Online Learning in Games: Clairvoyance, Convergence and Recurrence

Submitted by

Ryann SIM Wei Jian

Thesis Advisor

Georgios PILIOURAS

Engineering Systems and Design

A thesis submitted to the Singapore University of Technology and Design in fulfillment of the requirement for the degree of Doctor of Philosophy

2024

PhD Thesis Examination Committee

TEC Chair:	Prof. Rakesh Nagi
Main Advisor:	Prof. Georgios Piliouras
Internal TEC member 1:	Asst. Prof. Antonios Varvitsiotis
Internal TEC member 2:	Prof. Dario Poletti

Abstract

Engineering Systems and Design

Doctor of Philosophy

Online Learning in Games: Clairvoyance, Convergence and Recurrence

by Ryann SIM Wei Jian

While the standard paradigm of non-cooperative game theory focuses on centralized equilibrium analysis and computation, online learning in games is a setting where players employ simple learning dynamics to update their strategies over time. The primary aim of research in this direction is twofold: to either characterize the day-to-day behavior of the dynamics, or conversely to establish convergence results to game-theoretic equilibria. In the former, research into the day-to-day behavior of broad classes of continuous-time dynamics has shown that they exhibit cyclic, and formally Poincaré recurrent, behavior in zero-sum games. However, the robustness of this behavior has been less studied in non-standard game-theoretic settings. In the latter, the focus is primarily on establishing fast theoretical convergence rates of discrete-time algorithms to relevant notions of equilibria. One recent success in this direction has been the introduction of 'optimistic' variants of standard online learning algorithms, which speed up convergence while avoiding cyclical behavior. However, it is unclear how widely applicable optimism is, or if there are further modifications that outperform it.

In this dissertation, we tackle both of these challenges in various non-cooperative game theoretic settings. First, we establish Poincaré recurrence results of the ubiquitous Follow-The-Regularized-Leader dynamics (and variations thereof) in formulations of time-evolving games and quantum games. These types of games allow us to establish rich connections to the realms of evolutionary biology and quantum information theory respectively. Along the way, we also show that in certain formulations of time-evolving games, time-average convergence to equilibrium actually fails, elucidating a new failure mode of learning dynamics in games. Next, we propose and analyze discrete-time dynamics that can achieve fast convergence to equilibria in multi-player extensive-form games, as well as general-sum normal-form games with an arbitrary number of players and strategies. We establish that in multi-player extensive-form games which admit a network structure, optimistic gradient ascent converges to the set of Nash equilibria at a linear rate. Going beyond optimism, we then present a novel class of 'clairvoyant' learning algorithms for learning in general-sum games, which improves upon the state-of-the-art convergence rate to the set of coarse correlated equilibria. In their totality, the results in this dissertation represent a step forward in better understanding the recurrent behavior of continuous-time learning dynamics in zero-sum games, while also providing encouraging equilibrium convergence results in the algorithmic, discrete-time context.

Publications

(1) Stratis Skoulakis, Tanner Fiez, **Ryann Sim**, Georgios Piliouras, and Lillian Ratliff. **Evolutionary Game Theory Squared: Evolving Agents in Endogenously Evolving Zero-Sum Games**. In *AAAI Conference on Artificial Intelligence*, 2021.

(2) Tanner Fiez, **Ryann Sim**, Stratis Skoulakis, Georgios Piliouras, and Lillian Ratliff. **Online Learning in Periodic Zero-Sum Games**. *Advances in Neural Information Processing Systems*, 34, 2021.

(3) **Ryann Sim**, Stratis Skoulakis, Lillian J Ratliff, and Georgios Piliouras. **Fast Converg- ence of Optimistic Gradient Ascent in Network Zero-Sum Extensive Form Games**. In *International Symposium on Algorithmic Game Theory*. Springer, 2022.

(4) Georgios Piliouras, **Ryann Sim**, and Stratis Skoulakis. **Beyond Time-Average Convergence: Near-Optimal Uncoupled Online Learning via Clairvoyant Multiplicative Weights Update**. *Advances in Neural Information Processing Systems*, 35, 2022.

(5) Rahul Jain, Georgios Piliouras, and **Ryann Sim**. **Matrix Multiplicative Weights Updates in Quantum Zero-Sum Games: Conservation Laws & Recurrence**. *Advances in Neural Information Processing Systems*, 35, 2022.

(6) Wayne Lin, Georgios Piliouras, **Ryann Sim**, and Antonios Varvitsiotis. **Quantum Potential Games, Replicator Dynamics, and the Separability Problem**. Submitted to Quantum Journal, 2023.

(7) Volkan Cevher, Georgios Piliouras, **Ryann Sim**, and Stratis Skoulakis. **Min-Max Optimization Made Simple: Approximating the Proximal Point Method via Contraction Maps**. *SIAM Symposium on Simplicity in Algorithms*, 2023.

(8) Wayne Lin, Georgios Piliouras, **Ryann Sim**, and Antonios Varvitsiotis. **No-Regret Learning and Equilibrium Computation in Quantum Games**. Submitted to Quantum Journal, 2024.

Acknowledgements

The work in this dissertation would not have been possible without the support and kindness of many great people. No one exemplifies the courage and drive to explore new ideas more than my PhD advisor, Georgios Piliouras. In my four and a half years in the program, I have learned so much from you and your expertise in various areas of game theory and theoretical computer science. You have also shown me how to be persistent and excited in tackling new and oftentimes difficult problems, and I am so very grateful for the opportunity to do research with you. Most importantly, your compassion when dealing with people has been truly inspirational, and I am truly thankful for your understanding and support even when I was facing health problems.

I would also like to thank my thesis examination committee, Drs. Rakesh Nagi, Antonios Varvitsiotis, Dario Poletti and Stefano Galelli (prior to leaving SUTD), who have given me many helpful comments during my preliminary examination and beyond. During my time in the program, I was fortunate to have been part of several fruitful collaborations. Working with Stratis Skoulakis, Lillian Ratliff and Tanner Fiez in the first years of my PhD was an absolute joy, giving me a firm foundation upon which to build my research identity. In the remainder of my PhD, collaborating with Rahul Jain, Antonios Varvitsiotis and Wayne Lin opened my eyes to a new paradigm beyond normal-form games. I was also fortunate to have spent time in Shanghai with Wang Xiao and Feng Yi, with whom I was able to begin exploring ideas at the very frontier of our field. I am also grateful for the many discussions and hangouts with fellow grad students at SUTD, especially Iosif, Wayne, Lin Geng and the members of the GSA exco.

Many others have made my time in the PhD programme at SUTD so much richer. It was always a blast interacting with the members of Georgios' group: Stefanos, Stratis, Xiao, Marco, Shuyue, Jinxing, Barnabe and Sai among others. The ESD admin staff, particularly Lee Chen and Sin Chee, were always so kind and helpful, really making my PhD journey feel so smooth. The grad level courses I took under Profs. Shaowei, Bikram, Karthyek, Ioannis, and Antonios were intellectually stimulating and also just plain fun. Finally, I was fortunate to teach undergrad classes alongside Profs. Karthyek and Nuno, who both inspired me to strive to be a better educator.

As an undergrad at SUTD, I was fortunate to have met Profs. Lingjie, James, Selin, Wei Pin, Michael, Nazry and Rhema, who each in their own way inspired me to pursue what I was interested in fearlessly. My comrades in freshmore, ESD and all my fifth rows helped me learn so much beyond the confines of the classroom. A shout out to the F09 gang, Jing Yu, Kyra, Tea, Joel (Huang) and Joel (Tan) for your companionship over the years.

Outside of school, I've been fortunate to have met so many amazing people over the years. To the inhabitants of Changi Court: Samuel, Vincent, Zhiyuan and Nath, thank you for making the COVID lockdown far more tolerable, and for caring for me when I was ill. To the SSH gang: Yus, Sakshi, Weilee, Winnie, Shubham, Ayesha, Fabio and every other soul who entered our hallowed halls, thank you for the lovely conversations and fun times we had. To the council of AfterEight: Raj and Joel, thank you for building a new (rented) home alongside me. To the members of Aquilo Ventum and associated

adventuring parties, no words can express my appreciation for all the laughter and tears we shared over the years. To the SoDa Players and One Chamber Choir, thank you for letting me express myself and for all the lovely music we made together. And of course, thank you to the Yessers for yessing. Yes.

My journey in academia would have been impossible if not for my family and friends back home in Malaysia, who have always provided me with unending support. Mom and Dad, thank you for always letting me explore my weird scientific interests, and working so hard to provide for us. Aaron, Sarah, thank you for making my childhood one that can I look back upon fondly. Yeye, Ahma, Mama, Ee Ee May, Goh Ee, Ah Ku, thank you for taking such good care of me. To all my teachers and friends from back home, I am forever grateful for your encouragement and counsel over the years. There is no one I'm more lucky to have met than Michelle. Thank you for always being by my side through the last few years, from Singapore to Shanghai and back again. I would do it all over again without any hesitation. Finally, thank you to Bucky (my cat) who screams and eats a lot. Thank you for helping proof-read my dissertation.

Contents

List of Figures

List of Tables

List of Acronyms

AI	Artificial Intelligence
CCE	Coarse Correlated Equilibrium
CMWU	Clairvoyant Multiplicative Weights Update
EFG	Extensive-Form Game
EGT	Evolutionary Game Theory
FTL	Follow-The-Leader
FTRL	Follow-The-Regularized-Leader
GAN	Generative Adversarial Network
GDA	Gradient Descent Ascent
KL	Kullback-Leibler (Divergence)
LP	Linear Program
MD	Mirror Descent
ML	Machine Learning
MMWU	Matrix Multiplicative Weights Update
MP	Matching Pennies
MRD	Matrix Replicator Dynamics
MWU	Multiplicative Weights Update
NE	Nash Equilibrium
ODE	Ordinary Differential Equation
OGA	Optimistic Gradient Ascent
OGD	Optimistic Gradient Descent
OMD	Optimistic Mirror Descent
OMWU	Optimistic Multiplicative Weights Update
PD	Prisoner's Dilemma
POVM	Positive Operator-Valued Measurement
PPAD	Polynomial Parity Arguments on Directed Graphs
PSD	Positive Semi-Definite
PTAS	Polynomial Time Approximation Scheme
QRE	Quantum Relative Entropy
RD	Replicator Dynamics
RPS	Rock-Paper-Scissors

To my family, and those I lost along the way

Chapter 1

Introduction

Analyzing how groups of agents learn and adapt over time is a fundamental aspect of theoretical research, with a multitude of applications. An economist might seek to understand how markets shift over time and how different firms choose to adapt to these changes. A biologist might study the flocking of birds as they migrate, observing and quantifying how their complex formations minimize physical exertion in their extended journeys. An urban scientist might monitor traffic in a city, finding methods to reduce congestion during the morning rush. More recently, with the advent of the Internet and machine learning, studying the behavior of users of a social network, groups of online auction bidders and even autonomous robots learning how to play football have become core applications of multi-agent systems [156].

These applications are linguistically simple to describe but inherently complex when viewed from a theoretical lens: how can one model these situations in a way which is amenable to theoretical analysis? Indeed, there is a further question of what the analysis entails – do we wish to understand and quantify the behavior of the agents over time, or do we also wish to design learning processes that lead to desirable states for the agents? One framework that is able to provide insight into both the above questions is *learning in games* [35, 65].

In their most fundamental form, games are a distillation of the interactions between rational, utility-maximizing agents and the payoffs they obtain for selecting certain courses of action. The theory of games, and its connections to economic theory, was introduced in the seminal work of von Neumann and Morgenstern [174]. The equilibria of games can often be thought of as solutions to an accompanying learning problem — for instance, John Nash introduced the concept of a Nash equilibrium [128], wherein game players have no incentive to unilaterally deviate to any other strategy. A particularly effective technique for studying many problems in real-life arises by first modeling the system as a multi-agent game, and then using game-theoretic intuition and techniques to understand how agents act and learn within the game. For instance, in the realm of evolutionary biology, the diversity of a set of populations of organisms can be modeled as a non-transitive, rock-paper-scissors type game [114, 137, 179]. In machine learning, principal component analysis has been framed as the Nash equilibrium of a competitive game, allowing for scalable analysis of the problem [67].

Classical results in game theory have primarily been focused on centralized equilibrium computation and characterizing the equilibria (most notably Nash equilibria) in various

sub-classes of games [8, 128, 174]. Essentially, game theory in its most fundamental form aims to prescribe to agents what they should do with their available information, so that the system as a whole is maximally efficient. However, there are two major conceptual difficulties with this perspective. Firstly, there is the question of whether or not rational players of the game will actually choose to play using equilibrium strategies. This problem is compounded if the game admits multiple equilibria, where it becomes less clear to the players what equilibrium they should actually be playing [77]. Second, each player computing an equilibrium in isolation could itself be intractable. For instance, computing Nash equilibria beyond even the most simple two-player settings has been shown to be formally hard [37, 48]. In larger scale games with many possible strategies, it is thus unreasonable for expect players to analytically compute an optimal equilibrium and subsequently play it. As renown mathematician Stephen Smale boldly claimed in [161]: "the theory has not successfully confronted the question, 'How is equilibrium reached?'"

An alternative perspective, and one which we adopt in this dissertation, is to frame games as models of conflict and cooperation in a 'natural' process where agents fumble for better strategies, eventually arriving at a suitable notion of equilibrium. In this framework, 'natural' learning rules/algorithms can often be constructed by observing the intrinsic structure of the problem, and are typically designed to be run in an distributed and computationally inexpensive manner. The objective then becomes studying what equilibria are converged to by players using these learning rules, if they converge to an equilibrium at all. This approach thus allows us to gain a better understanding of how systems of distributed agents might actually learn in a potentially unfamiliar environment. Moreover, it allows us to borrow ideas and results from the realm of online learning, and more specifically *online optimization* [30, 80], and thus analyze performance guarantees and convergence rates of these learning rules.

This learning perspective is not an entirely modern paradigm – learning in games has been studied extensively from the very conception of the theory of games. One of the first learning rules is known as fictitious play [26, 143], which was introduced as a method to rationalize the Nash equilibrium. In fictitious play, players observe only their own payoffs and subsequently play a best response to the opponent's empirical frequency of play. Another well-studied dynamic is the replicator dynamic [151, 166], which is typically studied in evolutionary game theory [84, 116]. When using replicator dynamics, the frequency of a strategy within a population shifts over time based on the difference between its expected payoff and the population's average payoff. The natural interpretation of this payoff in the evolutionary game theoretic context is the 'fitness' of the strategy. These learning rules (and many others in the literature) are designed with the hope that players utilizing them eventually converge to some notion of equilibrium. Indeed, it is often necessary to study the limiting behavior of these learning rules in order to better understand what equilibrium is converged to, as well as the stability of said equilibrium [65].

A key application area for the learning in games framework has been in machine learning, where agents are typically required to learn to make better decisions in unfamiliar settings with limited data. As the complexity of machine learning systems grows, each agent in such a system has to make effective decisions while also interacting with large

groups of other potentially conflicting or even adversarial agents. Some of the greatest success stories in recent machine learning research have had a learning in games flavor. The advent of generative adversarial networks [74] has seen strong empirical results in various problem domains. The fundamental model of a GAN sees a repeated zero-sum game setting where a generator creates samples from an unknown target distribution, while a discriminator attempts to classify the samples as real or generated. However, the GAN training process is notoriously difficult, often leading to mode collapse where the system either gets stuck in a suboptimal state or cycles between several suboptimal states [61, 147]. By framing the GAN training process as learning equilibria in a zero-sum game, recent work has managed to provide intuition into why this happens, while also improving the stability of the GAN training process [95, 117, 173].

Learning in games is also useful for finding solutions to large scale real-life games, the most famous examples of which being extensive-form games (i.e. games where players make moves sequentially) such Poker and Go. Importantly, games like these are able to capture imperfect information, where players have to make decisions under uncertainty. Recent innovations have led to superhuman performance of algorithms based upon online learning within these games ([23, 28] for Poker and [158] for Go respectively), showing the empirical efficacy of the framework of learning in games.

The Landscape of Learning in Games

A fundamental aspect to learning in games beyond practical applications is exploring the theory behind the empirical success of this framework. As an indicative example, let us consider the fictitious play method described earlier, which seems like an intuitive way to converge to Nash equilibria. Indeed, [143] showed that if players use fictitious play dynamics in zero-sum games, the product of the empirical distribution of their strategies converges to a Nash equilibrium. There are however several caveats to this result: first, this convergence could potentially be exponentially slow [50], and second, in some games beyond zero-sum, fictitious play might not even converge to the Nash [155].

How can we mitigate these potential issues with convergence? An elegant and well-studied solution is to utilize the notion of no-regret learning, which arises as a marriage between several seminal works in online learning and game theory [19, 76, 78]. To understand how no-regret learning works, we first must introduce a key connection between online learning and game theory: the concept of 'regret', a notion which can be defined in both discrete- and continuous-time. The (external) regret of an algorithm or learning dynamic is intuitively the difference between the actual payoffs obtained by running the dynamic and the maximum possible payoff from selecting a fixed action in hindsight. A dynamic is said to have 'no-regret' if its time-average regret vanishes over time. One of the most well-known no-regret dynamics is the Multiplicative Weights Update (`MWU`), also known in the literature as Hedge or exponential weights [6, 110]. Indeed, `MWU` is the discrete-time counterpart of replicator dynamics, and also arises as a special case of the class of learning rules called Follow-The-Regularized-Leader (`FTRL`) [154].

Crucially, the time-averaged strategies for a player obtained via no-regret dynamics such as MWU converge to a superset of Nash equilibria called coarse correlated equilibria (CCE) [35, 145]. Moreover, in the special case of zero-sum games, the time-average strategies of no-regret play converge to a Nash equilibrium! Now, let us consider the rates of convergence of no-regret learning. If players of a general-sum game experience regret of at most $\epsilon(T)$, then it is known that their time-average strategies converge with rate $\mathcal{O}(\epsilon(T)/T)$ to a CCE. The question then becomes, what regret bound do standard no-regret dynamics exhibit in general? As it turns out, in the adversarial setting where an adversary can select the payoff/cost vector *after* the player has decided upon their mixed strategy, important learning dynamics such as MWU and FTRL experience an expected regret of $\mathcal{O}(\sqrt{T})$ [35]. Hence, if all players of a game utilize the above dynamics, they converge in time-average to CCE at a rate of $\mathcal{O}(1/\sqrt{T})$. Finally, if the game is zero-sum, the dynamics instead converge to Nash equilibria at the same rate. Notice that these rates are obtained in an adversarial setting. In a game theoretic setting where the game matrix/payoff vectors are predetermined, a long line of research has shown that players can actually converge faster by employing variations of standard no-regret algorithms. For instance, 'optimistic' variants of MWU have been shown to converge in time-average to CCE at faster and faster rates [47, 141, 164].

Hence, we have seen that no-regret learning manages to circumvent the issues that may arise when utilizing simple learning rules. If this were the full picture of learning in games, then a confluence between equilibrium and evolution would have been established, allowing for Nash equilibria and relaxations thereof to be considered 'natural' predictors for the long-term behavior of no-regret learning in myriad multi-agent systems. However, complications arise when instead of studying the *time-average* behavior of these dynamics, we instead observe their *day-to-day* behavior.

In the discrete-time case, the day-to-day behavior of MWU and more generally FTRL has been shown to diverge away from the Nash equilibrium when applied to two-player zero-sum games [12, 13, 40]. In the continuous-time case, [149] showed that no-regret algorithms can exhibit formally chaotic behavior even in two-player zero-sum games. Moreover, a string of recent results have shown that a notion of cyclical structure can still be captured in many classes of zero-sum games [11, 20, 117, 138]. This property is known as Poincaré recurrence, a dynamical property which certifies that for any initial condition, the dynamics return infinitely often to the initial condition. From the perspective of equilibration, it is clear then that even in the simplest of games, there exist online learning dynamics in both discrete- and continuous-time that effectively never actually converge to an equilibrium in the day-to-day sense.

From an algorithmic perspective, it is thus useful to seek out both recurrence (failure modes) and fast convergence to equilibria (time-average or last-iterate) as descriptors for the behavior of specific learning dynamics when applied to games. In doing so, we provide a clearer picture of the landscape of learning in games, providing theoretical insight into why certain ideas may or may not work in practice.

All of the above motivates the following two key questions, which we seek to explore in this dissertation:

(1) *Are there interesting game-theoretic settings for which recurrent behavior of online learning dynamics can be proven?*

(2) *Can we design dynamics that can guarantee fast convergence to relevant notions of equilibria in games? How fast do they converge to these equilibria?*

Notice that these questions are framed with the examples of GANs and large-scale game solving in mind. In the former, the objective is to understand the high-level (i.e. qualitative) dynamical behavior of the day-to-day dynamics in games, as well elucidating potential failure modes of the dynamics. In the latter, the focus is on finding practical dynamics that converge to equilibria as quickly as possible in games with potentially large numbers of players. To tackle these questions, we focus on two variables within learning in games which follow a rich line of inquiry. The first variable is the types of games we study. Rather than focusing on a particular class of games, we instead focus on different subclasses of zero-sum or general-sum games, often in a multi-agent context. The games we study provide a method of broadly understanding learning and adversarial behavior in biology, economics and even quantum systems.

The second variable we focus on is the learning process which game players utilize. In particular, we study broad classes of no-regret algorithms and examine their dynamical and convergent behavior when applied to various games. Moreover, we study a mixture of continuous-time dynamics and discrete-time algorithms. While discrete-time algorithms are more amenable to computer science/machine learning applications, continuous-time dynamics such as replicator dynamics are closely related to evolutionary biology, further concretizing games as an abstract model that can be applied to multiple areas of scientific research. Moreover, there are many strong connections between different learning rules in the literature. For instance, `MWU` is actually a discrete-time counterpart to replicator dynamics. In other words, by taking the continuous-time limit one can obtain replicator dynamics from `MWU`.

By tuning and manipulating these variables, we seek to answer our research questions by pairing up classes of games with relevant or natural online learning dynamics as follows:

(1) **Time-Evolving Games.** Many existing results for learning in games focus on static games where the rules of interaction between players are predetermined for all time. However, many real-life systems change over time, and the behavior of learning rules in these game formulations is less well-understood. In particular, the time-evolution can take the form of endogenous evolution (i.e. the game evolves as a function of the players' strategies) or exogenous evolution (i.e. the game evolves over time, irrespective of what the players do). As is typical in evolutionary biology, we study replicator dynamics and generalizations thereof in this class of games.

(2) **Quantum Games.** Various notions of quantum games where players control and exchange quantum information have been proposed [21, 55, 186]. However, much like in time-evolving games, little is known about online learning in this class of games. As it turns out, a matrix extension of MWU has been studied in this class of games, showing time-average convergence to the Nash equilibrium [90, 91]. To

better understand the dynamical behavior of learning in quantum games, we seek to formulate and understand a matrix extension of replicator dynamics in these games.

(3) **Multiplayer Extensive-Form and General-Sum Games.** Extensive-form and general-sum games have been studied extensively in the literature. Of particular interest are games with multiple players who seek to converge to equilibria as quickly as possible. This setting has applications in reinforcement learning, large-scale game solving and beyond. We focus on discrete-time algorithms that are able to converge quickly to equilibria either in the time-average or the last-iterate sense, inspired by recent modifications to MWU such as 'optimistic' MWU [141, 164], where players update their strategies with a recency bias (i.e. more emphasis is placed on payoffs obtained in the most recent iterate).

1.1 Contributions and Organization

In Chapter 2, we present the necessary preliminaries for game theory and online learning in games. In particular, we focus on non-cooperative games spanning general-sum normal form games, zero-sum polymatrix games and zero-sum extensive form games. We introduce several important learning dynamics for these games, alongside relevant recurrence and convergence results from the literature.

In Chapter 3, we study continuous-time dynamics in endogenously-evolving zero-sum games. For a model of endogenous evolution which has roots in evolutionary biology, we propose a novel reduction thereof to a static network polymatrix game (Chapter 3.3). By analyzing the static polymatrix game, we are able to derive Poincaré recurrence results for a wide class of endogenously-evolving games when players utilize replicator dynamics (Chapter 3.4). Moreover, we show that the time-average behavior and utility of the players converge to the time-average Nash equilibrium of the game (Chapter 3.5).

In Chapter 4, we study continuous-time dynamics in exogenously-evolving zero-sum games. Specifically, we study games where the payoffs evolve periodically over time, while still maintaining a time-invariant Nash equilibrium. We show that Poincaré recurrence still holds when players use continuous-time gradient descent-ascent `GDA` in periodic zero-sum bilinear games (Chapter 4.3) and follow-the regularized-leader in periodic zero-sum polymatrix games (Chapter 4.4). However, we also show that the time-average behavior of the dynamics might not converge to the time-invariant equilibrium in the respective classes of time-evolving games (Chapters 4.3.2 & 4.4.2). This negative result reveals a new failure mode for learning in games, showing that intuition from static games might totally fail in the time-evolving regime.

In Chapter 5, we study discrete- and continuous-time learning in quantum zero-sum games. Recent work has shown that a matrix variation of multiplicative weights update converges in time-average to Nash equilibria in quantum zero-sum games. We formulate the continuous-time variant of this algorithm, which we call Matrix Replicator Dynamics (`MRD`), and show that it exhibits Poincaré recurrence in two-player quantum zero-sum games (Chapter 5.5). Along the way, we show that the quantum relative entropy of the

players within the system is a constant of motion (Chapter 5.4). This result represents an initial foray into continuous-time learning in quantum games.

In Chapter 6, we study discrete-time 'optimistic' gradient descent-ascent (OGA) in network zero-sum extensive-form games. We first formulate this new class of extensive-form games, which have clear applications to real-life multiplayer systems (Chapter 6.3). We then show that the dynamics converge to Nash equilibrium in time-average as well as the last-iterate sense at a linear rate which is dependent on the game (Chapter 6.4). In order to establish these results, we frame the multi-player zero-sum extensive-form game as a two-player symmetric game so that a Nash equilibrium in one formulation is also a Nash equilibrium in the other. We then prove the last iterate convergence in this reduced two-player symmetric game.

In Chapter 7, we propose a novel discrete-time online learning algorithm which is able to achieve state-of-the-art performance in general-sum games. Specifically, we describe and analyze a variant of Multiplicative Weights Update which we call Clairvoyant Multiplicative Weights Update (CMWU) (Chapter 7.4). In its original form, players using CMWU select their strategies based on the behavior of everyone in the next time-step, i.e. the players are clairvoyant. However, actually implementing CMWU requires players to solve a fixed point problem at every time-step, which could result in dynamics that are not computationally efficient. To circumvent this, we introduce an uncoupled and efficient algorithm called CMWU dynamics (Chapter 7.5), which admit an $\mathcal{O}(\log(T))$-sparse subsequence that converges to CCE at a rate of $\Theta(\log(T)/T)$, improving upon the previous state-of-the-art of $\Theta(\log^4(T)/T)$.

Finally, in Chapter 8 we provide some perspectives on future work and potential applications of our research, while the Appendix contains omitted proofs from the main text and additional simulations.

Chapter 2

Preliminaries: Online Learning in Games

And you may ask yourself, "Am I right, am I wrong?"

Talking Heads, *Once in a Lifetime*

Much of the work contained within this dissertation is built upon the tools and ideas of far brighter minds, whose works have impacted many fields of science, economics and computing. These works span game theory, online learning, dynamical systems and even quantum theory. Due to the breadth and richness of the connections between these seemingly disparate yet highly interconnected areas, in this chapter we will focus on a game-theoretic perspective with which to introduce the necessary concepts for our work.

This chapter is designed to be a very high-level overview of learning in games, and thus will be far from a comprehensive treatment of the entire field. For a more complete picture, the reader is referred to the seminal works of [35, 80, 154, 156]. Within each of the subsequent chapters of this dissertation, any additional notation will be provided as necessary, so that each chapter can be read in isolation.

2.1 Game Theory

The seminal works of von Neumann, Morgenstern and Nash [128, 174] have laid the foundations for a model of rational human interaction which we now call game theory. Over the years, the central notions of game theory have been scrutinized, argued over, and rewritten to better serve our increasingly complex understanding of its applications. Indeed, in this dissertation we focus on non-cooperative games, wherein players compete selfishly and without communicating with one another. To ensure ease of readibility, in this treatment of game theoretic preliminaries, we will leave out the substantial literature which pertains to cooperative games. This overview is primarily inspired by standard texts in game theory [174], augmented with more recent works for learning in games [35, 65]. Throughout this chapter, we will also provide concrete examples which will hopefully make these oftentimes abstract concepts more digestible.

2.1.1 Normal-Form Games

When one thinks of game theory, one is likely to imagine a game of Rock-Paper-Scissors between two rivals, or perhaps a Prisoner's Dilemma-type situation. These situations (and many more) can be modeled as *normal-form* games. Formally, a normal-form game $\Gamma \coloneqq (\mathcal{N}, \mathcal{S}, u)$ can be defined in terms of a set of players $\mathcal{N} = \{1, \ldots, N\}$, a set of strategies $\mathcal{S}$ and a payoff function u. In particular, each player i selects from a set of strategies $\mathcal{S}_i$, which can be either discrete or continuous. Finally, each player has a payoff function $u_i : \mathcal{S} \equiv \Pi_i \mathcal{S}_i \to \mathbb{R}$ which assigns some payoff $u_i(s)$ to player i. Let us consider each of these three components more closely via a simple example - Rock-Paper-Scissors (RPS).

A typical game of RPS consists of two players, the strategy set for both being $\{rock, paper, scissors\}$. Since Rock beats Scissors, Scissors beats Paper and Paper beats Rock, we can ascribe a payoff for each of the possible outcomes for playing these strategies. In particular, we can write the following payoff matrix for an RPS game:

TABLE 2.1: Rock-Paper-Scissors

	R	P	S
R	$0, 0$	$-1, 1$	$1, -1$
P	$1, -1$	$0, 0$	$-1, 1$
S	$-1, 1$	$1, -1$	$0, 0$

If the game has two players and the payoff function can be expressed as a matrix for each player, we can also alternatively call the normal-form game a 'bimatrix' game. Notice that the payoffs are determined based on the specific actions (either Rock or Paper or Scissors) chosen by each player. These are known as 'pure' strategies, but selecting a pure strategy might not always be the best choice. For example, no amount of bluffing, double-bluffing or other psychological tricks can help one consistently defeat an opponent who is randomly selecting between the three pure actions each time the game is played. This gives rise to the notion of a 'mixed' strategy. A mixed strategy is a probability distribution over pure actions, which we typically denote by $x_i = (x_{is_i})_{s_i \in \mathcal{S}} \in \Delta(\mathcal{S}_i)$. Players can then compute their payoff for playing a mixed strategy linearly by taking expectations. Formally,

$$u_i(x) = \sum_{s \in \mathcal{S}} u_i(s) \prod_{i \in \mathcal{N}} x_{is_i} \tag{2.1}$$

A natural question that arises from the formalism of games we study is thus: "What is the best strategy for players to employ in order to maximize their payoffs?". In order to quantify and analyze this question, Nash proposed the concept of a Nash equilibrium [128], which has led to an explosion of works in the past 70 odd years which study different notions of game-theoretic equilibria.

2.1.2 Equilibrium Concepts in Game Theory

There are many equilibrium notions that can be described in this section – every subclass of games invariably has some specific notion of equilibration that is reasonable for the specified setting. However, in the main text of this dissertation we will focus only on several broader key concepts which are most relevant to the dissertation topic.

