74] Keqin Liu and Qing Zhao. Distributed Learning in Multi-Armed Bandit with Multiple Players. *IEEE Transactions on Signal Processing (TSP)*, 58(11):5667–5681, 2010.

[75] Luis López, Gemma del Rey Almansa, Stéphane Paquelet, and Antonio Fernández. A Mathematical Model for the TCP Tragedy of the Commons. *Theoretical Computer Science*, 343(1–2):4–26, 2005.

[76] David H Mguni, Yutong Wu, Yali Du, Yaodong Yang, Ziyi Wang, Minne Li, Ying Wen, Joel Jennings, and Jun Wang. Learning in Nonzero-Sum Stochastic Games with Potentials. In *Proc. International Conference on Machine Learning (ICML)*, volume 139, pages 7688–7699. PMLR, 18–24 Jul 2021.

[77] Ming Min and Ruimeng Hu. Signatured Deep Fictitious Play for Mean Field Games with Common Noise. In *Proc. International Conference on Machine Learning (ICML)*, volume 139, pages 7736–7747. PMLR, 18–24 Jul 2021.

[78] Ayush Mishra, Jingzhi Zhang, Melodies Sim, Sean Ng, Raj Joshi, and Ben Leong. Conjecture: Existence of Nash Equilibria in Modern Internet Congestion Control. In *Proc. Asia-Pacific Workshop on Networking (APNet)*, pages 37–42, 2021.

[79] John Nagle. On Packet Switches with Infinite Storage. *IEEE Transactions on Communications (TOC)*, 35(4):435–438, 1987.

[80] Akshay Narayan, Frank Cangialosi, Deepti Raghavan, Prateesh Goyal, Srinivas Narayana, Radhika Mittal, Mohammad Alizadeh, and Hari Balakrishnan. Restructuring Endpoint Congestion Control. In *Proc. the conference of the ACM Special Interest Group on Data Communication (SIGCOMM)*, pages 30–43, 2018.

[81] Heinrich H Nax, Maxwell N Burton-Chellew, Stuart A West, and H Peyton Young. Learning in a Black Box. *Journal of Economic Behavior & Organization*, 127:1–15, 2016.

[82] Arkadi Nemirovski. Interior Point Polynomial Time Methods in Convex Programming. *Lecture notes*, 42(16):3215–3224, 2004.

[83] Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*, volume 87. Springer Science & Business Media, 2013.

[84] Yurii Nesterov and Arkadii Nemirovskii. *Interior-Point Polynomial Algorithms in Convex Programming*. SIAM Studies in Applied Mathematics, 1994.

[85] Gergely Neu. Explore No More: Improved High-Probability Regret Bounds for Non-Stochastic Bandits. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 28, 2015.

[86] Duong D Nguyen, Hung X Nguyen, and Langford B White. Reinforcement Learning with Network-Assisted Feedback for Heterogeneous RAT Selection. *IEEE Transactions on Wireless Communications (TWC)*, 16(9):6062–6076, 2017.

[87] Francesco Orabona. A Modern Introduction to Online Learning. *arXiv preprint arXiv:1912.13213*, 2019.

[88] Gerasimos Palaiopanos, Ioannis Panageas, and Georgios Piliouras. Multiplicative Weights Update with Constant Step-Size in Congestion Games: Convergence, Limit Cycles and Chaos. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, pages 5872–5882, 2017.

[89] Christos Papadimitriou. Algorithms, Games, and the Internet. In *Proc. Annual ACM Symposium on Theory of Computing (STOC)*, pages 749–753, 2001.

[90] Binghui Peng and Aviad Rubinstein. Fast Swap Regret Minimization and Applications to Approximate Correlated Equilibria. In *Proc. Annual ACM Symposium on Theory of Computing (STOC)*, pages 1223–1234, 2024.

[91] Julia Robinson. An Iterative Method of Solving a Game. *Annals of Mathematics*, pages 296–301, 1951.

[92] Scott J Shenker. Making Greed Work in Networks: A Game-Theoretic Analysis of Switch Service Disciplines. *IEEE/ACM Transactions on Networking (TON)*, 3(6):819–831, 1995.