Nash Equilibrium (NE). One of the most seminal ideas in classical game theory was Nash's introduction of the Nash equilibrium. The idea is simple but elegant - if players cannot increase their payoff by unilaterally deviating from the current strategy x_i, then they are playing a Nash equilibrium.

Definition 2.1.1 (Pure strategy Nash equilibrium)**.** *A strategy vector s^* is a pure strategy Nash equilibrium if*

$$u_i(s^*) \geq u_i(s_i, s^*_{-i}) \tag{PSNE}$$

for all $s_i \in \mathcal{S}_i$.

Definition 2.1.2 (Mixed strategy Nash equilibrium)**.** *A mixed strategy vector x^* is a mixed strategy Nash equilibrium if*

$$u_i(x^*) \geq u_i(x_i, x^*_{-i}) \tag{MSNE}$$

for all $x_i \in \mathcal{S}_i$.

For the RPS game described before, one can examine the payoff matrix and see that no pure action is a best response to another players' pure action. Hence we can conclude that there cannot be a Nash equilibrium in pure strategies for the RPS game. However, it can be shown that a mixed strategy Nash equilibrium of the game is for each player to choose rock, paper or scissors with equal probability.

Due to the conceptual simplicity of this equilibrium notion, it is often the case that the Nash equilibrium is seen as some sort of "holy grail" – finding the Nash of a given game in a given setting can often be the subject of much research. Indeed, Nash's original paper proves that a mixed-strategy Nash equilibrium always exists in normal-form games, which in principle somewhat legitimizes the search for Nash equilibria. However, there are several problems that arise from this paradigm. First, the original proof for the existence of Nash equilibrium uses Brouwer's Fixed Point theorem [25], and is thus not a constructive proof. This means that even though the Nash always exists in normal-form games, it is not clear exactly how one can derive it efficiently. To make matters worse, recent results in complexity theory have shown that computing NE in general games (even approximately) is formally PPAD-complete [37, 48]. This makes the problem intractable and thus there is little hope for finding a polynomial-time algorithm that computes NE. Moreover, it is not clear how players in a game can be incentivized to select a Nash equilibrium strategy, barring the restriction that all other players have to play their own Nash strategies. Indeed, in games with multiple Nash equilibria the problem of *equilibrium selection* comes to the fore, calling into further question the implementability and practicality of these equilibria. However, as we will see, this does not mean that all hope is lost. Several interesting classes of games do indeed skirt around the issues with Nash equilibria outlined above, and one thing is for certain: the idea of the NE as a "catch-all" equilibrium notion is flawed.

Correlated Equilibrium (CE). In order to overcome the limitations of the Nash equilibrium, several other equilibrium concepts have been proposed in recent years. A major example is the *correlated equilibrium*, introduced by Aumann [8]. In a correlated equilibrium, we first require a coordinator or randomizer of some sort shared between all players. Then, consider a probability distribution σ over all possible combinations of strategies for each player (i.e. the joint actions for the players). Each player observes their own action in the joint action space sampled from σ, and σ is called a correlated equilibrium if no player can improve their payoffs by deviating from the prescribed action.

Definition 2.1.3 (Correlated equilibrium)**.** *A probability distribution σ over all possible combinations of strategies for each player such that each player is recommended to play strategy x_i is a correlated equilibrium if for all players i and strategies $x_i, x_i' \in \mathcal{S}_i$,*

$$\mathbb{E}_{x\sim\sigma}(u_i(x)|x_i) \geq \mathbb{E}_{x\sim\sigma}(u_i(x_i', x_{-i})|x_i) \tag{CE}$$

We now present several indicative examples of correlated equilibria.

Example 2.1.1. *Let us look at the game of Chicken, where players in a competition have the option to either Dare or Chicken Out. The game is defined by the payoff matrix in Figure 2.2.*

TABLE 2.2: Game of Chicken

	C	D
C	$(6,6)$	$(2,7)$
D	$(7,2)$	$(0,0)$

The game has two pure Nash, namely (C,D) and (D,C), and a mixed Nash where both players Dare with probability $1/3$. The expected payoff for this mixed Nash is $14/3$.

Suppose now that a central coordinator selects uniformly at random from the set $\{CC, CD, DC\}$ (note that the players potentially know the elements of this set). After doing so, she tells each player their corresponding recommended strategy (but not the strategy of the opponent). Suppose that a player is assigned D. Then, they would not want to deviate if the opponent does not deviate from their recommended strategy. Likewise, if a player is assigned C, their opponent plays C and D with equal probability. Thus, the expected payoffs for playing C and D are 4 and 3.5 respectively. Hence, neither player has incentive to deviate. This is precisely a correlated equilibrium. Note also that this correlated equilibrium results in expected payoff of 5, which is an improvement over the mixed strategy Nash equilibrium.

Another common example utilizes a variant of the RPS game where players now receive a much lower payoff for playing the same strategy.

Example 2.1.2. *Like the original RPS game, mod-RPS has no pure strategy Nash equilibria and indeed the unique mixed strategy Nash equilibrium is again for each player to mix uniformly among the three pure strategies. However, in this case the expected total utility is $-20\cdot\frac{1}{3}+0\cdot\frac{2}{3} = -\frac{20}{3}$, instead of 0 like in the case of RPS. We can construct a CE as follows: each player samples uniformly over the off-diagonal pairs, giving expected utility of 0. When a player is given a signal by the coordinator, they will know that the other player was given a signal to select one of*

TABLE 2.3: Modified Rock-Paper-Scissors (mod-RPS)

	R	P	S
R	$-10, -10$	$-1, 1$	$1, -1$
P	$1, -1$	$-10, -10$	$-1, 1$
S	$-1, 1$	$1, -1$	$-10, -10$

the other two pure actions. By disregarding the recommendation, they run the risk of incurring a loss of -10*. If the other player follows their signal, then the total expected utility would be strictly worse than the* 0 *expected if the first player were to follow the signal. Hence, this is a CE with expected utility* 0*.*

By definition, every Nash equilibrium is also a correlated equilibrium, and indeed it is clear that all mixtures of Nash equilibria are also correlated equilibria.

Coarse Correlated Equilibrium (CCE). Finally, we also consider the more general concept of coarse correlated equilibria (CCE) [126]. If a distribution σ is no worse than always following some fixed strategy x_i' no matter what the coordinator recommends, then it is a CCE. Indeed, the set of CE is a subset of the set of CCE. Let us see an example of this:

Example 2.1.3. *Consider a 3 strategy game with the following payoff matrix:*

	A	B	C
A	$(1, 1)$	$(-1, -1)$	$(0, 0)$
B	$(-1, -1)$	$(1, 1)$	$(0, 0)$
C	$(0, 0)$	$(0, 0)$	$(-1.1, -1.1)$

Now, suppose a coordinator selects from $\{AA, BB, CC\}$ *with equal probability and recommends the action selected to each player. First, note that this is not a correlated equilibrium. To see why this is true, notice that if a player is recommended* C*, they always have the incentive to deviate to play any other strategy. However, the expected payoff for a player according to the distribution is* $1/3 + 1/3 - 1.1(1/3) = 0.3$*, but the payoff for a player playing fixed actions A or B is* 0 *while the payoff for playing C is strictly negative. The given distribution is thus a CCE, but not a CE.*

Definition 2.1.4 (Coarse correlated equilibrium)**.** *A probability distribution* σ *over all possible combinations of strategies for each player such that each player is recommended to play strategy* x_i *is a coarse correlated equilibrium if it satisfies for all players* i *and strategies* $x_i' \in \mathcal{S}_i$*,*

$$\mathbb{E}_{x\sim\sigma}(u_i(x)) \geq \mathbb{E}_{x\sim\sigma}(u_i(x_i', x_{-i})) \tag{CCE}$$

In other words, following a CCE is no worse than always following some fixed strategy x_i' no matter what the coordinator recommends. It is clear to see that every Nash equilibrium must also be a coarse correlated equilibrium.

The relationship between the three equilibrium notions presented above is illustrated in Figure 2.1. Comparing the definitions of CCE and CE above suggests a subtle but

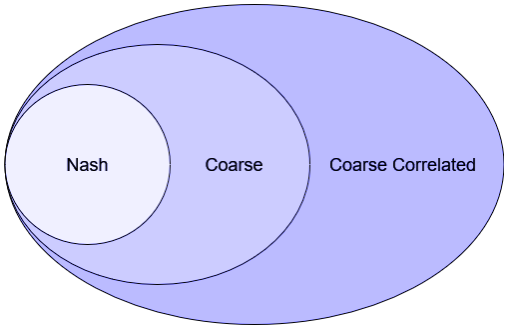


FIGURE 2.1: Hierarchy of Equilibrium Concepts

important difference between the two notions. In the case of CCE, the coordinator has the power to enforce the distribution. Specifically, the coordinator makes the distribution common knowledge and if players agree, the game is played according to this distribution, with players not being able to deviate. In other words, the coordinator has the power to take punitive action (in terms of payoffs) against any deviation. In the case of CE however, the coordinator again makes the distribution common knowledge to all players, but also broadcasts privately to each player their component in the outcome which was selected. Each player maintains the right to follow this recommendation or to deviate to some other strategy. Thus, in the case of CE, the only power required by the central authority (i.e. coordinator) is the ability to (privately) broadcast information to the players. The equilibrium distribution is thus self-enforcing.

Within this dissertation, we will focus primarily on Nash and coarse correlated equilibria. As we shall see, coarse correlated equilibria arise as an important solution concept because they are deeply connected to no-regret learning (see Chapter 2.2).

2.1.3 A Taxonomy of Non-Cooperative Games

In the wider taxonomy of games, some classic examples include *zero-sum* games [174], *coordination* games and *potential* games [124]. In the course of this dissertation, in additional to general normal-form games (sometimes called general-sum games), we will additionally focus on different formulations of zero-sum games. In totality, we call these games *non-cooperative*. While it would be reductive to state that these classes of games are representative of game theory research as a whole, they are simple and broad enough to allow theorists and computer scientists alike to find common ground and apply these models to interesting problems in machine learning and beyond.

Zero-Sum Games. A zero-sum game is simply a game where the net utility for all players is equal to zero. In the two-player setting, this means that the payoff for one player is exactly the negative of the payoff for the other given any outcome of the game. For instance, the RPS game (Table 2.1) is a zero-sum game. Many examples of competitive games in real life such as Poker, Go and Chess, are also zero-sum games. In addition to being a good model for competitive games, there are plenty of useful theoretical results surrounding two player zero-sum games. Some key classical results are summarized below:

- An equilibrium in mixed strategies always exists
- All equilibrium points give the same payoff to all players
- Computing these equilibria is relatively straightforward and can be performed in a tractable fashion

In other words, in two player zero-sum games we manage to circumvent many of the issues which exist for Nash equilibria in general games, since they can be computed tractably. The centerpoint result in this class of games is the well-known von Neumann Minimax Theorem [132], which is widely considered a fundamental result in game theory.

Theorem 2.1.1 (Minimax Theorem)**.** *Consider any two player zero-sum game with player set* $\mathcal{N} = \{1, 2\}$*, strategy sets* $\mathcal{S}_1$ *and* $\mathcal{S}_2$*, and payoff matrix* A *where players use mixed strategies* $x \in \mathcal{S}_1$ *and* $y \in \mathcal{S}_2$*. Then,*

$$\max_x \min_y x^\top Ay = \min_y \max_x x^\top Ay$$

This means that any two player zero-sum game can be seen as a minmax optimization problem, which thus allows for efficient computation of Nash equilibria via linear optimization techniques [45, 94]. This draws a strong connection between game theory and optimization, which can be exploited to derive online optimization-inspired algorithms to compute solutions in the game-theoretic setting.

On the flip side, negative results abound for games that have greater than two players or are not zero-sum. [48] and [37] proved that even two player non-zero-sum games are PPAD-complete, meaning that computing them is an intractable problem. Most settings beyond two player games are predictably intractable as well - three player zero-sum games are also PPAD-complete. This is not to say that Nash are not computable at all in general (multiplayer) bimatrix games - the Lemke-Howson algorithm [105] solves for NE of any bimatrix game, but has a worst-case running time which is exponential in the number of pure strategies of the game. Recently, it was even shown that it is PSPACE-complete to find any solution which can be obtained with the Lemke-Howson algorithm [70]. These intractability results have led to the research direction of finding classes of multiplayer games that do admit tractable equilibria, as well as finding polynomial time approximation schemes (PTAS) or even decentralized dynamics that converge to said equilibria.

Polymatrix Games. Perhaps surprisingly, a certain class of multiplayer games with a useful structural property does indeed possess tractable Nash equilibria – zero-sum network/polymatrix games. This type of game is defined on a graph where the nodes represent players and the edges between nodes represent two player bimatrix games between the players on the endpoints. Each node (player) has a set of strategies, and selects a mixed strategy to be used for all the edge games they are involved in. Finally, a player's payoff is computed by taking the sum over all bimatrix games the player participates in. We describe this class of games formally in Definition 2.1.5.

Definition 2.1.5 (Polymatrix game). *A graphical polymatrix game is a game defined on an undirected graph $\mathcal{G} = (V, E)$ such that the following holds:*

- *The vertices (or nodes) $V = \{1, \dots, N\}$ represent players, and edges E represent two-player games between a pair of players (i, j), where $i \neq j$.*
- *For each player $i \in V$, we associate a set of pure actions $\mathcal{S}_i = \{1, \dots, n_i\}$, from which the player can select at random via mixed strategy x_i. The set of mixed strategies for player $i \in V$ is the standard simplex in $\mathbb{R}^{n_i}$, denoted by $\mathcal{X}_i = \Delta^{n_i - 1} = \{x_i \in \mathbb{R}^{n_i}_{\geq 0} : \sum_{s \in \mathcal{S}_i} x_{is} = 1\}$, where x_{is} denotes the probability mass on action $s \in \mathcal{S}_i$.*
- *We call the set of all possible strategy profiles the* strategy space, *denoted by $\mathcal{X} = \Pi_{i \in V} \mathcal{X}_i$.*
- *For each edge $(i, j) \in E$, we associate a two-player bimatrix game A^{ij}, A^{ji} where $A^{ij} \in \mathbb{R}^{n_i \times n_j}$ and $A^{ji} \in \mathbb{R}^{n_j \times n_i}$. The strategy sets for the players are $\mathcal{S}_i \in \mathbb{R}^{n_i}$ and $\mathcal{S}_j \in \mathbb{R}^{n_j}$ respectively.*
- *An entry A^{ij}_{uv} for $(u, v) \in \mathcal{S}_i \times \mathcal{S}_j$ represents the reward player i obtains for selecting action u given that player j selects v. We note that the graph G might also contain self-loops, meaning that the player $i \in V$ plays a game defined by A^{ii} against themselves.*
- *The payoff of player $i \in V$ under strategy profile $x \in \mathcal{X}$ is denoted by $u_i(x)$ and is the sum of payoffs from the bimatrix games the player participates in. It can also be expressed as $u_i(x_i, x_{-i})$ when distinguishing between the strategy of player i and all other players $-i$. Formally,*

$$u_i(x) = \sum_{j:(i,j) \in E} x_i^\top A^{ij} x_j.$$

  We further denote by $u_{is}(x) = \sum_{j:(i,j) \in E (A^{ij} x_j)_s}$ the utility of player i under strategy profile $x = (s, x_{-i}) \in \mathcal{X}$ for $s \in \mathcal{S}_i$.
- *Finally, the game $\mathcal{G}$ is called (globally) zero-sum if for all strategy profiles $x \in \mathcal{X}$, we have that $\sum_{i \in V} u_i(x) = 0$. Moreover, if there are positive coefficients $\{\lambda_i\}_{i \in V}$ such that $\sum_{i \in V} \lambda_i u_i(x) = 0$ for all $x \in \mathcal{X}$ and the self loops are antisymmetric (meaning that $A^{ii} = -(A^{ii})^\top$), the game is called rescaled zero-sum.*

Initially, [24] studied this class of games and provided a linear programming formulation that could find equilibria. However, their formulation is exponential in constraints and variables. Subsequently, [51] circumvented these issues by showing that the class of (pairwise) zero-sum polymatrix games can be reduced to a two-player zero-sum bimatrix game. Pairwise zero-sum polymatrix games are a special case of the globally zero-sum polymatrix games we focus on, where each of the edge games is zero-sum.

Theorem 2.1.2 ([51]). *There is a polynomial-time reduction from any pairwise zero-sum polymatrix game $\mathcal{G}$ to a symmetric zero-sum bimatrix game Γ, such that from any Nash equilibrium of Γ one can recover a Nash equilibrium of $\mathcal{G}$ in polynomial time.*

The immediate corollary of this result is that the Nash equilibria of these games can be computed in polynomial time using linear programming! Moreover, the work also establishes that the set of Nash equilibria is convex and that no-regret algorithms converge to a Nash equilibrium in pairwise zero-sum polymatrix games. This indicates

that these games share many positive and negative properties with two-player zero-sum games, including PPAD-completeness.

We turn our attention now to known results for globally zero-sum polymatrix games. [32] showed a transformation between pairwise and globally polymatrix games:

Theorem 2.1.3 ([32])**.** *There is a polynomial-time computable payoff preserving transformation from every separable zero-sum multiplayer game to a pairwise constant-sum polymatrix game.*

A constant-sum game is a polymatrix game where every edge can be constant-sum for any arbitrary constants. Next, they also show a direct reduction of global zero-sum polymatrix games to linear programming by establishing a restricted zero-sum property satisfied by such games.

Finally, [31] showed that several results that hold for pairwise zero-sum polymatrix games also hold for (globally) zero-sum polymatrix games. Indeed the following results hold for zero-sum polymatrix games:

(1) A Nash equilibrium can be computed in polynomial- time using a simple linear program.

(2) Coarse correlated equilibria of these games collapse to Nash equilibria, implying that no-regret learning converges to Nash equilibria. No-regret learning and its relation to convergence to coarse correlated equilibria will be explored in more detail in Section 2.2..

Extensive-Form Games. Beyond normal-form games, *extensive-form games* (EFGs) are an important class of games which have been studied for more than 50 years [98]. EFGs capture various settings where selfish players sequentially perform actions which change the *state of nature*, with the action-sequence finally leading to a *terminal state*, at which each player receives a payoff. The most ubiquitous examples of EFGs are real-life games such as Chess, Poker, Go etc. Moreover, this type of game can model environments where the sequential timing of strategic decisions is important. As an example, a design firm has to make choices about what type of product to create depending on the available materials and their forecasts of the market/user preferences.

EFGs are defined on a finite tree, where each node represents a game state or decision point assigned to a player. Much like in normal-form games, at each decision point the players have a finite strategy set that determines their pure actions. Depending on the acting player's action, the game then proceeds down the tree. Finally, the players are assigned a payoff as a function of the history of choices made. This definition will be formalized in Chapter 6, where we study EFGs in greater detail.

An important aspect of EFGs compared to normal-form games is that they can capture *imperfect information* of the players – in many real-world settings, players may not have full knowledge of the game, their opponents' strategy set and so on. Think of a competitive Poker game, where each player has a deck of hidden cards that is privy to them and only them. In the formalism of EFGs we will study, *information sets* capture imperfect information. An information set for a player intuitively denotes the set of nodes which are indistinguishable to that player given what they know.

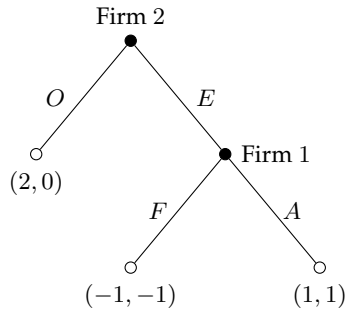


FIGURE 2.2: Entry Game with perfect information.

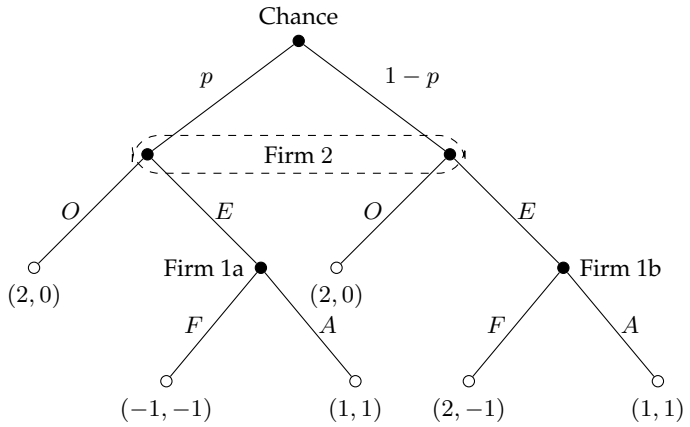


FIGURE 2.3: Entry Game with imperfect information. Information set for Firm 2 is denoted by dotted box.

We provide an indicative example of an extensive-form game called the Model of Entry (or the Entry Game).

Example 2.1.4. *Consider a game with two firms, where Firm 1 (player 1) is an incumbent monopolist who has footing in an industry. Firm 2 (player 2) is a potential entrant to the industry, who can decide to either enter (E) or stay out (O). Firm 1, upon observing the choice of Firm 2, can then choose to either fight (F) or accommodate (A). If Firm 2 decides to stay out, Firm 1 continues their incumbency, earning some profit. If Firm 2 decides to enter and Firm 1 decides to fight, both firms lose profit. If Firm 2 decides to enter and Firm 1 decides to accommodate them, both firms instead share the profits that are gained. The game tree can be seen in Figure 2.2.*

Example 2.1.5. *Consider a modification to the entry game where there are two possible types of incumbent firm. Firm 1a is rational and has the payoffs of the entry game, but Firm 1b is deranged and considers fighting to be honorable, thus deriving a higher payoff from fighting. Moreover, we let chance decide if the Firm 1 is of a rational or deranged type with probabilities p and $1 - p$ respectively. In this modified game, Firm 1 always knows its preferences, but Firm 2 is unsure of Firm 1's type. Thus, the game can be written in extensive form as shown in Figure 2.3.*

Much like in normal-form games, the Nash equilibrium is a key equilibrium concept in EFGs. Indeed, in small EFG instances one can write the EFG in strategic (i.e. normal-) form and solve for the Nash equilibrium in that form. However, due to the sequential

nature of EFGs, there are some Nash equilibria which are unreasonable, in the sense that they will never be reached in a rational sequence of play. In this case, there are several other notions of equilibria which are commonly studied in EFGs. One such example is the subgame perfect equilibrium [97]. In particular, if each player has perfect recall, in the sense that they never forget any of their previous moves and available strategies at each of the previous moves, then a subgame perfect equilibrium. Backward induction is a commonly utilized method to find 'reasonable' subgame perfect equilibria, which has found applications beyond game theory [150, 152]. In the context of this dissertation, however, we will investigate how simple learning dynamics converge to the set of Nash equilibria of EFGs that exhibit perfect recall, and the rationality of these equilibria is left to future work.

2.2 Online Learning and Dynamical Systems in Games

The framework of game theory we have presented above is primarily focused on equilibrium analysis – game theorists ask questions about the existence and computation of equilibria for a given game. However, when applying these ideas to real-world systems, one naturally has to ask the further question of: 'Does the empirical behaviour of learning agents actually approach the Nash equilibrium?'. One school of thought postulates that the Nash equilibrium exists as a seminal solution concept *because* a set of fully rational agents would naturally gravitate towards it as they learn and update their strategies. However, this theory is unfounded when considering more complicated scenarios, such as when the game does not have a unique equilibrium. Indeed as we have seen in the previous sections, the Nash equilibrium simply isn't enough as a solution concept for games in general.

To be more constructive, one could ask a slightly different question: 'Can we design natural learning algorithms/models that lead to equilibrium play? What type of equilibrium is reached?'. Many models for learning in games have been proposed along the lines of these questions, and one such model which we will borrow heavily from is *online learning*. Under this framework, the agents in a system update their strategies in a sequential manner, obtaining some form of reward at each timestep from the environment (i.e., the game). Their objective is then to maximize their cumulative reward, or equivalently, to minimize their *regret*. As we shall soon see, there is a strong connection between regret-minimization and convergence to game theoretic equilibria, which forms the basis of our research.

Online learning in games stems from the area of online optimization [30, 80]. For the purposes of this dissertation, we will focus on the model which is amenable to game-theoretic interpretation. First, it is instructive for us to focus on general-sum, normal-form games. In this setting, let us introduce additional notation to express player rewards in a more intuitive manner. Let $v_{is_i}(x) = u_i(s_i; x_{-i})$ denote the reward i receives if i opts to play pure strategy s_i when everyone else commits to their strategies described by x. Specifically, the notation $(s_i; x_{-i})$ denotes the strategy x after replacing x_i with s_i. This results in $u_i(x) = \langle v_i(x_{-i}), x_i \rangle$. With that in mind, the multiplayer online learning in games setting has the following model:

(1) At every timestep, each player i independently selects a mixed strategy x_i.

(2) Each player then observes their utility $u_i(x)$.

(3) The player updates their mixed strategy in the next timestep according to an update rule.

A remarkable guarantee that has been studied extensively in the literature is that if players update their strategies using a type of algorithm called *no-regret* algorithms [35, 76, 110, 145], then their actual payoff over time is close to the highest-rewarding action of the game in hindsight. This property is formally captured via the notion of *regret*.

Definition 2.2.1 (Regret). *Given a discrete-time sequence of mixed strategies $x^1, \ldots, x^T$ the regret of player i up until time T, Reg_i^T, is defined as*

$$Reg_i^T := \max_{x_i \in \mathcal{X}_i} \sum_{t=1}^{T} \langle v_i(x_{-i}), x_i^t \rangle - \sum_{t=1}^{T} \langle v_i(x_{-i}^t), x_i^t \rangle$$

Intuitively, the first term indicates the cumulative reward of the best fixed action in hindsight, while the second indicates the cumulative reward of the algorithm used. Any online learning algorithm which is able to guarantee $Reg_i^T = o(T)$ (i.e. that the time-average regret vanishes as $T \to \infty$) is called *no-regret*.

Crucially, there exists a connection between any no-regret online learning algorithm and coarse correlated equilibria, which we describe in the following theorem:

Theorem 2.2.1 (Folkore). *Given a sequence of mixed strategies $(x^1, \ldots, x^T)$, then the probability distribution $\hat{\mu} := \sum_{t=1}^{T} \mu_{x^t}/T$ is a (Reg^T/T)-approximate Coarse Correlated Equilibrium where $Reg^T := \max_{i \in \mathcal{N}} Reg_i^T$.*

Theorem 2.2.1 implies that if all players update their strategies with a no-regret online learning algorithm, then the time-average strategy vector converges to the set of CCE with rate $o(T)/T$. As a result, the time-average strategy vector converges to CCE as $T \to \infty$. This elucidates a fundamental connection between online learning dynamics and a static equilibrium concept, and has had a major impact on the literature of learning in games.

A closely related question is that of how quickly the regret vanishes for a given no-regret algorithm (equivalently, how fast the algorithm converges to CCE). It turns out that the rate at which the expected regret vanishes for any no-regret algorithm is lower-bounded by $\mathcal{O}(1/\sqrt{T})$. Indeed, consider the following simple example:

Example 2.2.1 ([145, 154]). *Consider an adversarial setting where at each timestep T, an adversary selects uniformly at random between payoff vectors $A = [1, 0]$ and $B = [0, 1]$. No matter how good an online learning algorithm is, it will always have expected payoff of exactly $T/2$. Then, using Hoeffding's inequality to bound the probability of obtaining extreme values, it can be shown that with constant probability, one of the two fixed actions in hindsight (A or B) has cumulative payoff of $T/2 - \Theta(\sqrt{T})$. It follows that best possible long run average regret is $\Theta(1/\sqrt{T})$. A similar argument shows that with m actions, the time-average regret is lower-bounded by $\Theta(\sqrt{\log(m)/T})$.*

Applications of No-Regret Learning in Games. Crucially, no-regret algorithms do indeed exist, and they are often applied to game-theoretic contexts. In the remainder of this chapter, we will describe several online learning frameworks in two primary regimes – continuous-time and discrete-time. While many machine learning applications rely on discrete-time algorithms, continuous-time dynamics are still important to study, as they represent a connection to the field of evolutionary game theory [84, 148]. This allows us to draw similarities between learning (in the context of computer science) and evolution (in the context of biology). Moreover, continuous-time dynamics allow for various discretizations, and thus their learning behavior can provide a higher-level, unified picture of their discrete-time counterparts [123].

2.2.1 Continuous-Time Dynamics

The family of continuous-time dynamics we focus on are the Follow-The-Regularized-Leader (FTRL) dynamics. These have been studied extensively in the literature [80, 154], and crucially can be specialized into the ubiquitous replicator dynamics and gradient descent dynamics. Within this dissertation, in Chapters 3 and 5 we focus entirely on replicator dynamics, while in Chapter 4 we study both continuous-time gradient descent and also general FTRL dynamics. Note that in an online optimization context, the dynamics we study are typically written using loss functions. However, since we deal with game-theoretic settings where players obtain rewards, we will write the update rules in terms of payoffs instead.

In continuous-time, we have an analogous definition of regret for any sequence of mixed strategies $x(t)$. Note here that we use the lowercase t to denote the continuous time-sequence, whereas in discrete-time it is customary to determine a fixed time horizon (which we denote T) for which to run the algorithm.

Definition 2.2.2 (Continuous-Time Regret)**.** *Given a continuous-time sequence of mixed strategies $x(t)$, the regret of player i is*

$$Reg_i(t) := \max_{x_i \in \mathcal{X}_i} \int_0^t [u_i(x_i; x_{-i}(\tau)) - u_i(x(\tau))] d\tau$$

Follow-The-Regularized-Leader. FTRL is a versatile class of dynamics that has been well-studied due to its no-regret properties. To better understand the reasoning for the construction of this class of dynamics, we must first describe the *Follow-The-Leader* (FTL) algorithm. in the FTL framework, each player simply chooses the strategy whose cumulative reward thus far is maximal. However, this naive algorithm has potential to perform disastrously – it can be shown that the regret of FTL grows linearly with time in the worst case [35].

FTRL arises as a modification to FTL where players are provided an initial payoff vector $y_i(0)$, and a *regularizer* is introduced. In essence, FTL adapts too aggressively to the payoffs obtained previously, and so the addition of a regularizer allows FTRL to avoid overfitting, improving upon the regret bound. The (con-FTRL) dynamic in

continuous-time is as follows:

$$\begin{aligned} y_i(t) &= y_i(0) + \int_0^t v_i(x(\tau))d\tau \\ x_i(t) &= \text{argmax}_{x_i \in \mathcal{X}_i}\{\langle y_i(t), x_i\rangle - \tfrac{h_i(x_i)}{\eta_i}\} \end{aligned} \tag{con-FTRL}$$

Here, $h_i(x_i)$ denotes a strongly convex and continuously differentiable regularizer and $y_i(t)$ denotes the *cumulative* payoffs of player i accrued until time t. η_is are parameters which tune the weight of the regularizer. With this setup, FTRL enjoys the following regret bound:

Theorem 2.2.2 ([117]). *A player following* (con-FTRL) *enjoys an $\mathcal{O}(1/t)$ regret bound, no matter what other players do. Specifically, if player $i \in \mathcal{N}$ follows* (con-FTRL) *then for every continuous trajectory of play of the opponents,*

$$Reg_i(t) \leq \frac{\max h_i - \min h_i}{t}$$

It is also worth noting that FTL and FTRL are respectively related to fictitious play [26] and smooth fictitious play [64], which are well-studied algorithms for learning in games. Players using fictitious play select the best response (respectively, regularized best response for smooth fictitious play) to the history of the opponent's strategies. Indeed, this frames FTRL as an important 'meta'-algorithm within the literature of learning in games. Moreover, by careful selection of regularizer, FTRL can be specialized into several important learning dynamics with rich history, further adding to its significance.

Projected Gradient Descent. Firstly, by selecting an L2-regularizer $h_i(x_i) = \frac{1}{2}\|x_i\|_2^2$, FTRL specializes to online (projected) gradient descent.

$$\begin{aligned} y_i(t) &= y_i(0) + \int_0^t v_i(x(\tau))d\tau \\ x_i(t) &= \text{argmax}_{x_i \in \mathcal{X}_i} \|y_i(t) - x_i\|_2^2 \end{aligned} \tag{con-GDA}$$

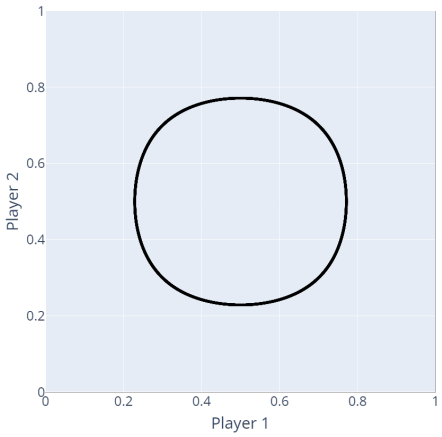
Without going into extraneous details, gradient descent is one of the most classical and important methods in computer science and online optimization [187].

Replicator Dynamics. By selecting entropy regularizer $h_i(x_i) = \sum_{s_i \in \mathcal{S}_i} x_{is_i} \log x_{is_i}$, FTRL specializes to replicator dynamics. In its classical form, it is typically written as follows:

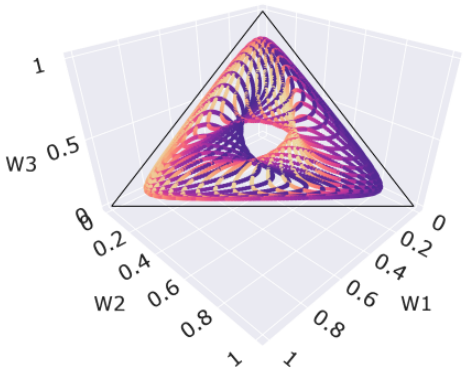
$$\dot{x}_{is_i} = x_{is_i}(v_{is_i}(x) - u_i(x)), \quad \forall s_i \in \mathcal{S}_i. \tag{RD}$$

The replicator dynamics are one of the most widely studied learning dynamics in evolutionary game theory [84] and population biology [151, 166], and is often used as a model for natural selection and evolution. In the language of evolutionary game theory, the $v_{is_i}(x)$ term denotes the fitness of a population of type i, while $u_i(x)$ denotes the average population fitness. The bulk of Part I of this dissertation, which focuses on continuous-time dynamics, will utilize continuous-time FTRL dynamics and specializations thereof.

Recurrence Results in Continuous-Time Dynamics. One of the most important recent results in learning in games has been the observation of cycling or recurrent behavior of many no-regret dynamics in zero-sum games. Initially, [138] showed that the trajectories of replicator dynamics in zero-sum games exhibits a property known as Poincaré Recurrence. Moreover, [20] fully characterized this property in zero-sum games with interior Nash equilibria (i.e. if the equilibrium is mixed and has full support). Moreover, if the games have two or fewer degrees of freedom (for instance, RPS or MP), then the dynamics will be formally periodic. [117] then showed that this recurrent behavior occurs in the wider class of FTRL dynamics as well. The study of Poincaré recurrence in games has led to a stronger understanding of the behavior of learning dynamics in non-cooperative settings, as well as the stark contrast between the time-average behavior and the day-to-day behavior of learning dynamics. In Figure 2.4, we show some examples of recurrent behavior in zero-sum games.



(A) RD in Matching Pennies



(B) RD in endogenously evolving zero-sum game (see Chapter 3)

FIGURE 2.4: RD exhibiting recurrent behavior in various types of zero-sum games.

In many real-world settings, this divide between time-average convergence to equilibrium and day-to-day cycling/recurrent behavior can cause major problems. For instance, in GANs, time-average convergence does not suffice in the training process, and instead the generator and discriminator need to converge pointwise to a suitable equilibrium [49, 173]. Hence, formally recurrent behavior can be viewed as a negative result which also serves as an intuitive explanation for mode collapse in GAN training [147]. In our study of various formulations of zero-sum games within Chapters 3, 4 and 5, we focus primarily on establishing recurrence results as a method of better understanding the day-to-day behavior of well-studied learning dynamics.

2.2.2 Discrete-Time Dynamics

Follow-The-Regularized-Leader. The FTRL dynamics can be written in discrete-time as follows:

$$x_i^{t+1} = \underset{x_i \in \mathcal{X}_i}{\operatorname{argmax}} \left\{ \left\langle \sum_{\tau=1}^{t} v_i(x^\tau), x_i \right\rangle - \frac{h_i(x_i)}{\eta_i} \right\} \tag{FTRL}$$

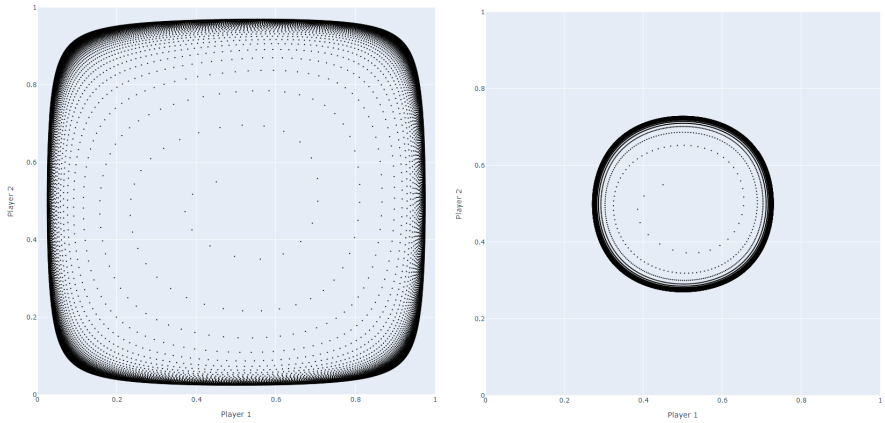
Similarly to the continuous-time case, the discrete-time FTRL can also be specialized by appropriately selecting continuous and strictly convex regularizer h_i. For instance, selecting Euclidean regularizer we obtain the (discrete-time) projected gradient dynamics and selecting entropic regularizer we obtain the discrete-time variant of replicator dynamics, the Multiplicative Weights Update (MWU).

Multiplicative Weights Update. In Chapter 7 we focus on MWU and variants thereof. In the notation of normal-form games, the update rule for MWU can be written with respect to each strategy as:

$$x_{is_i}^{t+1} = \frac{x_{is_i}^t \exp\{(\eta_i \cdot v_{is_i}(x^t)\})}{\sum_{\bar{s}_i \in \mathcal{S}_i} x_{i\bar{s}_i}^t \exp\{(\eta_i \cdot v_{i\bar{s}_i}(x^t))\}} \tag{MWU}$$

Here, η_i denotes the step-size/learning rate for player i. When players adopt MWU, they are ensured to experience sub-linear regret, $Reg_i(T) \leq \mathcal{O}\left(\sqrt{T}\right)$ no matter how the other players update their strategies [35]. This implies that the time-averaged difference between the *payoff of the best fixed strategy in hindsight* and *the actual produced reward* goes to zero with rate $\mathcal{O}(1/\sqrt{T})$.

While MWU admits a strong regret guarantee, it has been well-documented to diverge from the interior Nash equilibrium even in two-player zero-sum games such as Matching Pennies [13, 40] (Figure 2.5). Notice that as the step-size increases, the trajectories begin to resemble a periodic orbit, which is precisely the behavior of replicator dynamics in a Matching Pennies game.



(A) $\eta_i = t^{-1/2}$ (B) $\eta_i = t^{-2/3}$

FIGURE 2.5: MWU diverging to boundary in a Matching Pennies (two-player zero-sum) game with different decreasing step-sizes.

Several variants of MWU (as well as other online learning dynamics) have been proposed that can converge in day-to-day behavior to the Nash equilibrium, while also improving on the time-average regret guarantees. One such variation is called 'optimism' [49, 142, 164]. This modification makes the optimistic assumption that the state of the system in the next time-step will be identical to the current time-step, resulting in a slight recency bias in the learning behavior. One example is Optimistic Multiplicative Weights Update (OMWU) (also referred to as Optimistic Hedge), which is a variant of MWU where the payoff contributions of the last time-step are taken into account twice. The update rule for OMWU can be written as:

$$x_{is_i}^{t+1} = \frac{x_{is_i}^{t} \exp\{(\eta_i \cdot (2v_{is_i}(x^t) - v_{is_i}(x^{t-1})\})}{\sum_{\bar{s}_i \in \mathcal{S}_i} x_{i\bar{s}_i}^{t} \exp\{(\eta_i \cdot (2v_{i\bar{s}_i}(x^t) - v_{i\bar{s}_i}(x^{t-1}))\}} \tag{OMWU}$$

Such algorithms and variants thereof have enjoyed wide adoption in recent years, showing strong performance gains over their non-optimistic counterparts in various settings (e.g. [2, 9, 47, 49, 58, 59, 68, 72, 86, 119, 123, 177, 178]).

Online Mirror Descent. We shift our focus to a different class of algorithms which are studied in more detail in Chapter 6. (Online) Mirror Descent (MD) [130] arises as a type of no-regret algorithm with a similar regret bound as FTRL but with a simpler update rule. To define this class of dynamics, we first need to define the notion of *Bregman divergence*. For a convex function $h : \mathbb{R}^d \mapsto \mathbb{R}$, the corresponding Bregman divergence is defined as

$$D_h(x, y) := h(x) - h(y) - \langle \nabla h(y), x - y \rangle ,$$

where ∇ denotes the gradient operator. If h is γ-strongly convex, then $D_h(x, y) \geq \frac{\gamma}{2}\|x - y\|$, where $\|\cdot\|$ is shorthand for the L_2-norm. Then, the online mirror descent update rule for normal-form games is given by:

$$x_i^{t+1} = \text{argmax}_{x_i \in \mathcal{X}_i} \left\{ \langle \nabla v_i(x^t), x_i \rangle + \frac{1}{\eta_i} D_{h_i}\left(x_i, x_i^t\right) \right\} \tag{MD}$$

Note that when the payoff functions are linear, the FTRL and MD algorithms generate the exact same dynamics [80]. Indeed, one can also specialize mirror descent to projected gradient descent and multiplicative weights update by selecting suitable regularizer. Similar to the case of MWU and other FTRL variants, optimistic mirror descent can be defined, and has proven to be a particularly effective tool for learning in games [49, 59, 119, 141]. To avoid confusion in our notation, we denote the online mirror descent update by MD, while its optimistic variant is denoted by OMD.

Big O Notation. In order to analyze the asymptotic convergence properties of the dynamics described above, we will make use of the Bachmann-Landau notation [10, 103], also commonly known as big O notation. In particular, consider functions $f, g : \mathbb{R} \to \mathbb{R}$. Small O is denoted by $f(t) = o(g(t))$, which means that f is asymptotically dominated by g. Formally, $\limsup_{t\to\infty} f(t)/g(t) = 0$. Big O is denoted by $f(t) = \mathcal{O}(g(t))$, which indicates that f is asymptotically bounded above by g (up to a constant factor). Formally, there exists some positive constant $c > 0$ such that $|f(t)| \leq cg(t)$ for large t. Conversely,

big Omega is denoted by $f(t) = \Omega(g(t))$, which indicates that f is asymptotically bounded below by g, i.e. $g(t) = \mathcal{O}(f(t))$. Finally, big Theta is denoted by $f(t) = \Theta(g(t))$, which indicates that both $f(t) = \mathcal{O}(g(t))$ and $f(t) = \Omega(g(t))$ (i.e. f is asymptotically bounded above and below by g).

2.3 Dynamical Systems

Our study of continuous-time systems in games will typically revolve around the mathematics of dynamical systems. While discrete-time dynamical systems are important as well, our applications of discrete-time dynamics have a more machine learning flavor, so in that regime we instead focus on proving asymptotic convergence to equilibria and providing regret guarantees of the dynamics.

Whenever we move into the realm of discrete-time dynamical systems, we will introduce the necessary ideas within the corresponding chapter. The areas of learning in games and dynamical systems are intrinsically connected. As mentioned earlier, [138] initially showed that the trajectories of replicator dynamics can exhibit a property known as Poincaré Recurrence in zero-sum games. Since then, many works have proven similar results in various game settings [20, 117, 127, 135, 173].

To formulate our toolbox of dynamical systems theory, we first consider a continuous-time ODE on a topological space $\mathcal{X}$:

$$\dot{x} = f(x)$$

Flows: The existence and uniqueness theorem for ordinary differential equations [79] guarantees that there exists a unique continuous function $\phi : \mathbb{R} \times \mathcal{X} \to \mathcal{X}$, which is termed the *flow*, that satisfies (i) $\phi(t, \cdot) : \mathcal{X} \to \mathcal{X}$ (often denoted $\phi^t : \mathcal{X} \to \mathcal{X}$) is a homeomorphism for each $t \in \mathbb{R}$, (ii) $\phi(t + s, x) = \phi(t, \phi(s, x))$ for all $t, s \in \mathbb{R}$ and all $x \in \mathcal{X}$, and (iii) for each $x \in \mathcal{X}$, $\frac{d}{dt}|_{t=0}\phi(t, x) = f(x)$.

Liouville's Theorem. Liouville's theorem can be applied to any system of ordinary differential equations with a continuously differentiable vector field ξ on an open domain $\mathcal{Y} \in \mathbb{R}^d$. Let us draw a connection between the divergence of ξ and a notion of volume. The divergence of ξ at $y \in \mathcal{Y}$ is the trace of the Jacobian at y. Formally, we can write $\text{div}\ \xi(y) = \sum_{i=1}^{d} \frac{\partial \xi_i}{\partial y_i}(y)$. Because the divergence is continuous, it is integrable on Lebesgue measurable subsets of $\mathcal{Y}$. Given any such subset A, define the image of A under flow ϕ at time t as $A(t) = \{\phi(a, t) : a \in A\}$. By construction, $A(t)$ is measurable and it has volume $\text{vol}[A(t)] = \int_{A(t)} d\mu$.

Liouville's theorem states that the time derivative of the volume $\text{vol}[A(t)]$ exists and links it to the divergence of ξ in the following manner:

$$\frac{d}{dt}\left[\text{vol}\ A(t)\right] = \int_{A(t)} \text{div}(\xi) d\mu \tag{2.2}$$

If $\text{div}\ \xi(y)$ is null at any $y \in \mathcal{Y}$, then the volume is conserved. Since $\text{div}\ \xi$ is continuous, the converse statement is also true – if the volume is conserved on any open set, $\text{div}\ \xi(y)$

has to be null at any point $y \in \mathcal{Y}$. Intuitively, a flow ϕ^t is volume preserving if and only if the divergence of f at any point $x \in \mathbb{R}^d$ equals zero.

Preservation of Volume: The flow ϕ of a system of ODEs is called *volume preserving* if the volume of the image of any set $U \subseteq \mathbb{R}^d$ under ϕ^t is preserved. More precisely, for any set $U \subseteq \mathbb{R}^d$, $\text{vol}(\phi^t(U)) = \text{vol}(U)$. Whether or not a flow preserves volume can be determined by applying Liouville's theorem, which says the flow is volume preserving if and only if the divergence of f at any point $x \in \mathbb{R}^d$ equals zero—that is, $\text{div} f(x) = \text{Tr}(Df(x)) = \sum_{i=1}^{d} \frac{df(x)}{dx_i} = 0$.

Poincaré Recurrence: If a dynamical system preserves volume and every orbit remains bounded, almost all trajectories return arbitrarily close to their initial position, and do so infinitely often [140]. Given a flow ϕ^t on a topological space $\mathcal{X}$, a point $x \in \mathcal{X}$ is *nonwandering* for ϕ^t if for each open neighborhood U containing x, there exists $T > 1$ such that $U \cap \phi^T(U) \neq \emptyset$. The set of all nonwandering points for ϕ^t, called the *nonwandering set*, is denoted $\Omega(\phi^t)$.

Theorem 2.3.1 (Poincaré Recurrence [140])**.** *If a flow preserves volume and has only bounded orbits, then for each open set almost all orbits intersecting the set intersect it infinitely often: if ϕ^t is a volume preserving flow on a bounded set $Z \subset \mathbb{R}^d$, then $\Omega(\phi^t) = Z$.*

Homeomorphisms and Diffeomorphisms: A function f between two topological spaces is called a homeomorphism if it is a bijection, is continuous and admits a continuous inverse. On the other hand, a function f between two topological spaces is called a diffeomorphism if it is a bijection, is continuously differentiable and admits a continuously differentiable inverse.

We remark that the concepts utilized in this dissertation do not fully encompass the usage of dynamical systems for learning in games. For instance, stability results have been established for various notions of equilibria, and Lyapunov theory provides a theoretical grounding for the study of congestion games and beyond. Indeed, there is no doubt that the interplay between dynamical systems and learning in games is a rich area for continued research, with many more discoveries and connections to come.

Part I

Continuous-Time Dynamics: Conservation Laws and Recurrence

And though we pass them by today,
tomorrow we may come this way.

J.R.R. Tolkien, *The Fellowship of the Ring*

In the taxonomy of non-cooperative games described in Chapter 2, we explored a collection of game types which arise from *normal-form* game theory. We studied online learning algorithms that allow for utility-maximizing agents to arrive at various notions of equilbria within this paradigm. In the first part of this dissertation, we study continuous-time dynamics in games beyond the standard normal-form setting.

First, in Chapter 3 we study the class of games that are "doubly" or "endogenously"-evolving. This means that as the players of the game select their strategies, the payoffs of the game adjust in relation to their strategies. This models the situation where exploiting a particular strategy results in its relative advantages being reduced over time. Secondly, in Chapter 4 we shift our focus to periodic games, where the game payoffs evolve over time in a periodic fashion without taking into account the players' actions. Finally, in Chapter 5 we study a generalization of normal-form games called quantum zero-sum games, where competing players select density matrices as mixed strategies instead of probability vectors. In all of these game types, we study continuous-time online learning dynamics, exploring fundamental volume preservation laws and establishing recurrent behavior.

Chapter 3

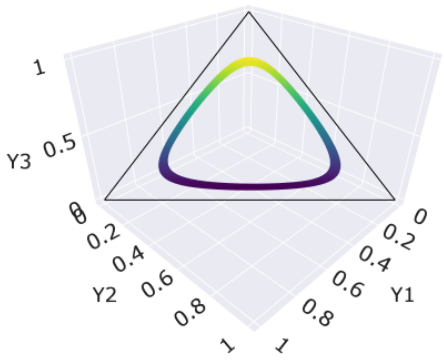
Replicator Dynamics in Endogenously Evolving Zero-Sum Games

This chapter is taken from (with minor modifications) our paper *'Evolutionary Game Theory Squared: Evolving Agents in Endogenously Evolving Zero-Sum Games'* [160].

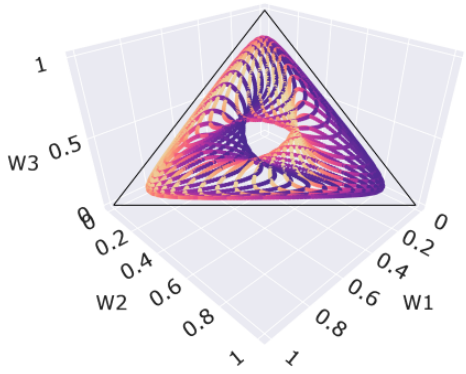
3.1 Introduction

Game theory has seen many applications ever since the initial burst of discovery and innovation in the early 20th century. One of the most important applications of game theory has to do with biology, via the paradigm of *evolutionary game theory* (EGT). This field arose as a strategic framework through which evolution in biological systems could be studied. Many learning dynamics in this setting (the most famous of which being replicator dynamics, which we have seen in Chapter 2.2) have found uses in related fields, such as population games [84, 148], online learning in games [35, 133], and multi-agent systems [156]. The dominant paradigm in these areas of research is that of evolutionary players adapting to each others behavior. In other words, the dynamism of the environment of each player is driven by the other players, whereas the rules of interaction between the players, i.e., the game which is being played, is static.

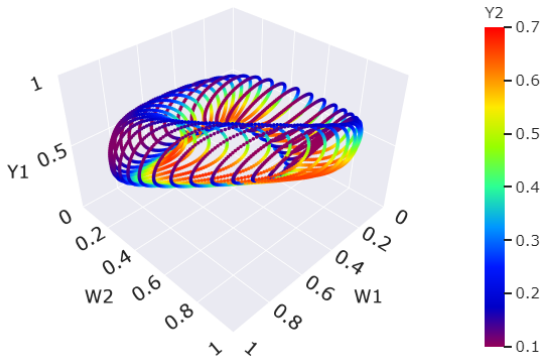
This separation between *evolving players* and a *static game* is so standard that it typically goes unnoticed. However, this fundamental restriction actually limits us from capturing many real-world applications of interest. In artificial intelligence [42, 66, 120, 162, 176, 182], biology and sociology [22, 163, 168, 169, 179] as well as economics [63, 107], the rules of interaction can themselves adapt to the collective history of the players' behavior. For example, in adversarial learning and curriculum learning [18, 87], the difficulty of the game increases over time due to the focus on finding improvement in settings where the agent has performed worst in history. Similarly, in biology or economics, if a particular advantageous strategy is used exhaustively by agents, then its relative advantages typically dissipate over time (sometimes known as diminishing returns), which once again drives the need for innovation and exploration. We call all these games



(A) Population evolution (y)



(B) Environment evolution (w)



(C) Coevolution of y and w

FIGURE 3.1: Poincaré recurrence in a time-evolving generalized Rock-Paper-Scissors game [114].

endogenously-evolving games, since their evolution over time depends on the actions and behavior of the game's very own players.

An endogenously-evolving game can be thought of almost like an algorithm – much like online learning algorithms in the literature, the game has a state which encodes the history of play and produces an output whenever players select new strategies. However, unlike online learning algorithms that receive a history of payoff vectors and output the players' next strategy (i.e., a probability distribution over pure actions), an endogenously-evolving game receives as input a history of players' behavior and instead outputs a new payoff matrix for the next epoch. Hence, the players of the game and the game itself can be thought of as "dual" algorithmic objects which are coupled in their evolution over time (Figure 3.1).

The obvious challenge that arises in studying this class of games is how to mathematically express the endogenous evolution - on one hand, we want to study a mechanism that is reasonable to find in real-world systems but on the other, we also require said mechanism to still be amenable to mathematical analysis. Moreover, once we move away from the safe haven of static games, we lose the prized standard methodology for

analysis that typically consists of two steps: i) compute/understand equilibria of the given game (e.g., Nash equilibria, coarse correlated equilibria, etc.) and their properties; ii) correlate behavior of learning dynamics to a target class of equilibria (e.g., convergence or recurrent behaviour). Indeed, the only prior work to ours that considers games larger than 2×2 [114], focused on a specific payoff matrix structure based on a Rock-Paper-Scissors game. They then prove that this system exhibits recurrent behavior via a tailored argument that was explicitly designed for the dynamical system in question, with no clear connections to game theory. In this chapter, we will revisit this problem and their formulation, and find a new systematic analysis based on game theory that generalizes to arbitrary *network zero-sum* games.

3.1.1 Our Contributions

We provide a general framework for analyzing learning in time-evolving zero-sum games, as well as rescaled network generalizations thereof. To begin, we develop a novel *reduction* that reduces time-evolving games to a (static) game-theoretic graph that generalizes both network/polymatrix zero-sum games and evolutionary zero-sum games. In this generalized but static game, evolving players and evolving games represent different types of nodes (nodes with and without self-loops) in a graph connected by edges which represent two-player games. The bridge we form between time-evolving games and static network games makes the latter far more interesting than previously thought: our reduction proves they are sufficiently expressive to capture not only multiple pairwise interactions, but time-varying environments as well. Moreover, by providing a path back to the familiar territory of evolving players interacting in a static game, the mathematical tools of game theory and dynamical systems theory become available. This allows us to perform a broad algorithmic analysis of commonly studied systems from machine learning and biology that previously required specialized treatment.

From an algorithmic learning perspective, we focus on the most studied evolutionary learning dynamic: the replicator dynamics. Remarkably, despite the chaotic co-evolution of players and games that forces players to continually innovate, the system can be shown to exhibit a number of regularities. We prove the system is *Poincaré recurrent*, with effectively all initializations of players and games lying on recurrent orbits that come arbitrarily close to their initial conditions infinitely often (Figures 3.1). As a crucial component of this result, we demonstrate that the dynamics obey information-theoretic *conservation laws* that couple the behavior of all players and games (Figure 3.14). Moreover, while the system never equilibrates, the conservation laws allow us to prove the time-average behavior and utility of the players converge to the time-average Nash equilibrium of their evolving games with bounded regret. Finally, we provide a *polynomial time algorithm* that can efficiently predict these time-average quantities. Moreover, the network structure of the games we study are interesting from a simulation perspective – many difficulties arise in interpreting recurrence in games, and we showcase comprehensive simulations which are able to intuitively corroborate the theoretical results.

3.2 Preliminaries and Definitions

In Definition 2.1.5 of Chapter 2, we have defined network/graphical polymatrix games, which is the key object we study in this chapter. We repeat the definition briefly in this section for convience. Importantly, we also need to extend the classical replicator dynamics (RD) to the polymatrix setting.

Polymatrix Games. An N-player *polymatrix game* is defined using an undirected graph $G = (V, E)$ where V corresponds to the set of players and E corresponds to the set of edges between players in which a *bimatrix game* is played between the endpoints [32]. Each player $i \in V$ has a set of actions/pure strategies $\mathcal{S}_i = \{1, \dots, n_i\}$ that can be selected at random from a distribution x_i called a *mixed strategy*. The set of mixed strategies of player $i \in V$ is the standard simplex in $\mathbb{R}^{n_i}$ and is denoted $\mathcal{X}_i = \Delta^{n_i - 1} = \{x_i \in \mathbb{R}^{n_i}_{\geq 0} : \sum_{\alpha \in \mathcal{S}_i} x_{i\alpha} = 1\}$ where $x_{i\alpha}$ denotes the probability mass on action $\alpha \in \mathcal{S}_i$. Note here that we deviate slightly from the notation established in Chapter 2, where actions were denoted s. This is because in this and the next chapter, our analysis requires granularity in comparing the actions of each player playing the bimatrix games on edges. Thus instead of using s, we will denote the players' actions as α and β for player i and player j respectively for clarity. The state of the game is then defined by the concatenation of the strategies of all players. We call the set of all possible strategies profiles the *strategy space*, and denote it by $\mathcal{X} = \prod_{i \in V} \mathcal{X}_i$.

The bimatrix game on edge (i, j) is described using a pair of matrices $A^{ij} \in \mathbb{R}^{n_i \times n_j}$ and $A^{ji} \in \mathbb{R}^{n_j \times n_i}$. An entry $A^{ij}_{\alpha\beta}$ for $(\alpha, \beta) \in \mathcal{S}_i \times \mathcal{S}_j$ represents the reward player i obtains for selecting action α given that player j chooses action β. We note that the graph G may also contain *self-loops*, meaning that a player $i \in V$ plays a game defined by A^{ii} against themselves. The *utility* or *payoff* of player $i \in V$ under the strategy profile $x \in \mathcal{X}$ is denoted by $u_i(x)$ and corresponds to the sum of payoffs from the bimatrix games the player participates in. The payoff is equivalently expressed as $u_i(x_i, x_{-i})$ when distinguishing between the strategy of player i and all other players $-i$. More precisely,

$$u_i(x) = \sum_{j:(i,j) \in E} x_i^\top A^{ij} x_j. \tag{3.1}$$

We also need a way of writing the utility of player $i \in V$ under a specific strategy profile $x = (\alpha, x_{-i}) \in \mathcal{X}$ for $\alpha \in \mathcal{S}_i$ (i.e. the player's utility for selecting action α when the other players play using strategy profile x). We denote this by $u_{i\alpha}(x) = \sum_{j:(i,j) \in E} (A^{ij} x_j)_\alpha$. The game is called *zero-sum* if $\sum_{i \in V} u_i(x) = 0$ for all $x \in \mathcal{X}$. Moreover, if there are positive coefficients $\{\eta_i\}_{i \in V}$ such that $\sum_{i \in V} \eta_i u_i(x) = 0$ for all $x \in \mathcal{X}$ and the self-loops are antisymmetric (meaning $A^{ii} = -(A^{ii})^\top$), the game is called *rescaled zero-sum*.

As described before, the Nash equilibrium of this class of games is a mixed strategy profile $x^* \in \mathcal{X}$ such that for each player $i \in V$,

$$u_i(x_i^*, x_{-i}^*) \geq u_i(x_i, x_{-i}^*), \ \forall x_i \in \mathcal{X}_i. \tag{3.2}$$

We denote the support of $x_i^* \in \mathcal{X}_i$ by $\mathrm{supp}(x_i^*) = \{\alpha \in \mathcal{S}_i : \ x_{i\alpha} > 0\}$. A Nash equilibrium is said to be an *interior* or *fully mixed* Nash equilibrium if $\mathrm{supp}(x_i^*) = \mathcal{S}_i$ for each $i \in V$.

Replicator Dynamics. In polymatrix games, *replicator dynamics* [148] for each $i \in V$ are given by

$$\dot{x}_{i\alpha} = x_{i\alpha}(u_{i\alpha}(x) - u_i(x)), \quad \forall \alpha \in \mathcal{S}_i. \tag{3.3}$$

We suppress the explicit dependence on time t in the system and do so throughout this chapter where clear from context to simplify notation. Moreover, we only consider initial conditions on the interior of the simplex, i.e. that the players' initial strategies have full support over all pure actions. The replicator dynamics are equivalently given in vector form for each $i \in V$ by the system

$$\dot{x}_i = x_i \cdot \left(\sum_{j:(i,j)\in E} A^{ij} x_j - \left(\sum_{j:(i,j)\in E} x_i^\top A^{ij} x_j \right) \cdot \mathbf{1} \right), \tag{3.4}$$

where $\mathbf{1}$ is an n_i–dimensional vector of ones and the operator $(\cdot)$ denotes elementwise multiplication.

For the purpose of analysis, the replicator dynamics in (3.3) are often translated by a diffeomorphism from the interior of $\mathcal{X}$ to the cumulative payoff space $\mathcal{C} = \prod_{i\in V} \mathbb{R}^{n_i - 1}$, which is defined by a mapping such that $x_i = (x_{i1}, \ldots, x_{in_i}) \mapsto (\ln \frac{x_{i2}}{x_{i1}}, \ldots, \ln \frac{x_{in_i}}{x_{i1}})$ for each player $i \in V$. The reasoning behind this transformation will be explained in more detail in later sections of this chapter. All other necessary preliminaries for dynamical systems utilized in this chapter are presented in Chapter 2.3. Note that since the replicator dynamics we study are Lipschitz continuous, a unique flow ϕ of the replicator dynamics exists.