[93] João L Sobrinho, Roland De Haan, and José M Brazio. Why RTS-CTS is Not Your Ideal Wireless LAN Multiple Access Protocol. In *Proc. IEEE Wireless*

Communications and Networking Conference (WCNC), volume 1, pages 81–87. IEEE, 2005.

[94] Gilles Stoltz. *Incomplete Information and Internal Regret in Prediction of Individual Sequences*. PhD thesis, Université Paris Sud-Paris XI, 2005.

[95] Vasilis Syrgkanis, Alekh Agarwal, Haipeng Luo, and Robert E Schapire. Fast Convergence of Regularized Learning in Games. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 28, 2015.

[96] Balazs Szorenyi, Róbert Busa-Fekete, István Hegedus, Róbert Ormándi, Márk Jelasity, and Balázs Kégl. Gossip-based Distributed Stochastic Bandit Algorithms. In *Proc. International Conference on Machine Learning (ICML)*, pages 19–27. PMLR, 2013.

[97] Shaolin Tan, Zhihong Fang, Yaonan Wang, and Jinhu Lü. An Augmented Game Approach for Design and Analysis of Distributed Learning Dynamics in Multiagent Games. *IEEE Transactions on Cybernetics*, 2022.

[98] Ao Tang, Jiantao Wang, Steven H. Low, and Mung Chiang. Equilibrium of Heterogeneous Congestion Control: Existence and Uniqueness. *IEEE/ACM Transactions on Networking (TON)*, 15:824–837, 2007.

[99] Pratiksha Thaker, Matei Zaharia, and Tatsunori Hashimoto. Don't Hate the Player, Hate the Game: Safety and Utility in Multi-Agent Congestion Control. In *Proc. ACM Workshop on Hot Topics in Networks (HotNets)*, pages 140–146, 2021.

[100] Daniel Vial, Sanjay Shakkottai, and R Srikant. Robust Multi-Agent Multi-Armed Bandits. In *Proc. International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing (MobiHoc)*, pages 161–170, 2021.

[101] Weichen Wang, Jiequn Han, Zhuoran Yang, and Zhaoran Wang. Global Convergence of Policy Gradient for Linear-Quadratic Mean-Field Control/Game in Continuous Time. In *Proc. International Conference on Machine Learning (ICML)*, volume 139, pages 10772–10782. PMLR, 18–24 Jul 2021.

[102] Chen-Yu Wei and Haipeng Luo. More Adaptive Algorithms for Adversarial Bandits. In *Proc. Conference on Learning Theory (COLT)*, pages 1263–1291. PMLR, 2018.

[103] Keith Winstein and Hari Balakrishnan. TCP Ex Machina: Computer-generated Congestion Control. *ACM SIGCOMM Computer Communication Review*, 43(4):123–134, 2013.

[104] Keith Winstein, Anirudh Sivaraman, and Hari Balakrishnan. Stochastic Forecasts Achieve High Throughput and Low Delay over Cellular Networks. In *Proc. USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, pages 459–471, 2013.

[105] Zhenchang Xia, Yanjiao Chen, Libing Wu, Yu-Cheng Chou, Zhicong Zheng, Haoyang Li, and Baochun Li. A Multi-Objective Reinforcement Learning Perspective on Internet Congestion Control. In *Proc. IEEE/ACM International Symposium on Quality of Service (IWQOS)*, pages 1–10. IEEE, 2021.

[106] Qiaomin Xie, Zhuoran Yang, Zhaoran Wang, and Andreea Minca. Learning While Playing in Mean-Field Games: Convergence and Optimality. In *Proc. International Conference on Machine Learning (ICML)*, volume 139, pages 11436–11447. PMLR, 18–24 Jul 2021.

[107] Francis Y Yan, Jestin Ma, Greg D Hill, Deepti Raghavan, Riad S Wahby, Philip Levis, and Keith Winstein. Pantheon: The Training Ground for Internet Congestion-control Research. In *Proc. USENIX Annual Technical Conference (ATC)*, pages 731–743, 2018.

[108] Martin Zinkevich. Online Convex Programming and Generalized Infinitesimal Gradient Ascent. In *Proc. International Conference on Machine Learning (ICML)*, pages 928–936, 2003.