3.3 Studying Endogenously-Evolving Processes via Polymatrix Games

Numerous applications from artificial intelligence (AI) and machine learning (ML) to biology cast competition between populations (e.g., neural networks/algorithms or species/agents) and the environment (e.g., hyperparameters/network configurations or resources) as a time-evolving dynamical system. The basic abstraction of such an endogenously-evolving system takes the form of a population y of *species* which evolve dynamically in time as a function of itself and some *environmental* parameters w whose evolution, in turn, depends on y. In this section, we review models from these applications and connect a broad class of time-evolving dynamical systems to static polymatrix games. This reduction provides a path toward analyzing complex non-stationary dynamics using tools developed for the typical static game formulation.

Endogenously-Evolving Behavior in AI and ML. Evolutionary game theoretic methods for training generative adversarial networks commonly exhibit time-evolving dynamic behavior. In this setting, there are two predominant endogenous evolutionary models [42, 43, 66, 120, 176]. In the first formulation, [176] describe training the generator network, with parameters y, via a gradient-based algorithm composed of *variation*, *evaluation*, and *selection*. The discriminator network, with parameters w updated via gradient-based learning, is modeled as the environment operating in a feedback loop with y. The second model is such that the generator and discriminator are different species (or *modules*) in the population y which follows evolutionary dynamics, and

network hyperparameters (or *chromosomes*) w evolve in time as a function of y [42, 66, 120].

Endogenously-Evolving Behavior in Biology. There are also two common endogenous evolutionary formulations emerging in biology. In the first, the focus is on the level of coordination in a population as a function of evolving environmental variables. The prevailing model is comprised of replicator dynamics $\dot{y} = y(1-y)((A(w)y)_1 - (A(w)y)_2)$ in which a population of two species y plays a prisoner's dilemma (PD) game against themselves. In this setting, the payoff matrix $A(w)$ depends on an environmental variable w which in turn depends on the population via $\dot{w} = w(1-w)G(y)$. Here, $G(y)$ is a feedback mechanism which intuitively models environmental degradation or enhancement, which occurs as a function of y [100, 168, 169, 179]. For example, in [179], $G(y)$ takes the form $\theta y - (1-y)$ for some $\theta > 0$, which represents the ratio of the enhancement rate to the degradation rate of 'cooperators' and 'defectors' in the time-evolving PD game.

In the second formulation, the focus is on studying how competition among species is modulated by resource availability. Indeed, from a biological perspective, [114] argue that the environmental parameters w on which a population y of N antagonistic species depend are not constant, but rather evolve over time. Since the species' fitness depends on the environment, the game among the species is also time-varying, and based on the Rock-Paper-Scissors game. They give a model of the dynamic behavior in this setting for each species $i \in \{1, \dots, N\}$, with initial conditions on the interior of the simplex for both w and y:

$$\begin{aligned} \dot{w}_i &= w_i \sum_{j=1}^{N} w_j (y_j - y_i) \\ \dot{y}_i &= y_i \big((P(w)y)_i - y^\top P(w) y \big) \end{aligned} \tag{3.5}$$

where $P(w) = P + \mu W$ for $\mu > 0$. In particular, P is defined as the generalized Rock-Paper-Scissors (RPS) payoff matrix:

$$P = \begin{pmatrix} 0 & -1 & 0 & \cdots & 0 & 0 & 1 \\ 1 & 0 & -1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 & -1 \\ -1 & 0 & 0 & \cdots & 0 & 1 & 0 \end{pmatrix},$$

and W is the environmental variation matrix:

$$W = \begin{pmatrix} 0 & w_1 - w_2 & \cdots & w_1 - w_n \\ w_2 - w_1 & 0 & \cdots & w_2 - w_n \\ \vdots & \vdots & \vdots & \vdots \\ w_n - w_1 & w_n - w_2 & \cdots & 0 \end{pmatrix}.$$

[114] studied this dynamical system in (3.5) and showed it exhibits a special type of cyclic behavior: *Poincaré recurrence*. As mentioned before, their analysis is tailored to this particular system, which has particular biological significance. Our goal is to

abstract and generalize this analysis, which would allow for a broader understanding of endogenously-evolving behavior in biology, ML and beyond.

3.3.1 Reducing Time-Evolving RPS to a Polymatrix Game

Let us first focus on the specific model of [114], and show how this can be written as a rescaled zero-sum polymatrix game (Proposition 3.3.1). By capturing the evolution of the environment (dynamics of the payoff matrix) as additional players that dynamically change their strategies, we reduce the co-evolution of w and y to a *static polymatrix game* of greater dimensionality (greater number of players). Given this reduction, Theorem 3.4.1, which establishes the Poincaré recurrence of replicator dynamics in rescaled zero-sum polymatrix games, immediately captures the results of [114] (see Corollary 3.4.1).

Proposition 3.3.1. *The time-evolving generalized rock-paper-scissors game from* (3.5) *is equivalent to replicator dynamics in a two-player rescaled zero-sum polymatrix game.*

We dedicate the remainder of this subsection to showing why Proposition 3.3.1 is true. Let y and w denote the mixed strategies of player 1 (population) and player 2 (environment) respectively. In what follows, we show that both the population and environment dynamics can be simplified in such a way that each player follows replicator dynamics in a static rescaled zero-sum polymatrix game.

Environmental Dynamics. We begin by considering the dynamics of the environment. As written in [114], the dynamics of player 2 (w-player) for each action $i \in \{1, \dots, n\}$ with initial condition on the interior of the simplex are given by:

$$\dot{w}_i = w_i \sum_{j=1}^{n} w_j \left(y_j - y_i\right). \tag{3.6}$$

Now observe that:

$$\begin{aligned}
\sum_{i=1}^{n} \dot{w}_i &= \sum_{i=1}^{n} w_i \sum_{j=1}^{n} w_j \left(y_j - y_i\right) \\
&= \sum_{i=1}^{n} w_i \sum_{j=1}^{n} w_j y_j - \sum_{i=1}^{n} w_i \sum_{j=1}^{n} w_j y_i \\
&= \sum_{i=1}^{n} w_i \sum_{j=1}^{n} w_j y_j - \sum_{j=1}^{n} w_j \sum_{i=1}^{n} w_i y_i \\
&= 0.
\end{aligned}$$

Since $\sum_{i=1}^{n} \dot{w}_i = 0$, and the given initial condition is such that $w(0) \in \Delta^{n-1}$, we conclude that $w(t) \in \Delta^{n-1}$ and $\sum_{j=1}^{n} w_j(t) = 1$ for any $t \geq 0$. Next, we obtain an equivalent form

of the dynamics given in (3.6) for each action $i \in \{1, \dots, n\}$ as follows:

$$\begin{aligned}
\dot{w}_i &= w_i \sum_{j=1}^{n} w_j \left(y_j - y_i\right) \\
&= w_i \sum_{j=1}^{n} w_j y_j - w_i \sum_{j=1}^{n} w_j y_i \\
&= w_i \sum_{j=1}^{n} w_j y_j - w_i y_i \\
&= w_i \big(- y_i + \sum_{j=1}^{n} w_j y_j \big).
\end{aligned} \tag{3.7}$$

The dynamics of the w-player from (3.7) in vector form are then given by:

$$\dot{w} = w \cdot (-Iy + w^\top I y). \tag{3.8}$$

These dynamics are precisely the replicator dynamics in which player 2 (environment) plays against player 1 (population) with the payoff matrix $A^{w,y} = -I$, where the superscript (w, y) indexes the respective players.

Population Dynamics. We now perform a similar analysis for the population dynamics. The dynamics for player 1 (y-player) for each action $i \in \{1, \dots, n\}$ with interior initial condition are given by:

$$\dot{y}_i = y_i \left((P(w)y)_i - y^\top P(w) y \right). \tag{3.9}$$

Note that these are already replicator dynamics, but written with respect to payoff matrix $P(y)$ – we want to rewrite this in terms of static matrix P. From an expansion of the payoff matrix $P(w)$ in (3.9), the dynamics of player 1 (y-player) for each action $\{1, \dots, n\}$ are equivalently:

$$\dot{y}_i = y_i \left((Py)_i - y^\top P y \right) + y_i \left(\mu \sum_{j=1}^{n} (w_i - w_j) y_j - \mu \sum_{\ell=1}^{n} y_\ell \sum_{j=1}^{n} (w_\ell - w_j) y_j \right). \tag{3.10}$$

Observe that:

$$\begin{aligned}
\sum_{\ell=1}^{n} y_\ell \sum_{j=1}^{n} (w_\ell - w_j) y_j &= \sum_{\ell=1}^{n} y_\ell \sum_{j=1}^{n} w_\ell y_j - \sum_{\ell=1}^{n} y_\ell \sum_{j=1}^{n} w_j y_j \\
&= \sum_{j=1}^{n} y_j \sum_{\ell=1}^{n} w_\ell y_\ell - \sum_{\ell=1}^{n} y_\ell \sum_{j=1}^{n} w_j y_j \\
&= 0.
\end{aligned}$$

Consequently, for each action $i \in \{1, \dots, n\}$, the dynamics in (3.10) simplify to the form:

$$\dot{y}_i = y_i \Big((Py)_i - y^\top Py \Big) + y_i \left(\mu \sum_{j=1}^{n} (w_i - w_j) y_j \right). \tag{3.11}$$

Finally, since $y(0) \in \Delta^{n-1}$, we have that $\sum_{j=1}^{n} y_j(t) = 1$ for any $t \geq 0$. Accordingly, for each action $i \in \{1, \dots, n\}$, we simplify the dynamics in (3.11) as follows:

$$\begin{aligned}
\dot{y}_i &= y_i \Big((Py)_i - y^\top Py \Big) + y_i \big(\mu \sum_{j} (w_i - w_j) y_j \big) \\
&= y_i \Big((Py)_i - y^\top Py \Big) + y_i \big(\mu w_i \sum_{j=1}^{n} y_j - \sum_{j=1}^{n} w_j y_j \big) \\
&= y_i \Big((Py)_i - y^\top Py \Big) + y_i \big(\mu w_i - \sum_{j=1}^{n} \mu w_j y_j \big).
\end{aligned} \tag{3.12}$$

The dynamics for player 1 (y-player) from (3.12) in vector form are then given by

$$\dot{y} = y \cdot (Py + y^\top Py \cdot \mathbf{1}) + y \cdot (\mu I w + \mu y^\top I w \cdot \mathbf{1}). \tag{3.13}$$

These dynamics are replicator dynamics in which player 1 (y-player) plays against itself with the payoff matrix $A^{y,y} = P$ and against player 2 (w-player) with the payoff matrix $A^{y,w} = \mu I$, where once again the superscripts in the payoff matrices index the respective players.

Putting it all together, we have shown that the dynamics of (3.6) and (3.9) correspond to replicator dynamics for a two-player polymatrix game in which player 1 (y-player) has utility $u_y(y, w) = y^\top Py + \mu y^\top I w$ and player 2 (w-player) has utility $u_w(y, w) = -w^\top I y$ for any strategy profile (y, w). The self-loop of player 1 (y-player) is antisymmetric and for $\eta_y = 1$ and $\eta_w = \mu$, the rescaled sum of utilities $\eta_y u_y(y, w) + \eta_w u_w(y, w) = 0$ for every strategy (y, w). This allows us to conclude that the time-evolving generalized rock-paper-scissors game studied in [114] is equivalent to replicator dynamics in a two-player rescaled zero-sum polymatrix game, with $\eta_y = 1$ and $\eta_w = \mu$.

3.3.2 A Generalized N-Player Reduction

A very natural follow-up question to the above observation is the following: *Are there more games for which dynamic co-evolution can be reduced to a static polymatrix game?* It turns out that the class of games which we coin 'endogenously-evolving' does admit such a reduction in general. These games are defined by a set of populations $y = (y_1, \dots, y_{n_y})$ and environments $w = (w_1, \dots, w_{n_w})$, where $y_\ell \in \Delta^{n-1}$ for each $\ell \in \{1, \dots, n_y\}$ and $w_k \in \Delta^{n-1}$ for each $k \in \{1, \dots, n_w\}$. It is a requirement that environments co-evolve only with populations and not with other environments. Intuitively, this means that the environment only changes based on what the species/population does. Moreover, any population can only co-evolve with environments and itself. Note that the self loops have to be antisymmetric (for example, an RPS game).

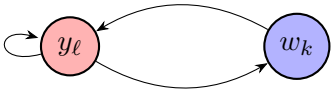


FIGURE 3.2: Basic interaction structure in the time-evolving systems that reduce to rescaled zero-sum polymatrix games. The *red* node denotes a population of species, while the *blue* node is an environment.

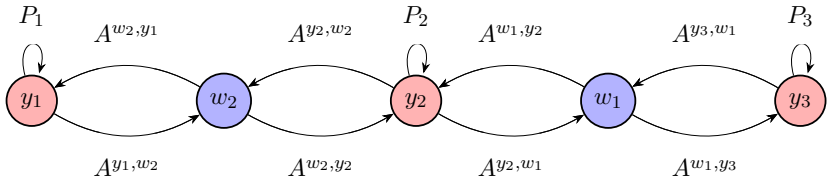


FIGURE 3.3: Example polymatrix game that can be formed from a reduction of a time-evolving dynamical system. The *red* nodes denote a population of species, while the *blue* nodes denote an environment.

Let $\mathcal{N}_k^w$ be the set of populations which co-evolve with w_k and let $\mathcal{N}_\ell^y$ be the set of environments which co-evolve with y_ℓ. Then, the time-evolving dynamics for each environment k and population ℓ are given componentwise by:

$$\dot{w}_{k,i} = w_{k,i} \sum_{\ell \in \mathcal{N}_k^w} \sum_j w_{k,j} \left((A^{k,\ell} y_\ell)_i - (A^{k,\ell} y_\ell)_j \right), \tag{3.14}$$

$$\dot{y}_{\ell,i} = y_{\ell,i} \left((P_\ell(w) y_\ell)_i - y_\ell^\top P_\ell(w) y_\ell \right), \tag{3.15}$$

where $P_\ell(w) = P_\ell + \sum_{k \in \mathcal{N}_\ell^y} W^{\ell,k}$ with $P_\ell \in \mathbb{R}^{n_\ell \times n_\ell}$ and $W^{\ell,k} \in \mathbb{R}^{n_\ell \times n_k}$ is defined such that the (i,j)–th entry is $(A^{\ell,k} w_k)_i - (A^{\ell,k} w_k)_j$.

Despite the complex nature of this dynamical system, we can show that it is equivalent to replicator dynamics in a polymatrix game. The complete proof of this result can be is deferred to Appendix A.1.

Theorem 3.3.1. *Any time-evolving system defined by the dynamics in Equations 3.14 & 3.15 is equivalent to replicator dynamics (Equation 3.3) in a rescaled zero-sum polymatrix game.*

The class of time-evolving systems that admit such a reduction possess a basic 'building block' (Figure 3.2). Here, the arrows between nodes are assigned a payoff matrix. The key component of any general structure formed using these building blocks is that every environment w_k is only connected to populations, and every population y_ℓ is only connected to environments, or to themselves via a self-loop. As an example of the type of generalized system that is possible using the reduction, consider the polymatrix game in Figure 3.3. Naturally, the graph which encodes the game need not be a line, but population nodes have to be separated by environment nodes.

The expressive power we gain from this reduction allows us to efficiently describe and characterize co-evolutionary processes of higher complexity than before, since we can now return to the familiar territory of analyzing dynamic agents in static games. In the

rest of this chapter, we focus on providing theoretical results for the subclass of time-evolving systems which reduce to rescaled zero-sum polymatrix games. However, this reduction is of independent interest since it can prove useful for future work analyzing the class of general-sum games via studying the behavior of network zero-sum games and rescaled generalizations thereof.

3.4 Poincaré Recurrence

In this section, we show that replicator dynamics are Poincaré recurrent in N-player rescaled zero-sum polymatrix games with interior Nash equilibria. In particular, for almost all initial conditions $x(0) \in \mathcal{X}$, the replicator dynamics will return arbitrarily close to $x(0)$ an infinite number of times.

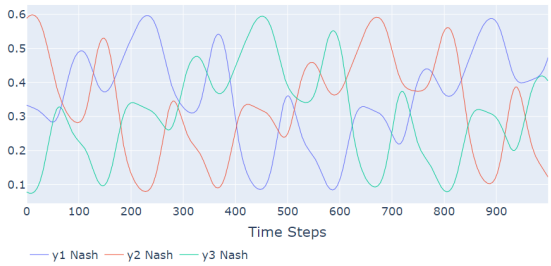
Theorem 3.4.1. *The replicator dynamics given in* (3.3) *are Poincaré recurrent in any N-player rescaled zero-sum polymatrix game that has an interior Nash equilibrium.*

There is a breadth of work studying the emergence of recurrent behavior of replicator dynamics in network zero-sum games [20, 117, 127, 135, 136, 138]. The standard method to prove such a result is showing that the Kullback-Leibler (KL) divergence between the replicator dynamics trajectory and the Nash equilibrium remains constant. However, it is not clear how to apply this proof method when the game is no longer static. Indeed, the Nash equilibrium of the games we study is not even fixed! This is a key technical challenge which we illustrate by considering the time-evolving generalized rock-paper-scissors game proposed by Mai et al. [114]. We show the evolution of the Nash equilibrium over time in Figure 3.4a, the evolution of the population strategy vector in Figure 3.4b, and the KL divergence between the evolving equilibrium and the replicator trajectory in Figure 3.4c. Clearly, the Nash equilibrium is no longer static and furthermore the KL divergence is not a constant of motion. This rules out the opportunity to follow standard proof techniques for showing replicator dynamics are recurrent in time-evolving games.

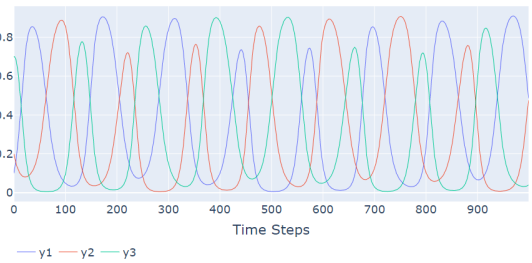
[20] is the closest known result to ours. They prove that replicator dynamics are Poincaré recurrent in N-player *pairwise zero-sum* polymatrix games with an interior Nash equilibria, which requires that $A^{ij} = -(A^{ji})^\top$ for every $(i, j) \in E$ (i.e. every game played between players is zero-sum). Our extension to N-player *rescaled* zero-sum polymatrix games is a far more general characterization of recurrence in replicator dynamics, since there are no explicit restrictions on the games played on each edge. Furthermore, the polymatrix game as a whole need not even be strictly zero-sum! The significance of this result is further enhanced by the connection developed in the previous section between a class of time-evolving games and N-player rescaled zero-sum polymatrix games. As a concrete example of the power of this connection, our recurrence result (Theorem 3.4.1) immediately recovers the main result of Mai et al. [114] as a corollary.

Corollary 3.4.1. *The time-evolving generalized rock-paper-scissors game in* (3.5) *is Poincaré recurrent.*

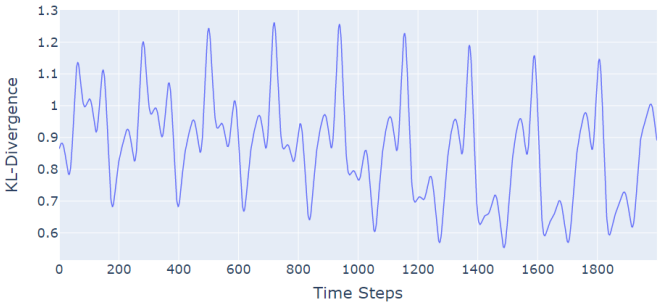
In the rest of this section, we present a sketch of the proof of recurrence in this class of games, and accompany them with simulations of this behavior in various games. It is worth noting that the technical results we prove in order to show the system is Poincaré



(A) Evolution of Nash



(B) Evolution of y-player strategies



(C) KL-divergence

FIGURE 3.4: Nash equilibrium of the time-evolving game, evolving strategies and the KL divergence between the Nash and strategies, from left to right, under replicator dynamics for the time-evolving generalized RPS model [114].

recurrent, namely volume preservation and the bounded orbits property, are themselves independently important as they provide conservation laws that couple the behavior of the players. In fact, they are fundamental to showing that while the system never equilibrates in the last-iterate sense, the time-average dynamics and utility converge to the Nash equilibrium and its utility (more on this in Section 3.5).

3.4.1 Overview of Proof Methods

To prove Poincaré recurrence, we need to show the flow corresponding to the system of ordinary differential equations in (3.3) is *volume preserving* and has *bounded orbits* (cf. Theorem 2.3.1). Notice that the flow of (3.3) always has bounded orbits since $x_{i\alpha} \geq 0$ and $\sum_{\alpha \in \mathcal{S}_i} x_{i\alpha}(t) = 1 \; \forall \; i \in V$. However, proving the volume preserving property is not as straightforward. To show volume preservation, we need to transform the dynamics via a *canonical transformation*. The idea is to choose a transformation which is a diffeomorphism to the original dynamics, while also allowing for a simpler volume preservation analysis in the transformed space. In the remainder of the dissertation, this will become a pattern – often, the challenge arises in finding such a transformation in order to facilitate our analysis of volume preservation. Due to this, we prove Poincaré recurrence of the flow of a system of ordinary differential equations which is diffeomorphic to the flow of the replicator equation in Equation 3.3. Then, by Theorem 3.3.1, we immediately obtain Poincaré recurrence of Equations 3.14 & 3.15.

The canonical transformation we utilize is as follows. Given $x \in \mathcal{X}$, consider the transformed variable $z \in \mathbb{R}^{n_1+\cdots+n_N-N}$ defined by

$$z_i = \left(\ln \frac{x_{i2}}{x_{i1}}, \ldots, \ln \frac{x_{in_i}}{x_{i1}}\right), \ \forall i \in V. \tag{3.16}$$

Given the vector z_i, the components of x_i are thus given by $x_{i\alpha} = e^{z_{i\alpha}}/(\sum_{\ell=1}^{n_i} e^{z_{i\ell}})$. Under this transformation, $\dot{z} = F(z)$ is given componentwise for each $\alpha \in \mathcal{S}_i$ and all $i \in V$ by:

$$\dot{z}_{i\alpha} = F_{i\alpha}(z) = \frac{\dot{x}_{i\alpha}}{x_{i\alpha}} - \frac{\dot{x}_{i1}}{x_{i1}} = \sum_{j\in V}\sum_{\beta\in\mathcal{S}_j}(A^{ij}_{\alpha\beta} - A^{ij}_{1\beta})e^{z_{j\beta}}/\sum_{\ell=1}^{n_j} e^{z_{j\ell}}. \tag{3.17}$$

Observe that $F_{i1} = 0$, which means that $z_{i1} = 0$ for all time. To show Poincaré recurrence of (3.3), we prove two key properties: *(i)* the flow of $\dot{z}$ is volume preserving, meaning the trace of the Jacobian of the vector field $\dot{z} = F(z)$ is zero, and *(ii)* $\dot{z}$ has bounded orbits from any interior initial condition. Then, the Poincaré recurrence of $\dot{z}$, and consequently $\dot{x}$, follows directly from Theorem 2.3.1.

3.4.2 Volume Preservation

To begin, we show that the trace of the vector field $F(z)$ is zero, which then from Liouville's theorem guarantees $\dot{z}$ (as defined in (3.17)) is volume preserving.

Lemma 3.4.1. *For any N-player rescaled zero-sum polymatrix game,*

$$\operatorname{tr}(DF(z)) = \sum_{i\in V}\sum_{\alpha\in\mathcal{S}_i}\frac{d}{dz_{i\alpha}}F_{i\alpha}(z) = 0,$$

where z is derived via the canonical transformation of x via Equation 3.16.

The proof of Lemma 3.4.1 crucially relies on the fact that the self-loops are antisymmetric, $(A^{ii})^\top = -A^{ii}$. The complete proof of this result can be found in A.2.

3.4.3 Bounded Orbits

Proving that the orbits of $z(0)$ are bounded is more complicated. To do so, we show that for any initial interior point $x(0)$, the orbits produced by replicator dynamics stay on the interior of the simplex. That is, there exists a fixed parameter $\epsilon > 0$ such that for any player $i \in V$ and strategy $\alpha \in \mathcal{S}_i$, $\epsilon \leq x_{i\alpha} \leq 1 - \epsilon$. Then, $|z_{i\alpha}|$ is clearly bounded since $z_{i\alpha} = \ln(x_{i\alpha}/x_{1\alpha})$. The bounded orbits property is shown in Lemma 3.4.2.

Lemma 3.4.2. *Consider an N-player rescaled zero-sum polymatrix game such that for positive coefficients $\{\eta_i\}_{i\in V}$, $\sum_{i\in V}\eta_i u_i(x) = 0$ for $x \in \mathcal{X}$. If the game has an interior Nash Equilibrium x^*, then $\Phi(t) = \sum_{i\in V}\sum_{\alpha\in\mathcal{S}_i}\eta_i x^*_{i\alpha}\ln x_{i\alpha}$ is time-invariant, meaning that $\Phi(t) = \Phi(0)$ for $t \geq 0$. Hence, orbits from any interior initial condition $x(0)$ remain on the interior of the simplex.*

To show $\Phi(t)$ is time-invariant, we prove that the time derivative of the function is equal to zero. From the given form of the replicator dynamics and the rescaled zero-sum property of the polymatrix game, we obtain $\dot{\Phi}(t) = \sum_{i\in V}\sum_{j:(i,j)\in E}\eta_i(x^*_i)^\top A^{ij}(x_j - x^*_j)$

nearly immediately, where the sum over edges describes how the rescaled utility of agent $i \in V$ changes at her equilibrium strategy when the rest of the players are allowed to deviate. To continue, we draw a key connection to a fascinating result regarding the payoff structure of zero-sum polymatrix games.

[32] proved there exists a payoff preserving transformation from any zero-sum polymatrix game to a pairwise constant-sum polymatrix game. We translate this result to rescaled zero-sum polymatrix games. The primary implication is that the change in player i's rescaled utility at equilibrium when all other players connected to i deviate is equal to the change in player j's rescaled utility from deviating while all other players connected to j remain in equilibrium. This is a direct consequence of the fact that the game is equivalent to a pairwise constant-sum game. Explicitly, we prove that $\dot{\Phi}(t) = \sum_{j\in V} \sum_{i:(j,i)\in E} \eta_j (x_j^* - x_j)^\top A^{ji} x_i^*$ and conclude $\dot{\Phi}(t) = 0$ since x^* is an interior Nash equilibrium, which means $u_{j\alpha}(x^*) = u_j(x^*)$ for $\alpha \in \mathcal{S}_j$ and any linear combination. The complete proof of this result can be found in A.3.

Let us discuss the implications of this result. Firstly, we need to explain how the constant of motion $\Phi(t) = \sum_{i\in V} \sum_{\alpha\in\mathcal{S}_i} \eta_i x_{i\alpha}^* \ln x_{i\alpha}$ implies that the orbits remain bounded away from the boundary. To see why this is the case, let x be an interior point which is not an equilibrium. That is, each $x_i \in \operatorname{int}(\Delta^{n-1})$. Let $\gamma(x)$ be the forward orbit of x i.e.,

$$\gamma(x) = \{\phi^t(x) : \ t \geq 0\}$$

Then, Lemma 3.4.2 implies that for any $y \in \gamma(x)$,

$$-c = \Phi(t) = \sum_{i\in V} \sum_{\alpha\in\mathcal{S}_i} \eta_i x_{i\alpha}^* \ln x_{i\alpha} = \sum_{i\in V} \sum_{\alpha\in\mathcal{S}_i} \eta_i x_{i\alpha}^* \ln y_{i\alpha} < 0$$

since $\Phi(t)$ is a constant of motion. Thus, for any i and any $y \in \gamma(x)$, $\sum_{\alpha\in\mathcal{S}_i} \eta_i x_{i\alpha}^* \ln y_{i\alpha} \in [-c, 0]$, since $\sum_{\alpha\in\mathcal{S}_i} \eta_i x_{i\alpha}^* \ln y_{i\alpha} \leq 0$ for any y and i. Hence, for any $\beta \in \mathcal{S}_i$,

$$-c \leq -c - \sum_{\alpha\neq\beta} x_{i\alpha}^* \ln y_{i\alpha} < x_{i\beta}^* \ln y_{i\beta}$$

since $\sum_{\alpha\neq\beta} x_{i\alpha}^* \ln y_{i\alpha} \leq 0$. This implies that

$$y_{i\beta} \geq \exp\big(-c/x_{i\beta}^*\big).$$

Let $\varepsilon = \min_{i\in V,\beta\in\mathcal{S}_i} \exp\big(-c/x_{i\beta}^*\big)$. This implies that for any $y \in \gamma(x)$, player $i \in V$ and strategy $\beta \in \mathcal{S}_i$, $y_{i\beta} \geq \varepsilon > 0$. This, in turn, implies that $\gamma(x)$ is bounded away from the boundary.

Another important point which we will use in our simulations is that the constant of motion from Lemma 3.4.2 can equivalently be written as:

$$\Phi(x) = -\sum_{i\in V} \eta_i (\mathrm{KL}(x_i^* || x_i) - h(x_i^*))$$

where $h(\cdot)$ denotes the entropy function, i.e. $h(x_i^*) = \sum_{\alpha\in\mathcal{S}_i} x_{i\alpha}^* \log x_{i\alpha}^*$ and $\mathrm{KL}(\cdot||\cdot)$

denotes the Kullback-Leibler (KL) divergence, i.e. $\mathrm{KL}(x_i^* \| x_i) = \sum_{\alpha \in \mathcal{S}_i} x_{i\alpha}^* \ln\left(\frac{x_{i\alpha}^*}{x_{i\alpha}}\right)$. Since $\sum_{i \in V} \eta_i h(x_i^*)$ is a constant, it does not change the time-invariant property, and hence the weighted sum of KL divergences is itself a constant of motion.

Corollary 3.4.2. *Under the assumptions of Lemma 3.4.2, $\Psi(t) = \sum_{i \in V} \eta_i(\mathrm{KL}(x_i^* || x_i))$ is also a constant of motion.*

Hence, we have established that a 'weighted' version of the sum of KL-divergences based on the rescaling of the polymatrix game is a constant of motion. To see this result in action, we once again look to the time-evolving RPS game studied by [114]. Figure 3.5 shows the weighted sum of KL-divergences from the equilibrium of the rescaled zero-sum RPS game. The figure shows that the sum is indeed a constant of motion, despite the population and environment's co-evolution. Importantly, while a constant of motion exists for the time-evolving generalized rock-paper-scissors game, it not the obvious choice – the values of η are determined via the reduction in Section 3.3.1.

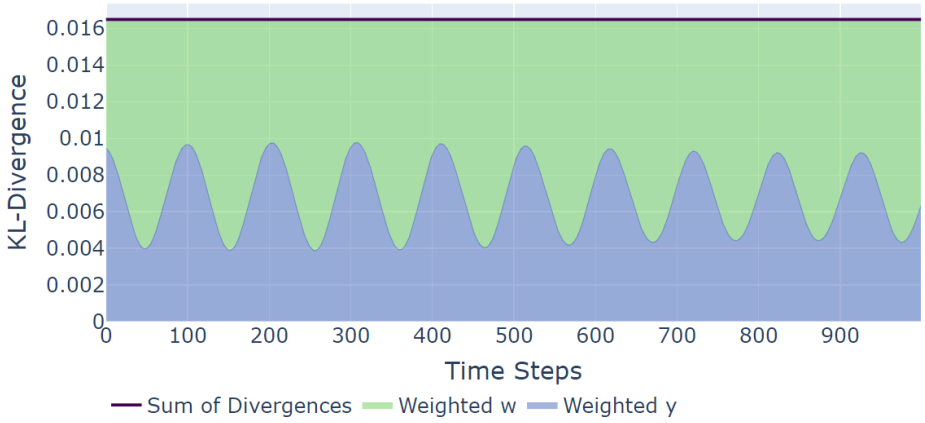


FIGURE 3.5: Constant sum of KL-divergence for time-evolving generalized rock-paper-scissors game.

With the intermediate results of volume preservation and bounded orbits, we are ready to prove our main result:

Proof of Theorem 3.4.1. The proof follows directly from Lemma 3.4.1, Lemma 3.4.2, and Theorem 2.3.1. Indeed, the dynamics in (3.17) are Poincaré recurrent since from Lemma 3.4.1 they are volume preserving and from Lemma 3.4.2 the orbits are bounded. This property in the cumulative payoff space carries over to the dynamics in the strategy space from (3.3) since the transformation is a diffeomorphism. □

3.4.4 Simulations

Theorem 3.4.1 states that any population/environment dynamics which can be captured via a *rescaled zero-sum game* (no matter the complexity of such a description) exhibit a type of *cyclic behavior* known as Poincaré recurrence. Indeed, the trajectories shown in Figure 3.1 from the time-evolving generalized RPS game of [114] are cyclic in nature. Specifically, Figure 3.1c shows the coevolution of the system for a fixed initial condition.

We plot the joint trajectory of the first two strategies for both the population y and environment w, which creates a 4D space where the color legend acts as the final dimension. The simulation demonstrates that as the initial conditions move closer to the interior equilibrium, the trajectories themselves remain bounded within a smaller region around the equilibrium, which confirms the bounded regret property of the dynamics from Proposition 3.5.1.

Poincaré sections, otherwise known as Poincaré maps, are a method of visualizing the dynamic behavior of potentially chaotic systems. In Figure 3.6a, we present a Poincaré section developed from simulating 10 trajectories of initial conditions $\{[0.5, 0.01k, 0.5 - 0.01k, 0.5, 0.25, 0.25]\}_{k=1}^{10}$ and taking the points that intersect the hyperplane $y_2 - y_1 - w_2 + w_1 = 0$. In Figure 3.6b, we show another example of a Poincaré section by simulating 10 trajectories using initial conditions $\{[1/3, 0.03k, 2/3 - 0.03k, 1/3, 1/3, 1/3]\}_{k=1}^{10}$ and marking where the trajectories intersected the hyperplane $y_2 + y + 1 + w_2 + w_1 = 4/3$. The intersection points indicate the system is quasi-periodic, since they lie on closed curves no matter the (interior) initial condition.

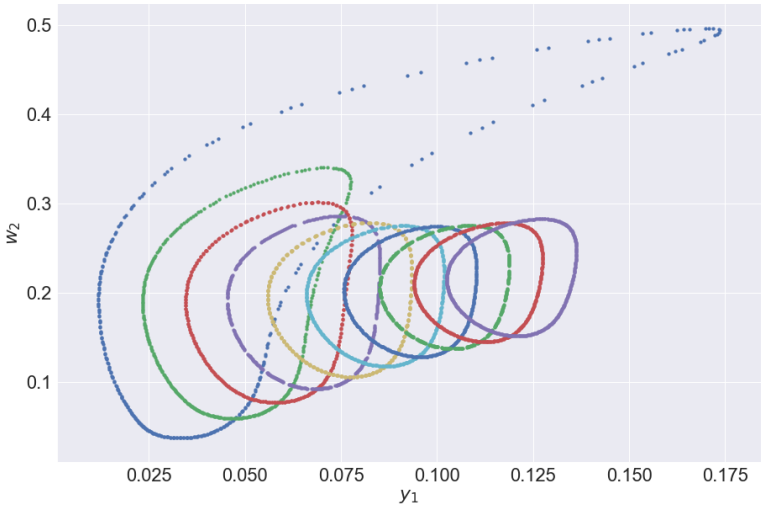
Notice that we are studying a time-evolving RPS game, which means that the mixed strategies are a vector of length 3 for both y and w. Visualizing the multidimensional system behavior is also an interesting task, since we want to retain the maximum amount of information. Thus, we generated Figure 3.7, which transforms the 3-dimensional data for players y and w respectively into two dimensions. To be precise, the transformations are given as follows:

$$y_1' = \frac{\sqrt{2}}{2}y_3 - \frac{\sqrt{2}}{2}y_2, \qquad y_2' = -\frac{1}{\sqrt{6}}y_3 - \frac{1}{\sqrt{6}}y_2 + \frac{\sqrt{2}}{\sqrt{3}}y_1$$
$$w_1' = \frac{\sqrt{2}}{2}w_3 - \frac{\sqrt{2}}{2}w_2, \quad w_2' = -\frac{1}{\sqrt{6}}w_3 - \frac{1}{\sqrt{6}}w_2 + \frac{\sqrt{2}}{\sqrt{3}}w_1$$

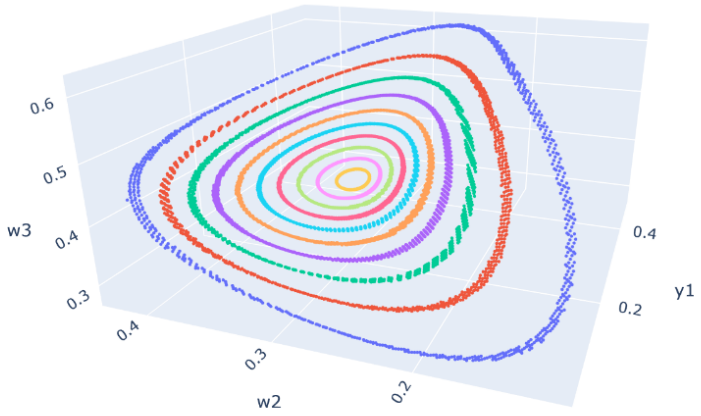
The 4-dimensional system is now visualized in the 3-dimensional plane, with color acting as the final dimension. The simulations are run for a range of initial conditions to show that when we start closer to the interior fixed point, trajectories are bounded closer to zero. These simulations were then compiled into an animation, which can be found in the supplementary code repository. Figure 3.7 and the corresponding animation are analogous to Figure 3.1c and its corresponding animation, but the transformation method allows for visualization of all 6 dimensions instead of only 4 dimensions. With this visualization, over time we are able to see that no matter the initialization, the behavior of the population and the environment stay coupled and bounded.

3.5 Time-Average Behavior, Equilibrium Computation, & Bounded Regret

In this section, we transition away from analyzing the dynamic behavior of replicator dynamics and focus on characterizing the long-term behavior along with connections to notions of equilibrium and regret. We prove that the long-term system behavior is guaranteed to satisfy a number of desirable game-theoretic metrics of consistency and optimality.



(A) Constant of motion



(B) Poincaré section

FIGURE 3.6: (a) 2D Poincaré section at $y_2 - y_1 - w_2 + w_1 = 0$ with 10 trajectories of initial conditions $\{[0.5, 0.01k, 0.5 - 0.01k, 0.5, 0.25, 0.25]\}_{k=1}^{10}$, (b) Side view of Poincaré section at $y_2 + y_1 + w_2 + w_1 = 4/3$ with 10 trajectories of initial conditions $\{[1/3, 0.03k, 2/3 - 0.03k, 1/3, 1/3, 1/3]\}_{k=1}^{10}$.

While the replicator dynamics exhibit complex dynamics and never equilibriate in rescaled zero-sum polymatrix games with interior Nash equilibrium, the time-average behavior of the dynamics is closely tied to the equilibrium. The following result shows that given the existence of a unique interior Nash equilibrium, the time-average of the replicator dynamics converges to the equilibrium and the time-average utility converges to the utility at the equilibrium.

Theorem 3.5.1. *Consider an N-player rescaled zero-sum polymatrix game that admits a unique interior Nash equilibrium x^*. The trajectory $x(t)$ produced by replicator dynamics given in* (3.3) *is such that* ***i)*** $\lim_{t\to\infty} \frac{1}{t}\int_0^t x(\tau)d\tau = x^*$ *and* ***ii)*** $\lim_{t\to\infty} \frac{1}{t}\int_0^t u_i(x(\tau))d\tau = u_i(x^*)$.

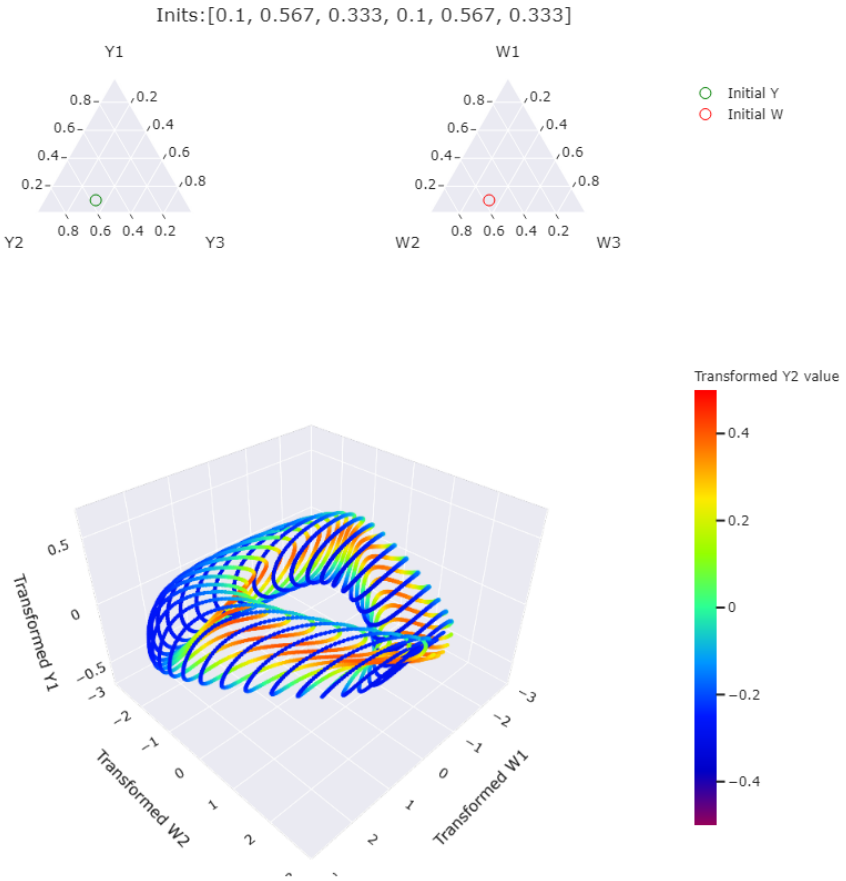
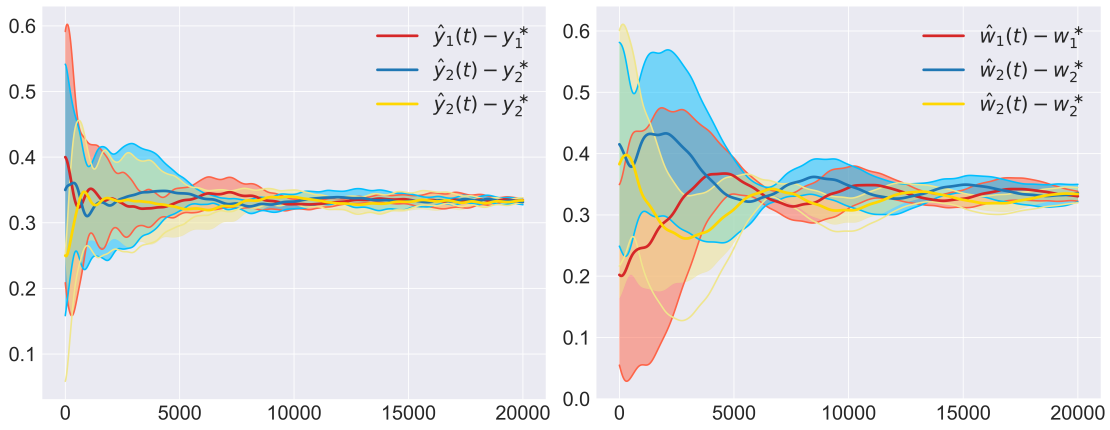


FIGURE 3.7: 4D embedding of trajectories for a range of initial conditions.

The preceding result provides a broad generalization of past results that show the time-average of replicator dynamics converges to the unique interior Nash equilibrium in zero-sum bimatrix games [85]. We remark that our proof crucially relies on Lemma 3.4.2 since the trajectory of the dynamics must remain on the interior of the simplex to guarantee there exists a bounded sequence which admits a subsequence that converges to a limit corresponding to the time-average. The proof of this result is deferred in Appendix A.4

In Figures 3.8a and 3.8b, we plot the time-average of the population y and environment w in the time-evolving generalized rock-paper-scissors game of [114], all of which converge to $1/3$ which is the equilibrium strategy.



(A) Time-avg strategy: y (B) Time-avg strategy: w

FIGURE 3.8: Time-average for y and w converging to Nash.

We now provide a polynomial time algorithm that efficiently predicts the time-average quantities even for an arbitrary networks of players. Linear programming formulations for computing and characterizing the set of Nash equilibria for zero-sum polymatrix games are known [31]. The following result extends this formulation to rescaled zero-sum polymatrix games.

Theorem 3.5.2. *Consider an N-player rescaled zero-sum polymatrix game such that for positive coefficients $\{\eta_i\}_{i\in V}$, $\sum_{i=1}^{N} \eta_i u_i(x) = 0$ for $x \in \mathcal{X}$. The optimal solution of the following linear program is a Nash equilibrium of the game:*

$$\min_{x\in\mathcal{X}} \left\{ \sum_{i=1}^{n} \eta_i v_i |\ v_i \geq u_{i\alpha}(x),\ \forall\, i \in V,\ \forall\, \alpha \in \mathcal{S}_i \right\}$$

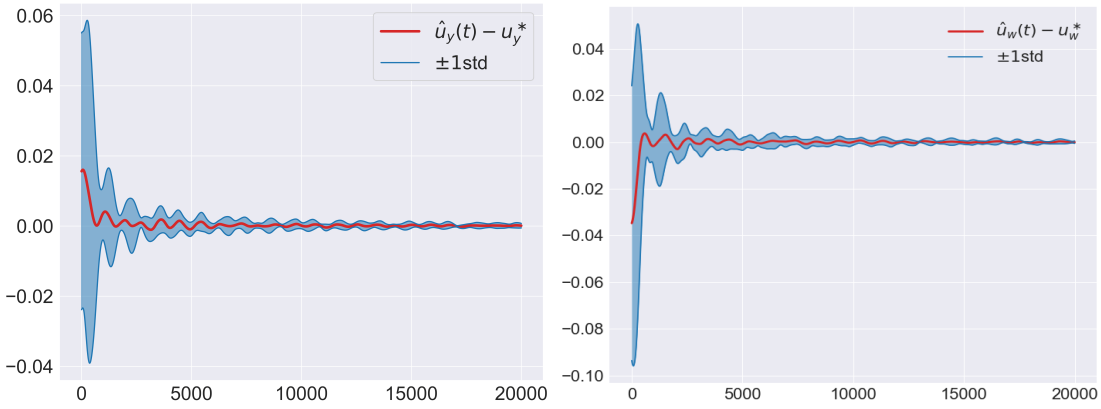
The proof of this result is deferred to Appendix A.5.

It cannot be universally expected that an interior equilibrium exists or that players are fully rational and obey a common learning rule. Similarly, players may not always be able to determine an equilibrium strategy *a priori* depending on the information available. This motivates an evaluation of the trajectory of a player who is oblivious to opponent behavior. We consider a notion of *regret* for a player. That is, the time-averaged utility difference between the mixed strategies selected along the learning path $t \geq 0$ and the fixed strategy that maximizes the utility in hindsight. Even in polymatrix games (with self-loops), the regret of replicator dynamics stays bounded.

Proposition 3.5.1. *Any player following the replicator dynamics* (3.3) *in an N-player polymatrix game (with self-loops) achieves an $\mathcal{O}(1/t)$ regret bound independent of the rest of the players. Formally, for every trajectory $x_{-i}(t)$, the regret of player $i \in V$ is bounded as follows for a player-dependent positive constant Ω_i,*

$$\mathrm{Reg}_i(t) := \max_{y\in\mathcal{X}_i} \frac{1}{t} \int_0^t \left[u_i(y, x_{-i}(\tau)) - u_i(x(\tau))\right] d\tau \leq \frac{\Omega_i}{t}.$$

The proof of this proposition closely mirrors more general arguments in [117], and we present this proof in Appendix A.6 for completeness. Similarly in Figures 3.9a and 3.9b, we plot the time-average utilities of the population y and environment w, showing that they converge to the equilibrium utility. In the game we consider, the equilibrium utility is zero, a fact which emerges visually in the simulation.



(A) Time-avg utility: y (B) Time-avg utility: w

FIGURE 3.9: Time-average utility converging with bounded regret.

3.6 Additional Simulations

The goal of this section is to highlight other empirically observed properties outside the established theoretical results, and present simulations beyond the time-evolving RPS game. We show that our results for recurrence and time-average convergence to Nash still hold even in large network games, a testament to the power of our generalized reduction.

3.6.1 Simulations of 5-player Rescaled Zero-Sum Polymatrix Game

As an initial foray into systems with greater than 2 players (i.e. more than one population and one environment), we simulated the rescaled zero-sum polymatrix game depicted in Figure 3.10 where each of the 5 players has 3 actions. As shown in Figure 3.11, the weighted sum of KL-divergences of each player's strategy from the equilibrium is a constant of motion, demonstrating the bounded orbits property of Lemma 3.4.2. In the simulation, we set $\mu_1 = 0.1, \mu_2 = 0.5, \mu_3 = 0.8, \mu_4 = 0.5$ and the initial conditions for the 5 players are $[0.3, 0.4, 0.3]$, $[0.2, 0.1, 0.7]$, $[0.5, 0.3, 0.2]$, $[0.7, 0.2, 0.1]$ and $[0.4, 0.2, 0.4]$ respectively. We also include the time averages of the trajectories and utility for player x_3 in Figure 3.12. The plots show that the player's trajectories converge to the interior Nash equilibrium at $(1/3, 1/3, 1/3)$ and that the time-average utility converges to the utility at this interior Nash equilibrium.

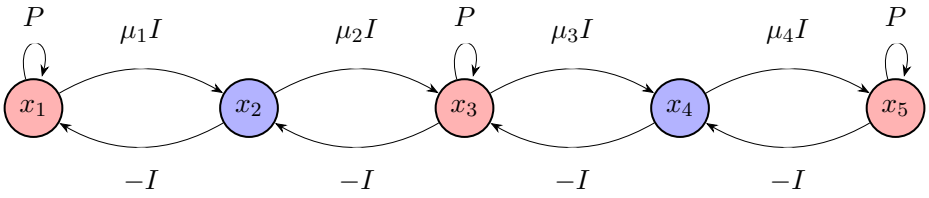


FIGURE 3.10: Five-node polymatrix game used in our small-scale simulations. Each node represents a player, with different initial strategies and values of μ_i.

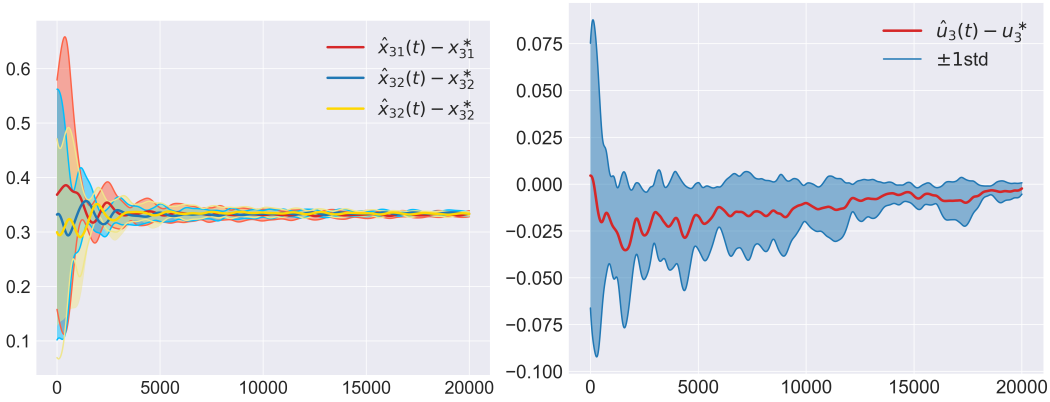
FIGURE 3.11: Weighted KL-divergence for five player time-evolving RPS game for 1000 iterations.

3.6.2 Simulations of Large-Scale Rescaled Zero-Sum Polymatrix Games

To show the potential for the scalability of this theory, we simulated larger systems with more complex dynamics between players, and experimentally confirm that our theorems still hold in these contexts.

Butterfly Game Simulation. In the simulations up to this point, we have been looking at rescaled zero-sum polymatrix games of a particular structure. Indeed, these simulations are extensions to the example polymatrix game as defined in Figure 3.3. However, our theory extends to more than just graphs of that form. So long as the graphs are formed from the basic building blocks in Figure 3.2, we will see similar results. We performed experiments using extensions of the 'butterfly' graph shown in Figure 3.13, where each red node is a population of species and each blue node is an environment. This describes a situation where there are many species affecting a single environment, and the competition between species can be reduced to self-loop games. The connections between blue and red nodes represent bimatrix games of the form $(-I, \mu I)$ and the self-loops represent self-play zero sum games. For the simulations, we use RPS as the self-play game.

As shown in Figures 3.14 and 3.15, we see that despite the much more complex



(A) Time-average strategy: x_3

(B) Time-average utility: x_3

FIGURE 3.12: (a) Time-average trajectories for x_3 showing convergence to Nash. (b) Time-average utility convergence for x_3 player with bounded regret. Initial conditions are fixed values chosen uniformly at random on the simplex for x_1, x_2, x_4, x_5 and for x_3, they take values in $(z, 0.75 - z, 0.25)$ for each $z \in \{0.1 + \frac{2k}{30}, k \in \{0, \ldots, 9\}\}$.

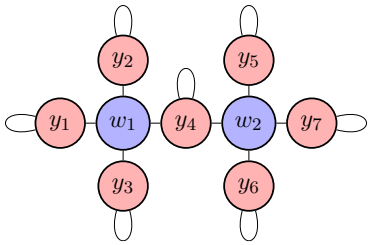


FIGURE 3.13: Two clusters of nodes that join together to form a 'butterfly' structure. Self-loops represent RPS self-play games, while edges between nodes represent $(I, -I)$. The *red* nodes denote a population of species, while the *blue* nodes stand for an environment.

graph structure and many nodes, and although each player-specific divergence term $\eta_i \mathrm{KL}(x_i^* || x_i(t))$ fluctuates, the weighted sum $\sum_{i \in V} \eta_i \mathrm{KL}(x_i^* || x_i(t))$ remains constant..

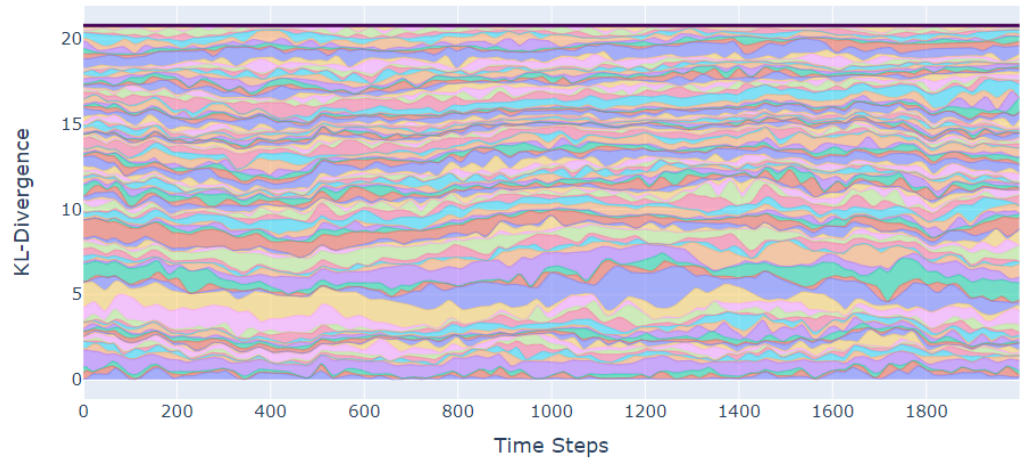


FIGURE 3.14: Weighted KL divergence for 25 cluster (100 player) time-evolving zero-sum game with 'butterfly' structure.

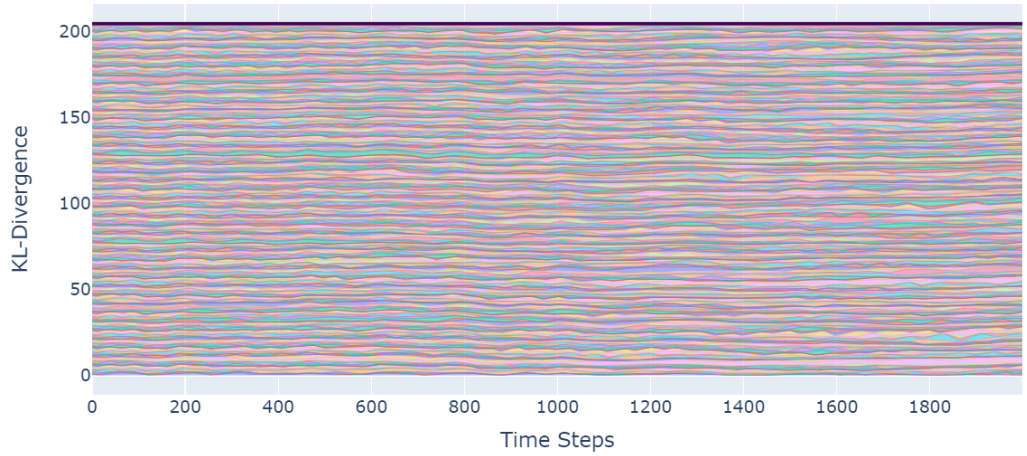


FIGURE 3.15: Weighted KL-divergence for 100 cluster (400 player) time-evolving zero-sum game with 'butterfly' structure.

Pikachu Image Simulation. In order to obtain the initial conditions of this simulation, we used an 1200x1200 pixel image of Pikachu and converted that image into an 8x8 array of RGB values. Then, we convert these RGB values into a set of initial conditions for the replicator dynamics. In order to see more obvious differences between colors as the strategies evolve, we applied a sigmoid function centered at 0.5 to each RGB value.

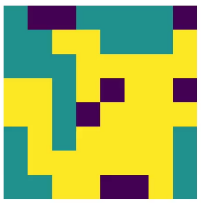


FIGURE 3.16: 8×8 grid of colors generated by sigmoid function

Due to the application of the sigmoid function, we expect to see a mostly dark blue or mostly bright yellow pixel whenever the respective strategy is far from the central value of 0.5. Indeed, in Figure 3.17 we see that the grid very quickly transforms into something that does not resemble the original Pikachu at all. After a (large) number of iterations, the recurrence property causes the Pikachu (or at least, something that looks similar to Pikachu) to reappear.

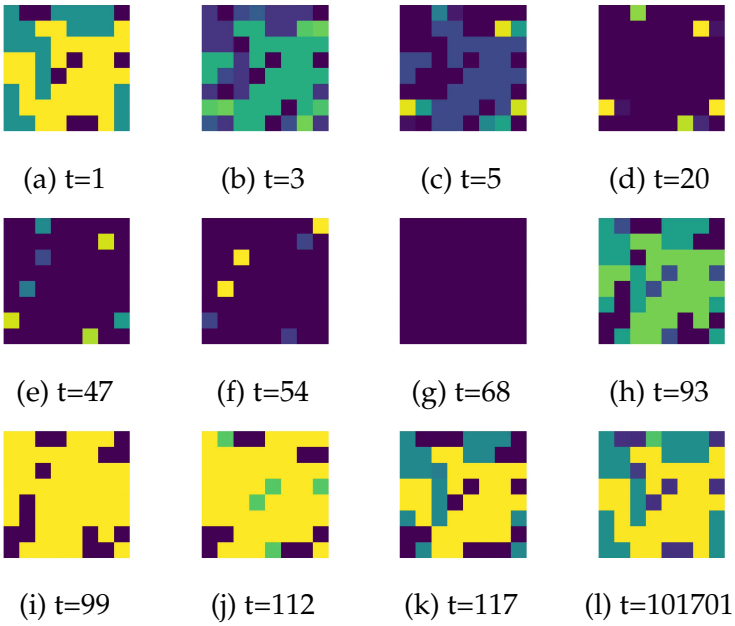


FIGURE 3.17: Sequence of Pikachu images showing approximate recurrence in an 8×8 zero-sum polymatrix game, where the changing color of each pixel on the grid represents the strategy of the player over time.

To generate Figure 3.17, we scale-up the game structure from Mai et al. [114] to 64 nodes. This is a relatively dense graph, where the initial condition of each player informs the RGB value of a corresponding pixel on a grid. If the system exhibits Poincaré recurrence, we should eventually see similar patterns emerge as the pixels change color over time (i.e., as their corresponding strategies evolve). In general, an upper bound on the expected time to see recurrence in such a system is exponential in the number of agents. As observed in Figure 3.17, the system returns near the initial image in the first

several hundred iterations, but takes more than 100k iterations for a clearer Pikachu to reappear.

An additional point to note is that our code for simulating such large-scale rescaled zero-sum polymatrix games was refactored from the previous, smaller scale code such that it now works for a general number of nodes N. Hence, future simulations could potentially model multiagent systems at a much wider level than shown in our work.

3.7 Conclusion

In this chapter, we have shown that systems in which populations of dynamic agents interact with environments that evolve as a function of the agents themselves can equivalently be modeled as polymatrix games. For the class of rescaled zero-sum games, we prove replicator dynamics are Poincaré recurrent and converge in time-average to the equilibrium, while experiments show the complexity of systems to which the results apply. An intriguing direction for future work is to find other applications which are subsumed by our framework. Moreover, a future direction for theoretical research is to study of games that evolve exogenously instead of only endogenously.

Moreover, there are several exciting applications where our theory has relevance. Google DeepMind trains populations of AI agents against each other and computes win probabilities in heads-up competition resulting in a symmetric constant-sum game [14, 44]. Up to a shift by an all 0.5 matrix, these are exactly anti-symmetric self-loop games connecting a population of users (programs) to itself as the programs are trying to out-compete each other. The game always remains (anti)-symmetric, but the payoff entries change as stronger agents replace old agents. While our model cannot capture the system fully, we can create an abstract model of it. The self-loop zero-sum game is the initialization of the system and is equal to the original anti-symmetric empirical zero-sum game. There is another zero-sum game between the population and a meta-agent which simulates the reinforcement policy that chooses which programs get replaced and thus generates a new empirical zero-sum payoff matrix. We can mimic this randomized choice of the policy as a mixed strategy that chooses a convex combination from a large number of possible empirical zero-sum payoff matrices. One of these payoff matrices is the all-zero matrix, and the initial strategy of the reinforcement policy chooses that game with high probability at time zero, so that the population is at the start of the process effectively playing just their original empirical game. As such, our results in this chapter provide some theoretical justification for the preservation of diversity and for the satisfying empirical performance.

To conclude, our framework is able to capture a wide range of endogenously time-evolving systems, and our theoretical results provide insights into the dynamical behavior of replicator dynamics in this class of games.

Chapter 4

Online Learning in Periodically Evolving Zero-Sum Games

This chapter is taken from (with minor modifications) our paper *'Online Learning in Periodic Zero-Sum Games'* [62].

4.1 Introduction

In the previous chapter, we studied a class of time-evolving games wherein *endogenous evolution* between populations of species and their environments was captured via a reduction to rescaled zero-sum polymatrix games. However, this analysis was more focused on establishing results with an evolutionary game theoretic slant – in that setting replicator dynamics are widely considered to be the most important learning dynamic. However, if we shift our focus to the *learning dynamic* which is used, there are many other dynamics which have been shown to exhibit recurrence and time-average convergence in recent years. For instance, [20, 117, 127, 135, 136, 138, 173] show that many continuous-time online learning dynamics exhibit recurrence in zero-sum games (and variants thereof). Moreover, these results have acted as fundamental building blocks in order to understand the limiting behavior of their discrete-time variants [11, 13, 39, 40, 119].

Despite the plethora of emerging results in zero-sum games, our understanding of the dynamical behavior of continuous dynamics in general time-evolving zero-sum games is less understood. Our result for Poincaré recurrence in the previous chapter is focused on a specific class of *rescaled* zero-sum polymatrix games which encode a wider class of endogenously-evolving zero-sum games. Thus, a natural extension has to do with the following questions:

> *What other forms of time-evolving games can be modeled and studied? How do continuous-time dynamics beyond replicator behave in these time-evolving games?*

In this chapter, we answer these questions and once again broaden the horizon of continuous-time learning in games beyond the static regime. Specifically, we consider a class of games which model a commonly observed phenomenon in the real-world – periodically-evolving games. Specifically, we focus on periodic zero-sum games with

a time-invariant equilibrium, which is a class of games that model exogenous time-evolution instead of endogenous time-evolution. In a periodic zero-sum game, the payoffs that dictate the game are both T-periodic and zero-sum at all times. We consider both periodic zero-sum *bilinear* games on infinite, unconstrained strategy spaces and periodic zero-sum matrix games (along with network generalizations thereof) on finite strategy spaces. The goal of this work is thus to evaluate the robustness of the archetypal online learning behaviors in zero-sum games (i.e. Poincaré recurrence and time-average equilibrium convergence) to this natural model of game evolution.

The time-evolving game model we study can be seen as a generalization of standard repeated game formulations. A time-invariant game is a trivial version of a periodic game, in which case we recover the repeated static game setting. For a general periodic zero-sum game with period T, each stage game now is chosen according to a fixed length T sequence of games, capturing interactions between the players and time-dependent payoffs.

Periodic zero-sum games can also fit into the frameworks of multi-agent contextual games [153] and dynamic games (see, e.g., [17]). In a multi-agent contextual game [153], the environment selects a context from a set before each round of play and this choice defines the game that is played. Periodic zero-sum games can be seen as a multi-agent contextual game where the environment draws contexts from the available set in a T-periodic fashion with each context corresponding to a zero-sum game with a common equilibrium. In the class of dynamic games, there is a game state on which the payoffs may depend that evolves with dynamics. Periodic zero-sum games can be interpreted as a dynamic game where the state transitions do not depend on the strategies of the players, the state is T-periodic, and the payoffs are completely defined by the state.

In addition, the periodic zero-sum game model also allows us to capture competitive settings where exogenous environmental variations manifest in a periodic/epochal fashion. This naturally occurs in market competitions where time-of-day effects, week-to-week trends, and seasonality can dictate the game between players. To illustrate this point, consider a competition between service providers that wish to maximize their users, while the total market size evolves seasonally over time. This evolution affects the utility functions, even if the fundamentals of the market, and consequently the equilibrium, remain invariant.

In conclusion, our model generalizes the standard repeated game formulation while also being a realistic and natural model of a repeated competition between players that depends on exogenous environmental variations such as time-of-day effects, week-to-week trends, and seasonality.

4.1.1 Our Contributions

In this chapter, for the classes of periodic zero-sum bilinear games and periodic zero-sum polymatrix games with time-invariant equilibria, we investigate the day-to-day and time-average behaviors of continuous-time gradient descent-ascent (`GDA`) and follow-the-regularized-leader (`FTRL`) learning dynamics respectively. This study highlights the careful attention that must be given to the dynamical systems in time-evolving (specifically, periodic) zero-sum games which preclude standard proof techniques for

Poincaré recurrence, while also revealing that intuition from existing results on static zero-sum games can be totally invalidated even by simple examples in periodic zero-sum games.

A key technical challenge we face is that the dynamical systems which emerge from learning dynamics in periodic zero-sum games correspond to *non-autonomous* ordinary differential equations, whereas learning dynamics in static zero-sum games correspond to *autonomous* ordinary differential equations. Consequently, the usual proof methods from static zero-sum games for showing Poincaré recurrence are insufficient on their own in periodic zero-sum games. We overcome this challenge by delicately piecing together properties of periodic systems to construct a discrete-time autonomous system that we are able to show is Poincaré recurrent. This approach allows to prove both the `GDA` and `FTRL` learning dynamics are Poincaré recurrent in the respective classes of periodic zero-sum games. Finally, we show both periodicity and a time-invariant equilibrium are necessary for such results in evolving games.

Given that Poincaré recurrence provably generalizes from static zero-sum games to periodic zero-sum games, it may be expected that the time-average strategies in periodic zero-sum games converge to the time-invariant equilibrium as in static zero-sum games. Surprisingly, we show that counterexamples can be constructed to this intuition even in the simplest of periodic zero-sum games. In particular, we prove the negative result that the time-average `GDA` and `FTRL` strategies do not necessarily converge to the time-invariant equilibrium in the respective classes of zero-sum games.

Despite the negative result for time-average strategy convergence, in the special case of periodic zero-sum bimatrix games we are able to show a complimentary positive result on the time-average utility convergence. Specifically, we show that the time-average utilities of the `FTRL` learning dynamics converge to the average of the equilibrium utility values of all the zero-sum games included in a single period of these time-evolving games.

4.2 Preliminaries and Definitions

In this chapter we study two major classes of games with different strategy sets, namely periodic zero-sum bilinear games with *continuous* strategy spaces, and periodic polymatrix zero-sum games with *finite* strategy spaces. In this section, we introduce these games and reiterate the necessary dynamical systems theory required to establish our results. Note that the dynamical systems preliminaries we will utilize in this chapter are more involved than before, since we deal with non-autonomous systems instead of autonomous systems.

4.2.1 Periodic Zero-Sum Games with Continuous Strategy Spaces

The first setting we study is intrinsically connected to min-max optimization. From a game theoretic perspective, the problem of interest we study is the following unconstrained max-min problem. Given a matrix $A \in \mathbb{R}^{n_1 \times n_2}$:

$$\max_{x_1 \in \mathbb{R}^{n_1}} \min_{x_2 \in \mathbb{R}^{n_2}} x_1^\top A x_2,$$

where there are two players whose strategy spaces are given by $\mathbb{R}^{n_1}$ and $\mathbb{R}^{n_2}$ respectively. The zero-sum game is then defined by the pair of payoff matrices $\{A, -A^\top\}$. Player 1 seeks to maximize their utility function $u_1(x_1, x_2) = x_1^\top A x_2$ while player 2 maximizes the utility $u_2(x_1, x_2) = -x_2^\top A^\top x_1$. The game is zero-sum since for any $x_1 \in \mathbb{R}^{n_1}$ and $x_2 \in \mathbb{R}^{n_2}$, the sum of utility over each player is zero. We call this class of games *zero-sum bilinear games*. In zero-sum bilinear games, a *Nash equilibrium* corresponds to a joint strategy (x_1^*, x_2^*) such that for each player i and $j \neq i$, $u_i(x_i^*, x_j^*) \geq u_i(x_i, x_j^*)$, $\forall x_i \in \mathbb{R}^{n_i}$. Note that $(x_1^*, x_2^*) = (\mathbf{0}, \mathbf{0})$ is always a (trivial) Nash equilibrium of a zero-sum bilinear game.

This formulation has been extensively studied in the literature due to its important applications in robust optimization and machine learning. More recently, in generative adversarial networks (GANs) [74], zero-sum bilinear games capture simple GANs formulations and provide useful convergence results [49, 69].

Periodic Zero-Sum Bilinear Games. We study the continuous-time GDA learning dynamics in a class of games we call periodic zero-sum bilinear games. The key distinction from a typical static zero-sum bilinear game is that the payoff matrices for each player $\{A, -A^\top\}$ are no longer fixed. Instead, the payoff matrix may change over time, so long as game remains zero-sum and the continuous-time sequence of payoffs is periodic. The next definition formalizes this class of games.

Definition 4.2.1 (Periodic Zero-Sum Bilinear Game)**.** *A periodic zero-sum bilinear game is an infinite sequence of zero-sum bilinear games $\{A(t), -A(t)^\top\}_{t=0}^{\infty}$ in which the player set and strategy spaces are fixed and the payoff matrix is such that $A(t) = A(t+T)$ for a finite period T and all $t \geq 0$. Note that in such a game, $(0, 0)$ is always a time-invariant Nash equilibrium. Furthermore, we assume that the dependence of the payoff entries on time is smooth everywhere except for a finite set of points.*

4.2.2 Periodic Zero-Sum Games with Finite Strategy Spaces

The second setting we study is an extension of the work in Chapter 3. In particular, previously we studied endogenously-evolving games where the game itself can be modeled as a player in a *rescaled zero-sum polymatrix game*. We then analyze the convergence properties of replicator dynamics in the *static* polymatrix game. In this chapter, we instead study polymatrix games which are intrinsically periodic in nature. Like before, let us briefly reiterate the definition of zero-sum polymatrix games.

Zero-Sum Polymatrix Games. An N-player polymatrix game is defined by an undirected graph $G = (V, E)$ where V is the player set and E is the edge set where a bimatrix game is played between the endpoints of each edge. Each player $i \in V$ has a finite set of pure actions $\mathcal{S}_i = \{1, \ldots, n_i\}$ which can be selected at random from a distribution x_i called a mixed strategy. The mixed strategy set of player $i \in V$ is the simplex in $\mathbb{R}^{n_i}$ denoted by $\mathcal{X}_i = \Delta^{n_i - 1} = \{x_i \in \mathbb{R}^{n_i}_{\geq 0} : \sum_{\alpha \in \mathcal{S}_i} x_{i\alpha} = 1\}$ where $x_{i\alpha}$ denotes the probability of action $\alpha \in \mathcal{S}_i$. The joint strategy space is denoted by by $\mathcal{X} = \Pi_{i \in V} \mathcal{X}_i$.

The bimatrix game on edge (i, j) is described using a pair of matrices $A^{ij} \in \mathbb{R}^{n_i \times n_j}$ and $A^{ji} \in \mathbb{R}^{n_j \times n_i}$. The utility or payoff of player $i \in V$ under the strategy profile $x \in \mathcal{X}$ is given by $u_i(x) = \sum_{j:(i,j) \in E} x_i^\top A^{ij} x_j$ and corresponds to the sum of payoffs

from the bimatrix games the player participates in. We further denote by $u_{i\alpha}(x) = \sum_{j:(i,j)\in E}(A^{ij}x_j)_\alpha$ the utility of player $i \in V$ under the strategy profile $x = (\alpha, x_{-i}) \in \mathcal{X}$ for $\alpha \in \mathcal{S}_i$. The game is called zero-sum if $\sum_{i\in V} u_i(x) = 0$ for all $x \in \mathcal{X}$. In other words, each bimatrix edge game need not be zero-sum in a zero-sum polymatrix game.

A *Nash equilibrium* in a polymatrix game is a mixed strategy profile $x^* \in \mathcal{X}$ such that for each player $i \in V$, $u_i(x_i^*, x_{-i}^*) \geq u_i(x_i, x_{-i}^*),\ \forall x_i \in \mathcal{X}_i$. A Nash equilibrium is said to be an interior if $\text{supp}(x_i^*) = \mathcal{S}_i\ \forall i \in V$ where $\text{supp}(x_i^*) = \{\alpha \in \mathcal{S}_i :\ x_{i\alpha} > 0\}$ is the support of $x_i^* \in \mathcal{X}_i$.

Periodic Zero-Sum Polymatrix Games. We analyze continuous-time `FTRL` learning dynamics, a generalization of replicator dynamics, in a a class of games called periodic zero-sum polymatrix games. This class of games is defined such that the payoffs of the edge games evolve periodically. Moreover, we only consider periodic evolutions such that all games in the infinite sequence of polymatrix games admit a common interior Nash equilibrium for all time. The following definition formalizes the games we study in finite strategy spaces.

Definition 4.2.2 (Periodic Zero-Sum Polymatrix Game)**.** *A periodic zero-sum polymatrix game is an infinite sequence of zero-sum polymatrix games $\{G(t) = (V(t), E(t))\}_{t=0}^{\infty}$ in which the set of players, strategy spaces, and edges are fixed and each bimatrix game on an edge (i, j) is such that $A^{ij}(t) = A^{ij}(t + T)$ and $A^{ji}(t) = A^{ji}(t + T)$ for some finite period T and all $t \geq 0$. We assume there is a common interior Nash equilibrium $x^* \in \mathcal{X}$ of the polymatrix game $G(t)$ for all $t \geq 0$. Furthermore, we assume that the dependence of the payoff entries on time is smooth everywhere except for a finite set of points.*

4.2.3 Non-Autonomous Dynamical Systems

We now cover additional concepts from dynamical systems theory that will help us analyze learning dynamics in periodic zero-sum games and prove Poincaré recurrence. Unlike in the previous chapter where we proved recurrence for static polymatrix games, careful attention must be given to the dynamical systems preliminaries in this work, since the systems we study are *non-autonomous* whereas typical recurrence analysis in the study of learning in games (and in the previous chapter) deals with autonomous dynamical systems. To see intuitively the difference, an autonomous dynamical system depends only on the time which has elapsed since starting, whereas a non-autonomous system depends on the actual time t itself. As a result, non-autonomous dynamical systems are able to mathematically capture changes from outside the system via evolutionary adaptation, periodic shifts or even random noise. As such, additional work is required to establish the necessary properties of such systems so that they are amenable to recurrence analysis.

Flows. Consider an ordinary *non-autonomous* differential equation $\dot{x} = f(t, x)$ on a topological space X. We can define the *flow* $\phi : \mathbb{R} \times X \to X$ of a dynamical system $\dot{x}$, for which the following holds: (i) $\phi(t, \cdot) : X \to X$, often denoted $\phi^t : X \to X$, is a homeomorphism for each $t \in \mathbb{R}$, (ii) $\phi(t + s, x) = \phi(t, \phi(s, x))$ for all $t, s \in \mathbb{R}$ and all $x \in X$, (iii) for each $x \in X$, $\frac{d}{dt}|_{t=0}\phi(t, x) = f(t, x)$, and (iv) $\phi(t, x_0) = x(t)$ is the solution.

Notice now instead of considering $\dot{x} = f(x)$ like in the previous chapter, we consider $\dot{x} = f(t, x)$.

Existence and Uniqueness. We utilize Carathéodory's existence theorem to guarantee the existence of a flow for *non-autonomous* $\dot{x}$, even for discontinuous functions f which satisfy certain conditions.

Theorem (Carathéodory's existence theorem [41, 75])**.** *Consider a differential equation $\dot{x} = f(t, x)$ on a rectangular domain $R = \{(t, y)| \, |t - t_0| \leq a, |x - x_0| \leq b\}$. If f satisfies the following conditions:*

(1) $f(t, x)$ is continuous in y for each fixed t,

(2) $f(t, x)$ is measurable in t for each fixed y,

(3) there exists a Lebesgue-integrable function $m : [t_0 - a, t_0 + a] \to [0, \infty)$ such that $|f(t, x)| \leq m(t)$ for all $(t, x) \in R$,

then the differential equation has a solution. Moreover, if f is also Lipschitz continuous, meaning $|f(t, x_1) - f(t, x_2)| \leq k(t)|x_1 - x_2|$ with some Lebesgue-integrable function $k : [t_0 - a, t_0 + a] \to [0, \infty)$, then there exists a unique solution of the differential equation.

In the settings we study, the above three conditions hold. Condition 1 holds because for every fixed t, the dynamics we study (specifically `GDA` and `FTRL`) are continuous functions of their state space. Condition 2 holds because the systems we study are finite and continuous almost everywhere, and so by Lusin's theorem [113] are measurable for each fixed y. Finally, Condition 3 is always satisfied because the games we study always admit bounded orbits. Hence, it follows that a unique flow exists for all the dynamical systems studied in this chapter.

Preservation of Volume. The flow ϕ of an ordinary differential equations is called *volume preserving* if the volume of the image of any set $U \subseteq \mathbb{R}^d$ under ϕ^t is preserved, meaning that $\text{vol}(\phi^t(U)) = \text{vol}(U)$. *Liouville's theorem* states that a flow is volume preserving if the divergence of f at any point $x \in \mathbb{R}^d$ equals zero: that is, $\text{div} f(t, x) = \text{tr}(Df(t, x)) = \sum_{i=1}^{d} \frac{\partial f(t,x)}{\partial x_i} = 0$.

4.2.4 Poincaré Recurrence in Autonomous Dynamical Systems and Beyond

The standard proof method for deriving Poincaré recurrence of continuous-time dynamics in static zero-sum games [20, 117, 138] crucially rely on the static nature of the game, for which the learning dynamics amount to an autonomous dynamical system. Informally, the standard Poincaré recurrence theorem states that if an autonomous dynamical system preserves volume and every orbit remains bounded, almost all trajectories return arbitrarily close to their initial position, and do so infinitely often. We now repeat the definition of the classical Poincaré recurrence theorem from Chapter 2.3, specifying that it holds for autonomous dynamical systems.

Theorem (Poincaré Recurrence for Autonomous Continuous-Time Systems [140])**.** *If a flow preserves volume and has only bounded orbits, then for each open set almost all orbits intersecting the set intersect it infinitely often: if ϕ^t is a volume preserving flow on a bounded set $Z \subset \mathbb{R}^d$, then the nonwandering set $\Omega(\phi^t) = Z$.*

The above definition of Poincaré recurrence cannot directly be applied to the model of time-evolving zero-sum games we study as a result of the non-autonomous nature of these systems. In fact, we can construct time-evolving zero-sum games without periodic payoffs or a time-invariant equilibrium, where online learning dynamics are not Poincaré recurrent. We formalize this statement in Proposition 4.3.1.

Despite this hurdle, we show that in the natural subclass of periodically time-evolving zero-sum games defined above (payoffs oscillate periodically and equilibrium is time-invariant), we can develop proof methods to show the Poincaré recurrence of online learning dynamics. Since we have established that non-recurrent behavior of learning dynamics might arise when there is not both periodic payoffs and a time-invariant equilibrium, this is perhaps the most general class of time-evolving zero-sum games with obtainable positive results in this direction. We now provide an overview of our approach, beginning by introducing several important properties of periodic systems.

Periodic Systems and Poincaré Maps. A system $\dot{x} = f(t, x)$ is T-periodic if $f(t+T, x) = f(t, x)$ for all (x, t). Let $\phi^t : \mathbb{R}^n \to \mathbb{R}^n$ denote the mapping taking $x \in \mathbb{R}^n$ to the value at time t. For a T-periodic system, $\phi^{T+s} = \phi^s \circ \phi^T$ such that $\phi^{kT} = (\phi^T)^k$ for any integer k. The mapping $\phi^T : \mathbb{R}^n \to \mathbb{R}^n$ is called the *Poincaré map*, sometimes also known as the *mapping at a period*.

If the differential equation is well-defined for all x and has a solution for all $t \in [0, T]$, then for each initial condition (note that here we have suppressed the dependence on x_0), the Poincaré map ϕ^T defines a discrete-time autonomous dynamical system $x^+ = \phi^T(x)$. Crucially, by construction, the discrete-time system $x^+ = \phi^T(x)$ forms a subsequence of the original continuous-time system $\dot{x} = f(t, x)$.

The learning dynamics we study in periodic zero-sum games form T-periodic dynamical systems. Thus, the discrete-time autonomous dynamical system $x^+ = \phi^T(x)$ formed by the Poincaré map is key to the analysis methods we pursue. In particular, our approach is to show that this system is Poincaré recurrent, which we then use to show that the original continuous-time non-autonomous system is Poincaré recurrent. In order to achieve this, we utilize an alternative formulation of the Poincaré recurrence theorem which is applicable to autonomous discrete-time systems.

Theorem (Poincaré Recurrence for Discrete-Time Maps [16])**.** *Let (X, Σ, μ) be a finite measure space and let $\phi \colon X \to X$ be a measure-preserving map. For any $E \in \Sigma$, the set of those points x of E for which there exists $N \in \mathbb{N}$ such that $\phi^n(x) \notin E$ for all $n > N$ has zero measure. In other words, almost every point of E returns to E. In fact, almost every point returns infinitely often. That is, $\mu\left(\{x \in E : \exists\ N \text{ s.t. } \phi^n(x) \notin E \text{ for all } n > N\}\right) = 0$.*

Given this result, proving the Poincaré recurrence of the system $x^+ = \phi^T(x)$ requires verifying the volume preservation and bounded orbit properties, which then implies the measure preserving property. The following result states that if the divergence of a T-periodic vector field $f(x, t)$ is divergence free (i.e. implying that the flow ϕ^t is volume preserving), then the Poincaré map ϕ^T and the resulting discrete-time dynamical system $x^+ = \phi^T(x)$ is also volume preserving.

Theorem (Volume preservation for T-Periodic Systems [5]). *If the T-periodic system $\dot{x} = f(t, x)$ is divergence-free, then ϕ^T preserves volume.*

Similarly, if orbits of $\dot{x} = f(t, x)$ are bounded, then clearly the orbits of $x^+ = \phi^T(x)$ are bounded. Hence, to show the system $x^+ = \phi^T(x)$ is Poincaré recurrent, we prove $\dot{x} = f(t, x)$ has a divergence-free vector-field (equivalently, that the flow is volume preserving) and its orbits remain bounded. This is then sufficient to conclude $\dot{x} = f(t, x)$ is Poincaré recurrent, since the discrete-time system forms a subsequence of the continuous-time system.

4.3 Gradient Descent-Ascent in Periodic Zero-Sum Bilinear Games

To begin, we focus on continuous-time Gradient Descent-Ascent (GDA) learning dynamics in periodic zero-sum bilinear games, which were introduced in Section 4.2.1. GDA dynamics are considered one of the standard learning dynamics for finding Nash equilibria in static two-player zero-sum games [11, 117]. These dynamics are such that each player seeks to maximize their utility by following the gradient with respect to their choice variable. Additionally, they arise as a special case of the FTRL and MD frameworks. Formally, gradient descent-ascent dynamics are given by:

$$\begin{aligned} \dot{x}_1 &= A(t)x_2(t) \\ \dot{x}_2 &= -A^\top(t)x_1(t). \end{aligned} \tag{GDA}$$

In the static zero-sum game regime, several key results describe the landscape of convergence and cycling of GDA dynamics. In particular, while the GDA dynamics are known to exhibit Poincaré recurrence in static zero-sum games, the time-average strategies in the long-run converge to the Nash equilibrium of the game. In this sense, these dynamics become a method of equilibrium computation as well. In this section, we seek to characterize both the recurrent and time-average behavior of GDA in periodic zero-sum bilinear games, mirroring the classical analysis.

4.3.1 Poincaré Recurrence

The focus of this section is on characterizing the transient behavior of the continuous-time GDA learning dynamics in periodic zero-sum bilinear games. Specifically, we show the following main result.

Theorem 4.3.1. *Continuous-time GDA learning dynamics are Poincaré recurrent in any periodic zero-sum bilinear game (Definition 4.2.1).*

Theorem 4.3.1 establishes that the recurrent nature of continuous-time GDA dynamics in static zero-sum bilinear games is robust to the dynamic evolution of the payoffs in periodic zero-sum bilinear games.

Prior to outlining the proof steps for Theorem 4.3.1, we elaborate on the claim from the previous section that without the periodicity property and a time-invariant equilibrium, such a result is unobtainable. In particular, we show the Poincaré recurrence of the GDA

dynamics is not guaranteed without both properties by constructing counterexamples when only one of the properties holds.

Proposition 4.3.1. *There exists time-evolving zero-sum bilinear games such that there is a time-invariant equilibrium or the payoffs are periodic (but not both simultaneously) in which the* GDA *dynamics are not Poincaré recurrent.*

The proof of this proposition is provided in Appendix B.1. This proposition highlights the strength of our results regarding GDA, given that the assumptions needed to obtain them are more or less tight.

We now outline the key intermediate results needed to obtain Theorem 4.3.1, following the techniques described in Section 4.2.3. For the GDA dynamics, we utilize the observation that the corresponding vector fields are divergence free to show that the learning dynamics are volume-preserving. We now state this result formally.

Lemma 4.3.1. *The* GDA *learning dynamics are volume preserving in any periodic zero-sum bilinear game as given in Definition 4.2.1.*

Proof. We show that the vector field which arises from GDA dynamics is divergence free and then apply Liouville's theorem. Indeed,

$$\operatorname{div}(\dot{x}) = \sum_{i=1}^{2} \sum_{j=1}^{n_i} \frac{\partial \dot{x}_{ij}}{\partial x_{ij}} = 0,$$

which follows from the fact that $\dot{x}_{ij}$ is independent of x_{ij} for each i, j. The divergence free property of the vector field then ensures that the flow ϕ^t of the differential equation is volume preserving by Liouville's theorem. □

We then proceed by showing that the GDA orbits are bounded by deriving a time-invariant function. This step relies on the fact that we have a time-invariant equilibrium.

Lemma 4.3.2. *The function* $\Phi(t) = \frac{1}{2}\big(x_1^\top(t)x_1(t) + x_2^\top(t)x_2(t)\big)$ *is time-invariant. Hence, the* GDA *orbits are bounded in any periodic zero-sum bilinear game as given in Definition 4.2.1.*

Proof. To prove this statement, we claim the following function is time-invariant:

$$\Phi(t) = \frac{1}{2}\big(x_1^\top(t)x_1(t) + x_2^\top(t)x_2(t)\big).$$

By taking the time-derivative of the $\Phi(t)$ we can verify the function is a constant of motion. Indeed:

$$\begin{aligned}
\frac{d\Phi}{dt} &= \frac{1}{2}\big(x_1^\top(t)\dot{x}_1 + \dot{x}_1^\top x_1(t) + x_2^\top(t)\dot{x}_2 + \dot{x}_2^\top x_2(t)\big) \\
&= x_1^\top(t)\dot{x}_1 + x_2^\top(t)\dot{x}_2 \\
&= x_1^\top(t)A(t)x_2(t) - x_2(t)^\top A^\top(t)x_1(t) \\
&= 0.
\end{aligned}$$

Finally, observe that given a bounded initial condition, the time-invariance of $\Phi(t)$ directly implies that no strategy of any player can become unbounded, so the flow ϕ^t of the differential equation has bounded orbits. □

Given the volume preservation and bounded orbit characteristics of the continuous-time GDA learning dynamics in periodic zero-sum games, the proof of recurrence follows by applying the arguments described in Section 4.2.3.

Proof of Theorem 4.3.1. Given the previous intermediate results, Theorem 4.3.1 follows from the arguments presented in Section 4.2.3. In particular, observe that by definition of the periodic zero-sum bilinear game, the GDA dynamics are T-periodic. Now, consider the discrete-time dynamical system defined by the Poincaré map ϕ^T that arises. This system retains the volume preservation property of the continuous-time system from Lemma 4.3.1 since as presented in Section 4.2.3, if a T-periodic system is divergence-free then the discrete-time system defined by ϕ^T is also volume preserving [5, 3.16.B, Thm 2]. Similarly, the discrete-time system defined by the ϕ^T retains the bounded orbits guarantee of the continuous-time system from Lemma 4.3.2 since it holds at any set of times. Thus, we are able to apply the Poincaré recurrence theorem for discrete-time systems from Section 4.2.3 to the discrete-time system defined by ϕ^T to conclude the discrete-time system is Poincaré recurrent. This immediately implies that the GDA dynamics are Poincaré recurrent since the discrete-time system defined by ϕ^T forms a subsequence of the continuous-time system. □

Matching Pennies Simulation. For continuous-time GDA dynamics, we show experimentally that Poincaré recurrence holds in a periodic zero-sum bilinear game. As an indicative example, we consider the ubiquitous Matching Pennies game with payoff matrix $A = \left[\begin{smallmatrix} 1 & -1 \\ -1 & 1 \end{smallmatrix}\right]$. We then use the following periodic rescaling with period 2π:

$$\alpha(t) = \begin{cases} \sin(t) & 0 \leq t \leq \frac{3\pi}{2} \\ \left(\frac{2}{\pi}\right)(t \bmod (2\pi) - 2\pi) & \frac{3\pi}{2} \leq t \leq 2\pi \end{cases} \tag{4.1}$$

Hence, the bilinear zero-sum game at time $t \geq 0$ is then described by the payoffs $\{\alpha(t)A(t), -\alpha(t)A(t)^\top\}$. When players use GDA dynamics, we see from Figure 4.1 that the players' trajectories when plotted alongside the value of the periodic rescaling are bounded.

4.3.2 Time-Average Convergence

The Poincaré recurrence of continuous-time GDA learning dynamics in periodic zero-sum bilinear games indicates that the system has regularities which couple the evolving players and evolving game despite the failure to converge to a fixed point. A natural follow-up question to the cyclic transient behavior of the dynamics is whether the long-run converges to a game-theoretically meaningful outcome.

Surprisingly, we show that in periodic zero-sum bilinear games, the time-average of GDA learning dynamics may not converge to the time-invariant Nash equilibrium.

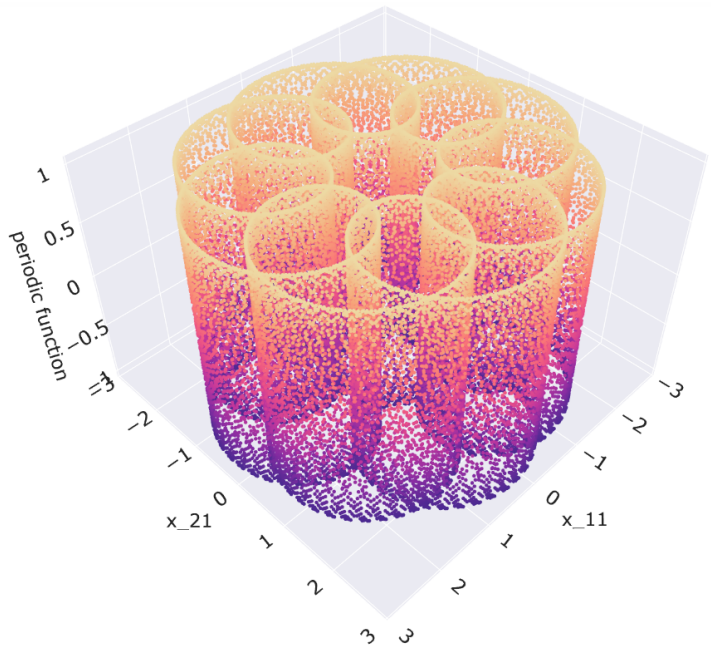


FIGURE 4.1: Bounded trajectories for a periodically rescaled Matching Pennies game updated using GDA. The color of the points denotes the value of the periodic rescaling function.

Proposition 4.3.2. *There exist periodic zero-sum bilinear games satisfying Definition 4.2.1 where the time-average strategies of the* GDA *dynamics fail to converge to the time-invariant equilibrium* $(\mathbf{0}, \mathbf{0})$.

Proof. Consider a periodic zero-sum bilinear game with $x_1, x_2 \in \mathbb{R}$ and a periodic payoff matrix $A(t)$ such that $A(t) = A(t+T)$ with period $T = 3\pi$ for any $t \geq 0$. Moreover, let the payoff matrix evolve over a period as follows:

$$A(t) = \begin{cases} -1 & 0 \leq t \leq \pi \\ 1 & \pi \leq t \leq \frac{3\pi}{2} \\ -1 & \frac{3\pi}{2} \leq t \leq 3\pi. \end{cases}$$

Clearly, the joint strategy $(x_1^*, x_2^*) = (\mathbf{0}, \mathbf{0})$ is the time-invariant Nash equilibrium. We now show that the time-average of the strategies produced by the GDA dynamics do not converge to the time-invariant Nash equilibrium.

The GDA dynamics in this periodic zero-sum bilinear game are given by

$$\begin{aligned} \dot{x}_1 &= A(t)x_2(t) \\ \dot{x}_2 &= -A(t)x_1(t). \end{aligned}$$

The solution to the differential equation that describes the GDA dynamics can be constructed in a piecewise manner. On each of the three intervals we have a linear system

defined by

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{bmatrix}0 & A(t)\\ -A(t) & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}.$$

Now recall the following identity

$$\exp\left(\theta\begin{bmatrix}0 & 1\\ -1 & 0\end{bmatrix}\right) = \begin{bmatrix}\cos(\theta) & \sin(\theta)\\ -\sin(\theta) & \cos(\theta)\end{bmatrix}.$$

Hence, for initial condition $(x_1(0), x_2(0))$ and interval $[0, \pi)$ we know that $A(t) = -1$ for all t in the interval which implies that the solution on this interval is given by:

$$\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix} = \begin{bmatrix}\cos(-t) & \sin(-t)\\ -\sin(-t) & \cos(-t)\end{bmatrix}\begin{bmatrix}x_1(0)\\ x_2(0)\end{bmatrix}.$$

On the interval $[\pi, 3\pi/2)$, $A(t) = 1$, so:

$$\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix} = \begin{bmatrix}\cos(t-\pi) & \sin(t-\pi)\\ -\sin(t-\pi) & \cos(t-\pi)\end{bmatrix}\begin{bmatrix}x_1(\pi)\\ x_2(\pi)\end{bmatrix}.$$

Finally, on $[3\pi/2, 3\pi)$, $A(t) = -1$, so:

$$\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix} = \begin{bmatrix}\cos(-(t-3\pi/2)) & \sin(-(t-3\pi/2))\\ -\sin(-(t-3\pi/2)) & \cos(-(t-3\pi/2))\end{bmatrix}\begin{bmatrix}x_1(3\pi/2)\\ x_2(3\pi/2)\end{bmatrix}.$$

Now, let us consider the initial condition $(x_1(0), x_2(0)) = (1, 0)$. With this initial condition, the solution on the interval $t \in [0, \pi)$ is:

$$(x_1(t), x_2(t)) = (\cos(t), \sin(t)) = (\cos(-t), -\sin(-t))$$

The solution on the interval $t \in [\pi, 3\pi/2)$ is:

$$(x_1(t), x_2(t)) = (\cos(t), -\sin(t)) = (-\cos(t-\pi), \sin(t-\pi))$$

And finally the solution on the interval $t \in [3\pi/2, 3\pi)$ is:

$$(x_1(t), x_2(t)) = (-\cos(t), -\sin(t)) = (\sin(-(t-3\pi/2)), \cos(-(t-3\pi/2)))$$

Observe that from this solution, the GDA dynamics return to the initial condition $(1, 0)$ at the end of a period. Thus, to assess convergence of the time-average it is sufficient to evaluate the time-average of the dynamics over a period of the evolving game. Integrating the solution over a period, we have that

$$\begin{aligned}\int_0^{3\pi} x_1(t)dt &= \int_0^{3\pi/2}\cos(t)dt + \int_{3\pi/2}^{3\pi} -\cos(t)dt\\ &= [\sin(3\pi/2) - \sin(0)] - [\sin(3\pi) - \sin(3\pi/2)]\\ &= -2\end{aligned}$$

and

$$\begin{aligned}\int_0^{3\pi} x_2(t)dt &= \int_0^{\pi} \sin(t)dt + \int_{\pi}^{3\pi} -\sin(t)dt \\ &= [-\cos(\pi) + \cos(0)] + [\cos(3\pi) - \cos(\pi)] \\ &= 2\end{aligned}$$

This implies that the time-average strategies of the players do not equal to zero, so the time-average of the GDA dynamics do not converge to the time-invariant Nash equilibrium. The convergence to non-zero still holds even with different initial conditions.

This completes the proof and shows that there exists periodic zero-sum bilinear games where the time-average GDA strategies do not converge to the time-invariant Nash equilibrium. □

Given the simplicity of this example, it effectively rules out hope to provide a meaningful time-average convergence guarantee in this class of games.

Matching Pennies Simulation. We show experimentally that even a simple periodic rescaling of the Matching Pennies game fails to converge to the time-invariant equilibrium. This simple example can be constructed by considering a periodic zero-sum bilinear game with $x_1, x_2 \in \mathbb{R}$ and a periodic rescaling $\beta(t)$ such that $\beta(t) = \beta(t+T)$ with $T = 3\pi$ for any $t \geq 0$. Moreover, let the rescaling evolve over a period as follows:

$$\beta(t) = \begin{cases} -1 & 0 \leq t \leq \pi \\ 1 & \pi \leq t \leq \frac{3\pi}{2} \\ -1 & \frac{3\pi}{2} \leq t \leq 3\pi. \end{cases} \tag{4.2}$$

For the simulation, we consider the payoff matrices of a periodically rescaled Matching Pennies game described by $\{\beta(t)A, -\beta(t)A^\top\}$ where $A = \left[\begin{smallmatrix} 1 & -1 \\ -1 & 1 \end{smallmatrix}\right]$. First, note that the equilibrium of the game is always $[1/2, 1/2]$ for either player at any point in a period. In Figure 4.2, we show for this example that when both players use GDA, the time average strategy of each player remains bounded away from the Nash $[1/2, 1/2]$, even though the time-average utilities of the players go to zero.

4.4 Follow-the-Regularized-Leader in Periodic Zero-Sum Polymatrix Games

We now turn to the games with more players, and a network polymatrix structure similar to the one analyzed in Chapter 3. In this section, we analyze continuous-time FTRL learning dynamics in periodic zero-sum polymatrix games. As mentioned before, FTRL dynamics represent a generalization of the replicator dynamics studied in Chapter 3. In words, players that follow FTRL learning dynamics in this class of games select a mixed strategy at each time that maximizes the difference between the cumulative payoff evaluated over the history of games and a regularization penalty. Intuitively, this adaptive strategy balances exploitation based on the past payoffs with exploration of new strategies.

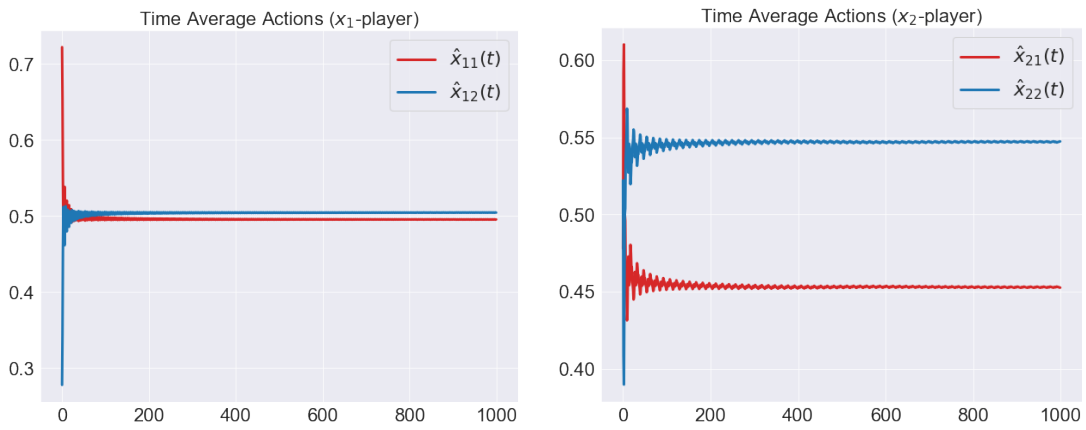


FIGURE 4.2: Time average results for MP rescaled with $\beta(t)$ function. Notice that the time average actions/strategies of the player 1 (left) and player 2 (right) both do not converge to the time-invariant Nash [1/2, 1/2].

Building off of the definition of FTRL from Chapter 2.2, the continuous-time FTRL learning dynamics for any player $i \in V$ in a periodic zero-sum polymatrix game with an initial payoff vector $y_i(0) \in \mathbb{R}^{n_i}$ are given by

$$\begin{aligned} y_i(t) &= y_i(0) + \int_0^t \sum_{j:(i,j)\in E} A^{ij}(\tau) x_j(\tau) d\tau \\ x_i(t) &= \operatorname{argmax}_{x_i \in \mathcal{X}_i} \{\langle x_i, y_i(t)\rangle - h_i(x_i)\} \end{aligned} \tag{FTRL}$$

where $h_i : \mathcal{X}_i \to \mathbb{R}$ is a regularization term which encourages exploration away from the strategy which maximizes the cumulative payoffs in hindsight. We assume that the regularization function $h_i(\cdot)$ for each player $i \in V$ is continuous, strictly convex on $\mathcal{X}_i$, and smooth on the relative interior of every face of $\mathcal{X}_i$. These assumptions ensure the update $x_i(t)$ is well-defined since a unique solution exists.

By carefully selecting the regularizer h_i, one can derive various common learning dynamics in the literature. For instance, the replicator dynamics for a player $i \in V$ arise from the entropic regularization function $h_i(x_i) = \sum_{\alpha \in \mathcal{S}_i} x_{i\alpha} \log x_{i\alpha}$. Moreover, the projected gradient descent dynamics (effectively, gradient descent dynamics subject to a constraint) for a player $i \in V$ are derived from using Euclidean regularizer $h_i(x_i) = \frac{1}{2}\|x_i\|_2^2$.

To simplify notation, the FTRL dynamics above can equivalently be formulated as the following update:

$$\begin{aligned} y_i(t) &= y(0) + \int_0^t v_i(x(\tau), \tau) d\tau \\ x_i(t) &= Q_i(y_i(t)). \end{aligned} \tag{4.3}$$

Here, we utilize the definition of a joint strategy profile $x = (\alpha, x_{-i}) \in \mathcal{X}$ at time $\tau \geq 0$, which means that player i plays with pure strategy $\alpha \in \mathcal{S}_i$ while all other players play according to strategy profile x. Then, the payoff of player i over all their edge games

can be encapsulated using $v_i(x(\tau), \tau) = (u_{i\alpha}(x(\tau), \tau))_{\alpha \in \mathcal{S}_i}$.

Moreover, $Q_i : \mathbb{R}^{n_i} \to \mathcal{X}_i$ is known as the choice map and is defined as

$$Q_i(y_i(t)) = \operatorname{argmax}_{x_i \in \mathcal{X}_i} \{\langle y_i(t), x_i \rangle - h_i(x_i)\}.$$

With this notation, the utility of the player $i \in V$ under the joint strategy $x = (x_i, x_{-i}) \in \mathcal{X}$ at time $t \geq 0$ is given by $u_i(x, \tau) = \langle v_i(x, \tau), x_i \rangle$. Observe that in our notation of utility, we now include the time index to make the dependence on the evolving game and corresponding payoffs explicitly clear.

Finally, for any player $i \in V$ we denote by $h_i^* : \mathbb{R}^{n_i} \to \mathbb{R}$ the convex conjugate of the regularization function $h_i : \mathcal{X}_i \to \mathbb{R}$ which is given by the quantity $h_i^*(y_i(t)) = \max_{x_i \in \mathcal{X}_i} \{\langle x_i, y_i(t) \rangle - h_i(x_i)\}$.

It is known that the continuous-time `FTRL` learning dynamics are Poincaré recurrent in static zero-sum polymatrix games [117]. However, [32] show that time-average convergence to the Nash equilibrium *fails* in static zero-sum polymatrix games. Much like the case of periodic zero-sum bilinear games, we now investigate these results in our time-evolving setting.

4.4.1 Poincaré Recurrence

We now focus on characterizing the transient behavior of the continuous-time `FTRL` learning dynamics in periodic zero-sum polymatrix games. The following result demonstrates that Poincaré recurrence holds even in games that are evolving in a periodic fashion with a time-invariant equilibrium, providing a broad generalization of known results.

Theorem 4.4.1. *The* `FTRL` *learning dynamics are Poincaré recurrent in any periodic zero-sum polymatrix game as given in Definition 4.2.2.*

For the remainder of this subsection, we describe our proof methods. The general approach is that we prove the Poincaré recurrence of a transformed system using the techniques described in Section 4.2.3. This conclusion then allows us to derive recurrence for the original `FTRL` system.

The utility differences for each player $i \in V$ and pure strategy $\alpha_i \in \mathcal{S}_i \setminus \beta_i$ evolve following the differential equation

$$\dot{z}_{i\alpha_i} = v_{i\alpha_i}(x(t), t) - v_{i\beta_i}(x(t), t). \tag{4.4}$$

Toward proving that this system is Poincaré recurrent, we show that the vector field $\dot{z}$ is divergence free and hence volume preserving.

Lemma 4.4.1. *The dynamics defined by the system $\dot{z}$ are volume preserving in any periodic zero-sum polymatrix game as given in Definition 4.2.2.*

Proof. We first describe how the $\dot{z}$ dynamics are formulated. Then, we show that the divergence of this vector field is zero, from which we conclude the dynamics are volume preserving by Liouville's theorem. This proof closely follows arguments in [117].

For each player $i \in V$, given a fixed strategy $\beta \in \mathcal{S}_i$, for all $\alpha \in \mathcal{S}_i \setminus \beta$ the cumulative utility differences are defined by

$$z_{i\alpha}(t) = y_{i\alpha}(t) - y_{i\beta}(t).$$

This transformation from the cumulative utilities to the cumulative utility differences yields a linear map $\Pi_i : \mathbb{R}^{n_i} \to \mathbb{R}^{n_i - 1}$ from $y_i(t)$ to $z_i(t)$ for each player $i \in V$. Moreover, $\Pi = (\Pi_1, \dots, \Pi_{|V|})$ defines the product map of the linear maps Π_i of each player $i \in V$. Note that this map is surjective, but not injective.

Observe that the cumulative utility differences for each player $i \in V$ and all $\alpha \in \mathcal{S}_i \setminus \beta$ evolve following the differential equation

$$\dot{z}_{i\alpha}(t) = v_{i\alpha}(x(t), t) - v_{i\beta}(x(t), t)$$

due to the definition of y_i and the fundamental theorem of calculus. Moreover, recall that for any player $i \in V$ and pure strategy $\gamma \in \mathcal{S}_i$, the quantity $v_{i\gamma}(x(t), t)$ denotes the utility of player $i \in V$ at any time $t \geq 0$ for selecting the pure strategy $\gamma \in \mathcal{S}_i$.

To analyze the dynamics from the system in Equation (4.4) we need it to be well-defined, which is not immediately obvious since it depends on $x(t) = Q(y(t))$. Moreover, the mapping from $y(t)$ to $z(t)$ via Π is not invertible, so $y(t)$ cannot be expressed as a function of $z(t)$. Despite this, the system is in fact well-defined. To see why this is the case, for each player $i \in V$, consider the reduced choice map $\hat{Q}_i : \mathbb{R}^{n_i - 1} \to \mathcal{X}_i$ defined as $\hat{Q}_i(z_i(t)) = Q_i(y_i(t))$ for some $y_i(t) \in \mathbb{R}^{n_i}$ such that $\Pi_i(y_i(t)) = z_i(t)$. Note that $\Pi_i(y_i(t))$ is guaranteed to exist since Π_i is surjective. Then, the fact that $\hat{Q}_i(z_i(t))$ is well-defined for each player $i \in V$ holds since by the construction, $\Pi_i(y_i(t)) = \Pi_i(y_i'(t))$ if and only if $y_{i\alpha}'(t) = y_{i\alpha}(t) + c$ for $c \in \mathbb{R}$ and every $\alpha_i \in \mathcal{S}_i$. This immediately implies that $Q_i(y_i'(t)) = Q_i(y_i(t))$ if and only if $\Pi_i(y_i(t)) = \Pi_i(y_i'(t))$. Finally, let $\hat{Q} = (\hat{Q}_1, \dots, \hat{Q}_{|V|})$ be the combined reduced choice map and note that $Q(y(t)) = \hat{Q}(\Pi(y(t)) = \hat{Q}(z(t))$ by construction. As a result, the dynamics from the system in Equation (4.4) are equivalently given by the following system

$$\dot{z}_{i\alpha} = v_{i\alpha_i}(\hat{Q}_i(z(t), t)) - v_{i\beta_i}(\hat{Q}_i(z(t)), t).$$

This system is well-defined by the arguments above, which then ensures that the system in Equation (4.4) is well-defined.

Now that we have shown the system is well-defined, we prove that it is volume preserving. To see this, observe that the vector field is divergence free. Indeed,

$$\operatorname{div}(\dot{z}) = \sum_{i \in V} \sum_{\alpha \in \mathcal{S}_i} \frac{\partial \dot{z}_{i\alpha}}{\partial z_{i\alpha}} = \sum_{i \in V} \sum_{\alpha \in \mathcal{S}_i} \sum_{\gamma_i \in \mathcal{S}_i} \frac{\partial \dot{z}_{i\alpha}}{\partial x_{i\gamma}} \frac{\partial x_{i\gamma}}{\partial z_{i\alpha_i}} = 0.$$

Note that the equation above holds since for each player $i \in V$, the pure strategy utilities at any time $t \geq 0$ given by $v_i(x(t), t)$ where $v_{i\alpha}(x(t), t) = u_i((\alpha, x_{-i}(t)), t)$ do not depend on $x_i(t)$. Finally, the divergence free property of the vector field ensures that the flow ϕ^t of the differential equation is volume preserving by Liouville's theorem. □

Following the standard proof technique of Poincaré recurrence, we next construct a time-invariant function of the evolution of the system. This suffices to guarantee that the orbits generated by the $\dot{z}$ dynamics are bounded. Recall that x^* denotes the common, time-invariant interior Nash equilibrium of the periodic zero-sum polymatrix game.

Lemma 4.4.2. *The function* $\Phi(x^*, y(t)) = \sum_{i \in V} \left(h_i^*(y_i(t)) - \langle x_i^*, y_i(t) \rangle + h_i(x_i^*) \right)$ *is time-invariant. Hence, the orbits generated by the* $\dot{z}$ *dynamics are bounded in any periodic zero-sum polymatrix game as given in Definition 4.2.2.*

We now provide a sketch of the proof of Lemma 4.4.2. In order to show that the function $\Phi(x^*, y(t)) = \sum_{i \in V} \left(h_i^*(y_i(t)) - \langle x_i^*, y_i(t) \rangle + h_i(x_i^*) \right)$ is time-invariant, we show that the time derivative of the function is zero. In particular, $\frac{d\Phi(x^*, y(t))}{dt}$ can be written as:

$$\frac{d\Phi(x^*, y(t))}{dt} = \frac{d}{dt} \sum_{i \in V} h_i^*(y_i(t)) + \frac{d}{dt} \sum_{i \in V} \langle x_i^*, y_i(t) \rangle.$$

We show both terms in the RHS of the equation are zero, which then directly implies that $\Phi(x^*, y(t))$ is time-invariant. The full proof of this step is more involved, and we present it along with the full proof and reasoning in Appendix B.2.

From this point, we follow the arguments from Section 4.2.3 to conclude the $\dot{z}$ dynamics are Poincaré recurrent. Finally, we show that Poincaré recurrence of the $\dot{z}$ system is sufficient to guarantee the Poincaré recurrence of the `FTRL` learning dynamics.

Proof of Theorem 4.4.1. Theorem 4.4.1 states that the `FTRL` dynamics are Poincaré recurrent in periodic zero-sum polymatrix games. The proof of this claim follows from Lemma 4.4.1, Lemma 4.4.2 and the methods described in Section 4.2.3. Indeed, to begin, observe that the $\dot{z}$ dynamics given in (4.4) are T-periodic. This follows immediately from the definition of a T-periodic system as described in Section 4.2.3 and the fact that the payoff matrices are T-periodic. Consider the discrete-time dynamical system defined by the Poincaré map ϕ^T that arises. This system retains the volume preservation property of the continuous-time system from Lemma 4.4.1 since as presented in Section 4.2.3, if a T-periodic system is divergence-free then the discrete-time system defined by ϕ^T is also volume preserving [5, 3.16.B, Thm 2]. Furthermore, the bounded orbits guarantee of the continuous-time system from Lemma 4.4.2 imply the discrete-time system defined by ϕ^T has bounded orbits since it is a subsequence of the continuous-time system.

Thus, we are able to apply the Poincaré recurrence theorem to the system defined by ϕ^T to conclude that the discrete-time system is Poincaré recurrent. This implies that the $\dot{z}$ dynamics are Poincaré recurrent since the discrete-time system defined by ϕ^T forms a subsequence of the continuous-time system. Finally, the Poincaré recurrence of the $\dot{z}$ dynamics directly imply the Poincaré recurrence of the `FTRL` strategies. Indeed, since there is an increasing sequence of times t_n such that $z(t_n) \to z(0)$ by Poincaré recurrence, so using continuity there is also an increasing sequence of times t_n such that $x(t_n) = Q(y(t_n)) = \hat{Q}(z(t_n)) \to \hat{Q}(z(t_0)) = x(0)$ which means the `FTRL` dynamics are Poincaré recurrent. □

Matching Pennies Simulations. For the case of FTRL dynamics, we perform simulations on a periodically rescaled Matching Pennies game updated with replicator dynamics (since it is a special case of FTRL). Similarly to the case of GDA, we utilize the periodic rescaling described in Equation 4.1 and obtain recurrent dynamics, as seen in Figure 4.3. Note that in this section, we present simulations on two-player Matching Pennies games, which are trivially also zero-sum polymatrix games with two nodes and a single edge between them.

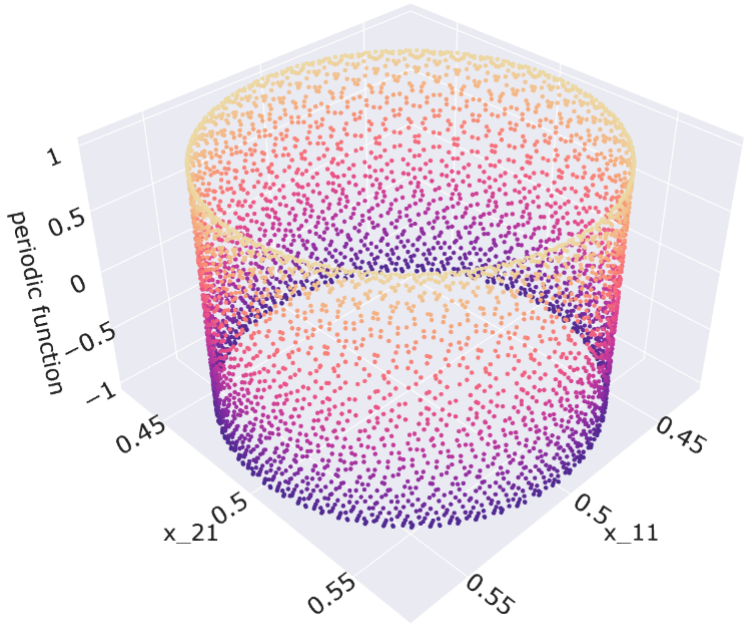


FIGURE 4.3: Bounded trajectories for a periodically rescaled Matching Pennies game updated using FTRL (replicator dynamics). The color of the points denotes the value of the periodic rescaling function.

Moreover, the time-invariant function presented in Lemma 4.4.1 and its proof can be demonstrated in any periodic zero-sum polymatrix game. For replicator dynamics, the invariant function is precisely the KL-divergence between each player's strategy and the unique mixed Nash. Using the same simulated data that was used to generate Figure 4.3 (i.e. in the two-player case), we show that the sum of KL-divergences is indeed constant when both players are using replicator dynamics (Figure 4.4). Specifically, the blue area represents the KL-divergence of the first player from the mixed Nash over time, and the green area represents the divergence of the second player.

4.4.2 Time-Average Convergence

A number of well-known properties of zero-sum bimatrix games fail to generalize to zero-sum polymatrix games. Indeed, fundamental characteristics of zero-sum bimatrix games include that each player has a unique utility value in every Nash equilibrium, and that equilibrium strategies are interchangeable. However, [32] show that neither of these properties are guaranteed in zero-sum polymatrix games. Consequently, in the setting of polymatrix games, time-average convergence to the set of equilibrium values

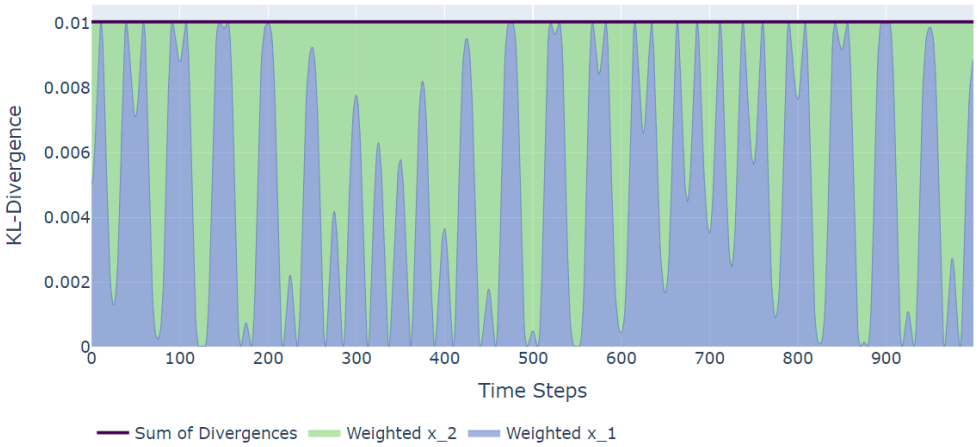


FIGURE 4.4: Time invariant function for two player periodically rescaled Matching Pennies game

in the utility and strategy spaces does not equate to the stronger notion of pointwise convergence.

For the reasons just outlined, we pursue a different notion of time-average convergence in periodic zero-sum polymatrix games. Specifically, we consider the subclass of periodic zero-sum bimatrix games (2-player periodic zero-sum polymatrix games) and show that the time-average utility of each player converges to the time-average of the game values (that is, the unique utility the player obtains at any Nash equilibrium) over a period of the game.

Theorem 4.4.2. *In periodic zero-sum bimatrix games satisfying Definition 4.2.2, if each player follows* `FTRL` *dynamics, then the time-average utility of each player converges to the time-average over a period of the game equilibrium utility values.*

Theorem 4.4.2 paints a positive view of the time-average behavior of `FTRL` learning dynamics in periodic zero-sum games, and we provide its proof in Appendix B.3. Unfortunately, the following result demonstrates that much like in the case of `GDA` in periodic zero-sum bilinear games, the time-average strategies are not guaranteed to converge to the time-invariant Nash equilibrium.

Proposition 4.4.1. *There exist periodic zero-sum bimatrix games satisfying Definition 4.2.1 in which the time-average strategies of* `FTRL` *dynamics fail to converge to the time-invariant Nash equilibrium.*

We prove the negative result in this proposition by constructing a counterexample that corresponds to a time-varying rescaling of Matching Pennies (full proof in Appendix B.4).

Matching Pennies Simulations. Theorem 4.4.2 states that the time-average utility of each player converges to the time-average value of periodic zero-sum bimatrix games when each player follows `FTRL` dynamics. However, Proposition 4.4.1 states that there exist periodic zero-sum bimatrix games where the time-average strategies of `FTRL`

dynamics fail to converge to the time-invariant Nash equilibrium. Here, we show a simple example that exhibits both results. Consider a Matching Pennies game that is rescaled with a $\sin$ function. Specifically, the periodic bimatrix game is given by:

$$A(t) = \sin(t) \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \tag{4.5}$$

Even with this simple instance, the time-average utilities for both players go to zero (the equilibrium value), while the time-average strategies do not converge to the $[1/2, 1/2]$ time-invariant Nash, as seen in Figure 4.5.

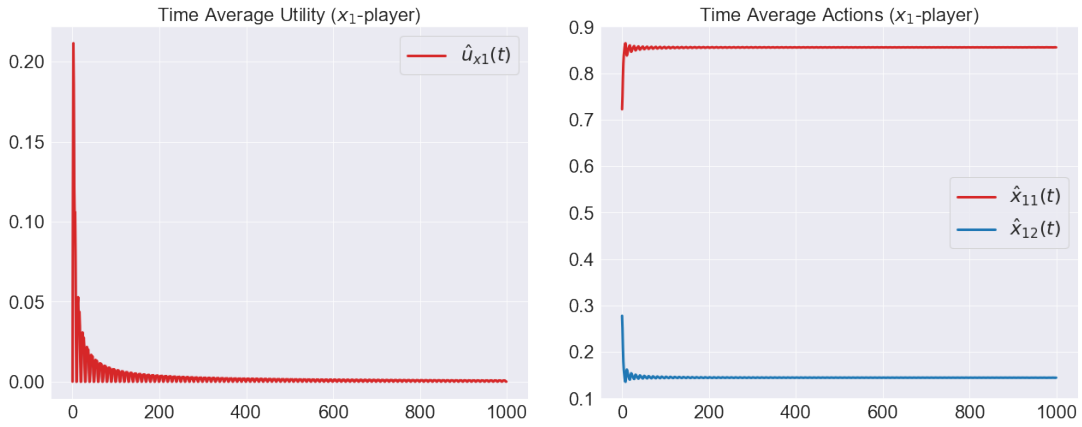


FIGURE 4.5: Time average results for MP rescaled with sin function. Notice that although the average utility (left) goes to 0, the time average actions/strategies of the player (right) does not go to the time invariant Nash $[1/2, 1/2]$.

4.5 Additional FTRL Simulations

In this section, we present several additional simulations, with a particular focus on zero-sum polymatrix games. In our polymatrix simulations, a number of players are arranged in a line. Each player then plays a bimatrix game against the player directly adjacent to them, and the final player also plays against the first player. This results in a 'toroid'-like chain of games, where each player plays against two other players. Finally, each pair of players plays the Matching Pennies game rescaled with a $\sin$ function against each other (Equation 4.5), and additionally each $\sin$ function is phase-shifted by a random amount. This models the scenario wherein the non-autonomous periodic evolution of the system differs across the players in the game, while still maintaining a broader pattern over all players.

KL-Divergence Constant of Motion. To show the claim of Lemma 4.4.2 about the existence of a constant of motion for games with more than two players, we ran a 64-player simulation using replicator dynamics (RD) and computed the weighted constant of motion (which in this case is the KL-divergence) at each timestep. Figure 4.6 depicts

the claim presented in the lemmas: although each player's specific divergence term $\mathrm{KL}(x_i^* \| x_i(t))$ fluctuates, the sum $\sum_{i \in V} \mathrm{KL}(x_i^* \| x_i(t))$ remains constant.

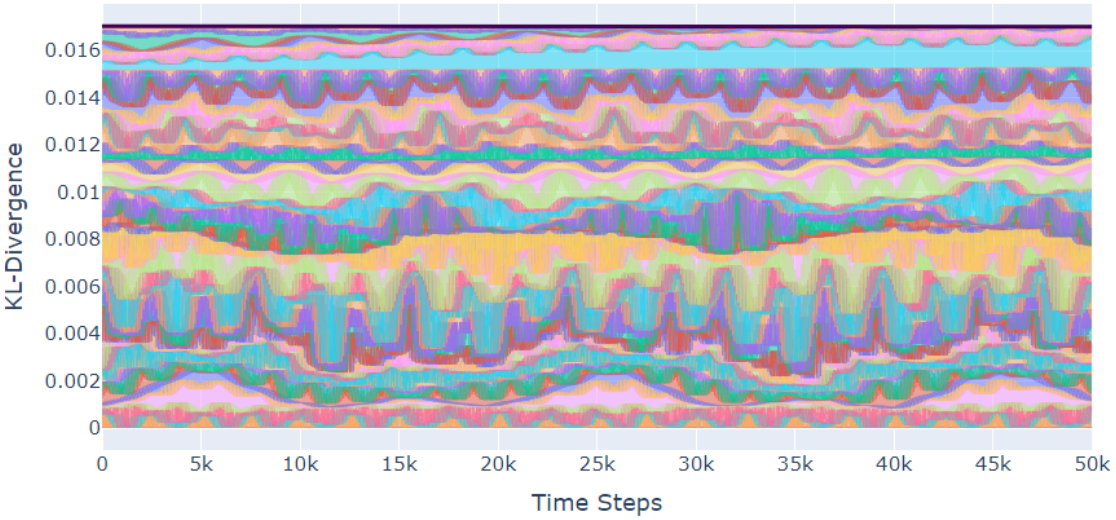


FIGURE 4.6: Weighted sum of KL-divergences for a 64-player periodically rescaled Matching Pennies game using replicator dynamics. Note that despite the complicated trajectories of each player, the weighted sum of their KL-divergences remains constant.

In Figure 4.7 we show zoomed-in plots of Figure 4.6. Here, similar to previous plots, each player's KL-divergence is represented by a different-colored area. The figure showcases that on a more granular level, the individual KL-divergences of each agent can become extremely erratic, and look nowhere near periodic. Nevertheless, we see from Figure 4.6 that the weighted sum of KL-divergences remains constant.

Clyde Image Simulation. Similarly to Chapter 3, we also represent the trajectories of each player by equating the strategy values of each player to RGB values in an 8x8 image. In particular, the color of each pixel on an image of Clyde from Pac-Man represents the probability of the respective player playing the first strategy, tuned with a sigmoid function. We then select initial conditions that correspond to RGB values such that they form the image shown in Figure 4.8.

With the sigmoid function, any changes from the initial condition are reflected by changes in the color of the individual pixels. Thus, as players play their pairwise bimatrix games using replicator dynamics, the colors of the grid evolve. If the system exhibits Poincaré recurrence, we should eventually see similar patterns emerge as the pixels change color over time (i.e., as their corresponding mixed strategies evolve). As observed in Figure 4.9, for the case of the Matching Pennies games rescaled with $\sin$ (without phase-shifts), the system returns near the initial image at time $T = 6226$.

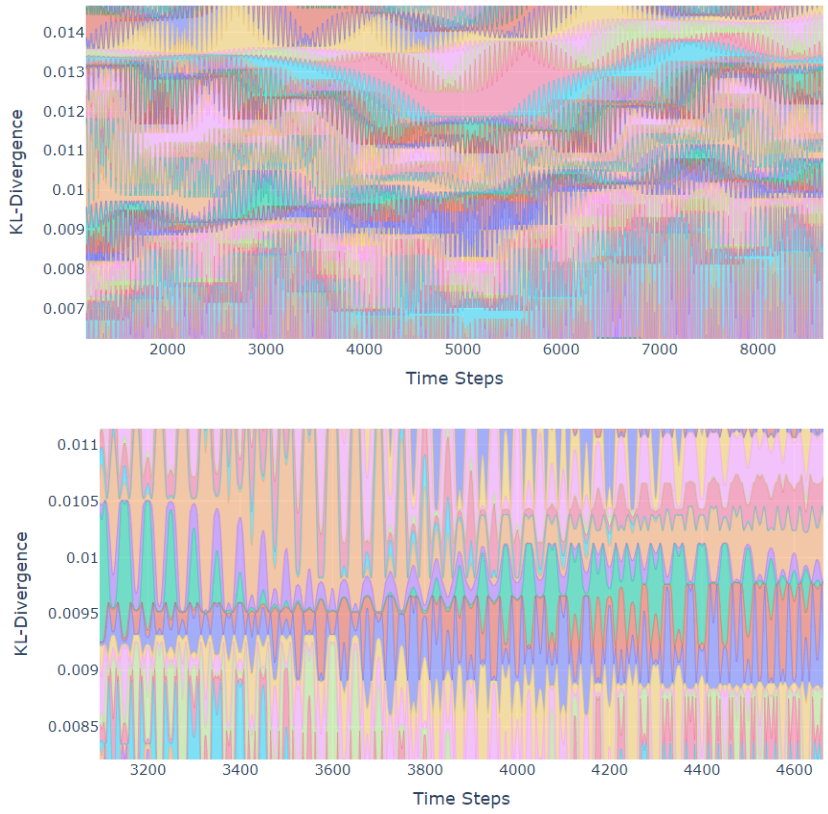


FIGURE 4.7: Zoomed-in time invariant functions for 64-player game.

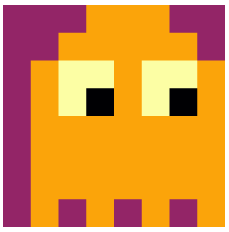
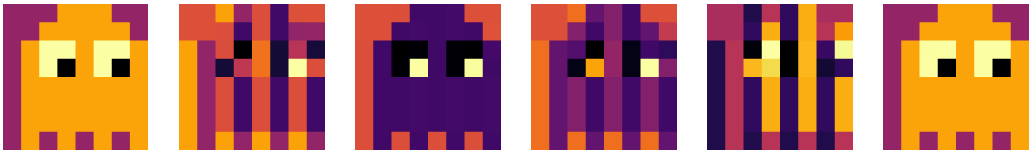


FIGURE 4.8: 8×8 grid of colors generated by sigmoid function

4.6 Conclusion

We study both GDA and FTRL learning dynamics in periodically evolving zero-sum games. We prove that the recurrent nature of such dynamics carries over from static games to the classes of evolving games we study. Yet, in the settings we analyze, the time-average convergence behavior from static zero-sum games can fail to generalize. This work takes a step toward understanding the behavior of classical learning algorithms for games in the more realistic setting where the game itself is not fixed.

However, there are several other models of exogeneous time-evolution which have been studied. In a class of time-evolving games where the evolution can be arbitrary [33], algorithms have been designed to provide a novel type of regret guarantee called Nash equilibrium regret. Intuitively, these algorithms are designed to be competitive



(a) t=0 (b) t=700 (c) t=1500 (d) t=3000t (e) t=5000 (f) t=6226

FIGURE 4.9: Sequence of images showing Poincaré recurrence in an 8×8 zero-sum polymatrix game, where the changing color of each pixel on the grid represents the mixed strategy of the player over time. After time $T = 6226$, we see that an approximation of the original image is recovered, showing that the recurrence property holds.

against the Nash equilibrium of the long-term-averaged payoff matrix. In an analysis of discrete-time FTRL dynamics in evolving games that are strictly/strongly monotone [53], sufficient conditions under which the dynamics converge to the evolving equilibrium are derived. While these settings are distinct from ours, this growing literature indicates that exogenously time-evolving games can oftentimes exhibit more complex behavior than their classic static counterparts. As such, there is much potential for future work towards a better understanding of online learning dynamics in such games.

Chapter 5

Matrix Replicator Dynamics in Quantum Zero-Sum Games

This chapter is taken from (with minor modifications) our paper *'Matrix Multiplicative Weights Updates in Quantum Zero-Sum Games: Conservation Laws & Recurrence'* [89].

5.1 Introduction

In this chapter, we move on from time-evolving games and instead focus on studying continuous-time dynamics in a different class of games with rich connections to quantum theory and semidefinite optimization. In order to do so, we study a discrete-time matrix generalization of MWU known as *Matrix Multiplicative Weights Update* (MMWU), which changes the paradigm of standard MWU in the following manner: Instead of selecting probability vectors as mixed strategies, players using MMWU now select density matrices (Hermitian positive semidefinite matrices with trace 1). Moreover, their payoffs are obtained via the measurement of a payoff observable instead of a payoff matrix. This generalization has been independently discovered and studied extensively by [170] as Matrix Exponentiated Gradient Updates and [6] as the Matrix Multiplicative Weights algorithm. Applications of the MMWU method include solving semi-definite programs (SDPs) [7] and obtaining bounds on the sample complexity for learning problems in quantum computing [91]. From a game-theoretic standpoint, the MMWU method applies to online learning in a generalization of classical zero-sum games called *quantum zero-sum games*. [90] analyzed MMWU in these games, and proved time-average convergence to approximate Nash equilibria.

Our approach in this chapter studies the MMWU method in a game-theoretic setting, with a focus on establishing dynamical properties. In particular, recall that classical MWU is one of the most ubiquitous discrete-time online learning dynamics in the literature, and that replicator dynamics (RD) is its continuous-time counterpart. This connection between discrete- and continuous-time dynamics is a rich one for a major reason: continuous-time dynamics can be framed as high-level 'meta'-dynamics which can then be discretized in various ways to obtain discrete-time dynamics with certain desirable properties. For example, this type of approach has led to a unified continuous-time perspective of accelerated methods in gradient descent [181]. Moreover, in classical congestion games,

a different discretization of RD known as *linear*-MWU has desirable convergence properties to equilibria, while standard MWU is shown to exhibit formally chaotic behavior [134]. As such, it is instructive to study continuous-time dynamics as a meaningful way to broaden the analysis of their discrete-time counterparts.

In line with this perspective, we formulate a generalization of the classical replicator dynamics we call Matrix Replicator Dynamics (MRD), so that it can be discretized as MMWU. Using this approach, we are able to derive dynamical properties (such as Poincaré recurrence) of systems of players using MRD in quantum zero-sum games, while simultaneously introducing the continuous-time framework of learning dynamics in games beyond the classical normal-form regime.

As described in previous chapters, there have been several important results studying classical MWU and classical RD. [13] showed that classical MWU does not converge in day-to-day behaviour to the mixed Nash equilibrium when applied to two player zero-sum games, even though the time-average values converge to the Nash. [117] showed that the orbits of players in classical zero-sum games exhibit Poincaré recurrence when using replicator dynamics. The time-average convergence of MMWU to Nash equilibria in quantum zero-sum games has already been shown in [90]. However, the continuous-time behavior of the matrix/quantum version of replicator dynamics (or indeed, any matrix online learning dynamic) is not well understood.

5.1.1 Our Contributions

In this chapter, we first define quantum zero-sum games, a non-commutative extension of classical normal-form games. Moreover, we establish preliminaries on quantum information theory which are necessary in our analysis. Utilizing these tools, we study the dynamics of discrete-time MMWU in quantum zero-sum games, utilizing tools from information theory and classical game theory. We provide bounds on the rate of change of the total *quantum relative entropy* in the system.

Following that, we formulate and study the continuous counterpart of MMWU, which we call matrix replicator dynamics (MRD). Analyzing this continuous dynamic allows for various discretizations in order to obtain different discrete-time algorithms for specific purposes. Furthermore, the dynamics exhibited by both players' trajectories do not converge to equilibrium (in the last-iterate sense) nor oscillate periodically, but rather exhibit *Poincaré recurrence*. In our proof of recurrence, we show that the total quantum relative entropy (rather than the KL-divergence) in the system is a constant of motion. This represents the first foray into the dynamical properties of continuous-time learning dynaimcs in quantum games. Although the high-level analysis mirrors the classical setting, it is worth noting that our proof of recurrence requires novel ideas since we are proving the recurrence result in a general compact convex space instead of beyond the simplex case. Finally, we present several simulations which corroborate our theoretical results.

5.2 Preliminaries and Definitions

In this chapter we study online learning in quantum games on the intersection between discrete-time MMWU and continuous-time MRD. As such, in this section we will first introduce the necessary definitions for a framework of quantum games. Moreover, we will utilize quantum information theory and dynamical systems theory to prove recurrence for MRD, so the necessary concepts are described in this section.

5.2.1 Quantum Theory

Our focus in this chapter is not quantum theory per se, but some basic definitions are needed in order to describe the formulation of quantum games we study. In the games we study, 'pure actions' correspond to d-dimensional quantum registers, which are also commonly referred to as qudits. A d-dimensional quantum register is mathematically described as the set of unit vectors in a d-dimensional Hilbert space $\mathcal{H}$.

The *state* of a qudit quantum register $\mathcal{H}$ is represented by a *density matrix*, i.e., a $d \times d$ Hermitian positive semidefinite (PSD) matrix with trace equal to 1. The state space of a quantum register $\mathcal{H}$ is denoted by $D(\mathcal{H})$. When two quantum registers with associated spaces $\mathcal{A}$ and $\mathcal{B}$ of dimension n and m respectively are considered as a joint quantum register, the associated state space is given by density operators on the tensor product space, i.e., $D(\mathcal{A} \otimes \mathcal{B})$. If the two registers are independently prepared in states described by ρ and σ, then the joint state is described by the density matrix $\rho \otimes \sigma \in \mathbb{C}^{nm \times nm}$.

Measurement is a fundamental aspect of quantum theory. Indeed, in order to interact with a quantum register we need to measure it, which intuitively is a process which assigns some probabilty to each of the possible outcomes of the system. The mathematical formalism for measuring a quantum system we will focus on is called the positive operator-valued measurement (POVM), defined as a set of positive semidefinite operators $\{P_i\}_{i=1}^m$ such that $\sum_{i=1}^m P_i = \mathbb{1}_{\mathcal{H}}$, where $\mathbb{1}_{\mathcal{H}}$ is the identity matrix on $\mathcal{H}$. If the quantum register $\mathcal{H}$ is in a state described by density matrix $\rho \in D(\mathcal{H})$, upon performing the measurement $\{P_i\}_{i=1}^m$ we will get the outcome i with probability $\langle P_i, \rho \rangle$, where $\langle A, B \rangle = \mathrm{Tr}\big(A^\dagger B\big)$ is the *Hilbert-Schmidt inner product* defined on the linear space of Hermitian matrices. Note that $\langle A, B \rangle$ is a real number for any Hermitian matrices A and B, and is non-negative if A and B are positive semidefinite.

Given a finite-dimensional Hilbert space $\mathcal{H} = \mathbb{C}^n$, $\mathrm{L}(\mathcal{H})$ denotes the set of linear operators acting on $\mathcal{H}$, i.e., the set of all $n \times n$ complex matrices over $\mathcal{H}$. A linear operator that maps matrices to matrices, i.e., a mapping $\Phi : \mathrm{L}(\mathcal{B}) \to \mathrm{L}(\mathcal{A})$, is called a *super-operator*. The adjoint super-operator $\Phi^\dagger : \mathrm{L}(\mathcal{A}) \to \mathrm{L}(\mathcal{B})$ is uniquely determined by the equation $\langle A, \Phi(B) \rangle = \langle \Phi^\dagger(A), B \rangle$. A super-operator $\Phi : \mathrm{L}(\mathcal{B}) \to \mathrm{L}(\mathcal{A})$ is called *positive* if it maps PSD matrices to PSD matrices. There exists a linear bijection between matrices $R \in \mathrm{L}(\mathcal{A} \otimes \mathcal{B})$ and super-operators $\Phi : \mathrm{L}(\mathcal{B}) \to \mathrm{L}(\mathcal{A})$ known as the *Choi-Jamiołkowski isomorphism*. Specifically, for a super-operator Φ its *Choi matrix* is:

$$C_\Phi = \sum_{1 \le i,j \le m} \Phi(E_{i,j}) \otimes E_{i,j} \in \mathrm{L}(\mathcal{A} \otimes \mathcal{B}), \tag{5.1}$$

where $\{E_{i,j}\}_{i,j=1}^{m}$ is the standard orthonormal basis of $\mathrm{L}(\mathcal{B}) = \mathbb{C}^{m\times m}$. Conversely, given an operator $R = \sum_{1\leq i,j\leq m} A_{i,j} \otimes E_{i,j} \in \mathrm{L}(\mathcal{A}\otimes\mathcal{B})$, we can define $\Phi_R : \mathrm{L}(\mathcal{B}) \to \mathrm{L}(\mathcal{A})$ by setting $\Phi_R(E_{i,j}) = A_{i,j}$ from which it easily follows that $C_{\Phi_R} = R$. Explicitly, we also have that

$$\Phi_R(B) = \mathrm{Tr}_{\mathcal{B}}(R(\mathbb{1}_{\mathcal{A}} \otimes B^{\top})), \tag{5.2}$$

where the partial trace $\mathrm{Tr}_{\mathcal{B}} : \mathcal{L}(\mathcal{A}\otimes\mathcal{B}) \to \mathcal{L}(\mathcal{A})$ is the *unique* function that satisfies:

$$\mathrm{Tr}_{\mathcal{B}}(A \otimes B) = A\,\mathrm{Tr}(B),\ \forall A, B.$$

Moreover, the adjoint map is $\mathrm{Tr}_{\mathcal{B}}^{\dagger}(A) = A \otimes \mathbb{1}_{\mathcal{B}}$. Lastly, a superoperator Φ is completely positive (i.e., $\mathbb{1}_m \otimes \Phi$ is positive for all $m \in \mathbb{N}$) if and only if the Choi matrix of Φ is positive semidefinite. In particular, if the Choi matrix of the super-operator Φ is PSD, it follows that Φ is positive.

With these concepts, we can define the notion of quantum zero-sumgames which will be specialized and studied within this dissertation.

5.2.2 Quantum Zero-Sum Games

The notion of quantum zero-sum games which we study is precisely the formulation of [90], which is a widely studied formulation in the literature which models strategic interactions between rational players that exchange quantum information [21, 186]. In their model, called *non-interactive* quantum games, each player has access to a register of fixed size. In the two-player case, Alice has access to register $\mathcal{A} = \mathbb{C}^n$ and Bob has access to register $\mathcal{B} = \mathbb{C}^m$. To play the game, Alice and Bob independently prepares a state (i.e. Alice prepares a density matrix $\rho \in \mathcal{A}$ and Bob prepares density matrix $\sigma \in \mathcal{B}$). Their individual states are then sent to a referee, who performs a measurement (POVM) on the joint state $\rho \otimes \sigma$ to determine the payoffs to each player.

Thus, the referee's measurement can be described by a collection

$$\{R_a : a \in \mathcal{S}\} \subset \mathrm{Pos}(\mathcal{A}\otimes\mathcal{B}) \tag{5.3}$$

which satisfies the condition $\sum_{a=1}^{k} R_a = \mathbb{1}_{\mathcal{A}\otimes\mathcal{B}}$. We associate each possible measurement outcome a with a payoff for each player. In the zero-sum case, if the payoff that Alice receives from the measurement is $v(a)$, then Bob's corresponding payoff will be $-v(a)$. Thus, for a given choice of ρ and σ, Alice's expected payoff is:

$$u(\rho,\sigma) = \sum_{a=1}^{k} v(a)\langle R_a, \rho\otimes\sigma\rangle = \langle R, \rho\otimes\sigma\rangle \tag{5.4}$$

where $R = \sum_{a=1}^{k} v(a)R_a$. Likewise, Bob's corresponding expected payoff is $-u(\rho,\sigma) = -\langle R, \rho\otimes\sigma\rangle$. R will be referred to throughout the rest of this chapter as a *payoff observable*. A necessary and sufficient condition for matrix R acting on $\mathcal{A}\otimes\mathcal{B}$ to be obtained from some real valued payoff function v is that R is Hermitian.

Moreover, note that from Equation 5.2, we can equivalently define the expected payoff of Alice as $u(\rho, \sigma) = \langle \rho, \Phi(\sigma^\top) \rangle$. This is because:

$$\langle \rho, \Phi(\sigma^\top) \rangle = \langle \rho, \mathrm{Tr}_{\mathcal{B}}(R(\mathbb{1}_{\mathcal{A}} \otimes \sigma)) \rangle = \langle \rho \otimes \mathbb{1}_{\mathcal{B}}, \mathrm{Tr}_{\mathcal{B}}(R(\mathbb{1}_{\mathcal{A}} \otimes \sigma)) \rangle = \mathrm{Tr}(R(\rho \otimes \sigma))$$

A similar argument applies for the expected payoff of Bob. In the rest of this chapter, we will utilize the latter formulation of expected payoff for each player. For notational convenience, we will drop the transpose from the $\sigma^\top$ term. This can be thought of as Bob selecting $\sigma^\top$ as his strategy in the learning dynamics to follow.

In the matrix/quantum setting, we define the pair of mixed states (ρ^*, σ^*) as a Nash equilibrium of R if

$$u(\rho^*, \sigma^*) \geq u(\rho, \sigma^*) \quad \text{and} \quad u(\rho^*, \sigma^*) \geq u(\rho^*, \sigma) \tag{5.5}$$

for all ρ, σ. That is, neither Alice nor Bob would prefer to unilaterally deviate from playing ρ^* and σ^* respectively. Moreover, an equilibrium (ρ^*, σ^*) is called *fully mixed* if the payoff of a player for deviating from the equilibrium to any other strategy remains exactly equal to their equilibrium payoff. A standard example in classical game theory is Rock-Paper-Scissors, where the mixed Nash equilibrium $[1/3, 1/3, 1/3]$ gives payoff 0 to both players.

5.2.3 Quantum Information Theory

The setting we study is a non-commutative extension of classical game theory. As such, we will utilize several key concepts from quantum information theory in our analysis. We present these concepts below.

Shannon Entropy. The Shannon entropy of a random variable X where each strategy x is obtained with probability $p(x)$ is given by $H(X) = -\sum_x p(x) \log p(x)$. Intuitively, this is a measure of randomness or uncertainty in the system. A natural generalization of the Shannon entropy to a quantum regime is the von Neumann entropy. For a quantum system defined by density matrix ρ, the von Neumann entropy is given by $S(\rho) = -\mathrm{Tr}(\rho \log \rho)$.

Bregman Divergence. We are also interested in the notion of Bregman divergence, which measures the distance between two points. Let $x^* \in \mathcal{X}$ be a Nash equilibrium and let $x \in \mathcal{X}$ be an arbitrary strategy profile. Also, let F be a continuously-differentiable, strictly convex function. The Bregman divergence from x^* to x is given by $D_F(x^* \| x) = F(x^*) - F(x) - \langle \nabla F(x), x^* - x \rangle$.

The Bregman divergence can also be defined in the quantum information theoretic context. We define the quantum relative entropy between two quantum states using the von Neumann entropy $S(\rho)$. In particular, the quantum relative entropy between two quantum states ρ and σ is defined as $S(\rho \| \sigma) = \mathrm{Tr}\left(\rho(\log \rho - \log \sigma)\right)$.

Algorithm 1 Matrix Multiplicative Weights Update (MMWU)

Initialize: $A_0 = \mathbb{1}_{\mathcal{A}}$, $\rho_0 = A_0/\mathrm{Tr}(A_0)$, $B_0 = \mathbb{1}_{\mathcal{B}}$, and $\sigma_0 = B_0/\mathrm{Tr}(B_0)$.
for $j = 1 \dots t$ **do**
  $A_j = \exp\left(\mu \sum_{i=0}^{j-1} \Phi(\sigma_i)\right)$
  $\rho_j = A_j/\mathrm{Tr}(A_j)$
  $B_j = \exp\left(-\mu \sum_{i=0}^{j-1} \Phi^*(\rho_i)\right)$
  $\sigma_j = B_j/\mathrm{Tr}(B_j)$
end for

5.2.4 Dynamical Systems

The necessary preliminaries for dynamical systems in this chapter are presented in Chapter 2.3. We further specialize the definitions as necessary within our framework.

Flows. Consider a differential equation $\dot{x} = f(x)$ on a topological space $\mathcal{X}$. In Section 5.4, we introduce the notion of matrix replicator dynamics, which are Lipschitz continuous differential equations. Hence, a unique flow ϕ of these replicator dynamics exists.

Diffeomorphisms of Flows. A function $\mathbf{f}$ between two topological spaces is called a diffeomorphism if i) $\mathbf{f}$ is a bijection, ii) $\mathbf{f}$ is continuously differentiable, iii) $\mathbf{f}$ has a continuously differentiable inverse. Two flows $\Phi^t : A \to A$ and $\Psi^t : B \to B$ are diffeomorphic if there exists a diffeomorphism $\mathbf{f} : A \to B$ such that for each $x \in A$ and $t \in \mathbb{R}$, $\mathbf{f}(\Phi^t(p)) = \Psi^t(\mathbf{f}(p))$. For the purpose of our analysis, the replicator dynamics defined in (MRD) are translated via a diffeomorphism from the interior of $\mathcal{P}$ to a space $\mathcal{C} = \Pi_{i \in V} \mathbb{R}^{n-1}$, which allows us to show certain desirable properties of the dynamics.

Poincarè Recurrence. Much like our analysis from previous chapters, our goal is to study the dynamical properties of (MRD) in quantum zero-sum games. The primary result we want to show is Poincarè recurrence, and we present the definition thereof again for convenience.

Theorem 5.2.1 (Poincarè recurrence)**.** *If a flow preserves volume and has only bounded orbits then for each open set there exist orbits that intersect the set infinitely often.*

5.3 MMWU in Quantum Zero-Sum Games

In [90], the MMWU algorithm for zero-sum games is shown to exhibit time-average convergence to an approximate Nash equilibrium in two-player quantum zero-sum games. We repeat the algorithm in Algorithm 1 for convenience. Note that here we focus specifically on two-player games, and utilize the expected payoffs defined via super-operators $\Phi : \mathrm{L}(\mathcal{A}) \to \mathrm{L}(\mathcal{B})$ as seen in Equation 5.2. Moreover, μ is the step-size of the update rule.

In this section, we examine closely the update steps for each player in the MMWU algorithm (Algorithm 1) and analyze the limiting behaviour of the total quantum relative entropy in the system. First, we introduce two useful facts which will aid in the analysis.

Fact 5.3.1 (Golden-Thompson inequality [71, 167])**.** *Let A, B be Hermitian matrices. Then*

$$\operatorname{Tr}\exp(A+B) \leq \operatorname{Tr}\exp(A)\exp(B)$$

Fact 5.3.2. *Let $0 \leq A \leq \mathbb{1}$ be a PSD matrix and δ be a real number. Then,*

$$\exp(\delta A) \leq \mathbb{1} + \delta\exp(\delta)A$$

Next, we put forward two results which will help us prove Theorem 5.3.1. In particular, these lemmas present important relationships between the values of A and B from one time step to the next and the total quantum relative entropy within the system. We first define $\Delta S(\rho^*\|\rho_j) = S(\rho^*\|\rho_j) - S(\rho^*\|\rho_{j-1})$ and $\Delta S(\sigma^*\|\sigma_j) = S(\sigma^*\|\sigma_j) - S(\sigma^*\|\sigma_{j-1})$.

Lemma 5.3.1. *The sum of quantum relative entropies in a quantum zero-sum game with fully-mixed Nash equilibrium is given by:*

$$\Delta S(\rho^*\|\rho_j) + \Delta S(\sigma^*\|\sigma_j) = \log\frac{\operatorname{Tr}A_j}{\operatorname{Tr}A_{j-1}} + \log\frac{\operatorname{Tr}B_j}{\operatorname{Tr}B_{j-1}} \tag{5.6}$$

Proof. We use the definition of the quantum relative entropy. First, we consider the change in quantum relative entropy from timestep $j-1$ to j.

$$\begin{aligned}
S(\rho^*\|\rho_j) - S(\rho^*\|\rho_{j-1}) &= \operatorname{Tr}\left(\rho^*(\log\rho_{j-1} - \log\rho_j)\right) \\
&= \operatorname{Tr}\left(\rho^*\left(\log\frac{A_{j-1}}{\operatorname{Tr}A_{j-1}} - \log\frac{A_j}{\operatorname{Tr}A_j}\right)\right) \\
&= \operatorname{Tr}\left(\rho^*\left(\log A_{j-1} - \log A_j + \log\operatorname{Tr}A_j - \log\operatorname{Tr}A_{j-1}\right)\right) \\
&= \operatorname{Tr}\left(\rho^*\left(-\mu\Phi(\sigma_{j-1}) + \log\operatorname{Tr}A_j - \log\operatorname{Tr}A_{j-1}\right)\right)
\end{aligned}$$

Hence,

$$S(\rho^*\|\rho_j) - S(\rho^*\|\rho_{j-1}) = -\mu\operatorname{Tr}(\rho^*\Phi(\sigma^*)) + \operatorname{Tr}(\rho^*(\log\operatorname{Tr}A_j - \log\operatorname{Tr}A_{j-1})) \tag{5.7}$$

Similarly, we know that

$$S(\sigma^*\|\sigma_j) - S(\sigma^*\|\sigma_{j-1}) = -\mu\operatorname{Tr}(\sigma^*\Phi^\dagger(\rho^*)) + \operatorname{Tr}(\sigma^*(\log\operatorname{Tr}B_j - \log\operatorname{Tr}B_{j-1})) \tag{5.8}$$

Summing up equations 5.7 and 5.8:

$$\begin{aligned}
&\Delta S(\rho^*\|\rho_j) + \Delta S(\sigma^*\|\sigma_j) \\
&= \mu\left[-\operatorname{Tr}(\rho^*\Phi(\sigma^*)) + \operatorname{Tr}(\sigma^*\Phi^\dagger(\rho^*))\right] + \log\frac{\operatorname{Tr}A_j}{\operatorname{Tr}A_{j-1}} + \log\frac{\operatorname{Tr}B_j}{\operatorname{Tr}B_{j-1}} \\
&= \log\frac{\operatorname{Tr}A_j}{\operatorname{Tr}A_{j-1}} + \log\frac{\operatorname{Tr}B_j}{\operatorname{Tr}B_{j-1}}
\end{aligned}$$

□

Lemma 5.3.2. *The following trace inequalities hold for PSD matrices A and B updated with* `MMWU`*:*

$$\Delta S(\rho^* \| \rho_j) + \Delta S(\sigma^* \| \sigma_j) \geq \mu \exp(-\mu) \operatorname{Tr}\left(\rho_j \Phi(\sigma_{j-1})\right) - \mu \exp(\mu) \operatorname{Tr}\left(\rho_{j-1} \Phi(\sigma_j)\right) \quad (5.9)$$

$$\Delta S(\rho^* \| \rho_j) + \Delta S(\sigma^* \| \sigma_j) \leq \mu \exp(\mu) \operatorname{Tr}\left(\rho_{j-1} \Phi(\sigma_{j-1})\right) - \mu \exp(-\mu) \operatorname{Tr}\left(\rho_{j-1} \Phi(\sigma_{j-1})\right) \quad (5.10)$$

Proof. We first show the upper bound on the sum $\log\left(\frac{\operatorname{Tr} A_j}{\operatorname{Tr} A_{j-1}}\right) + \log\left(\frac{\operatorname{Tr} B_j}{\operatorname{Tr} B_{j-1}}\right)$. Note that the definition of A_j is:

$$A_j = \exp\left(\mu \sum_{i=0}^{j-1} \Phi(\sigma_i)\right)$$

We can alternatively write:

$$A_j = \exp\left(\log(A_{j-1}) + \mu \Phi(\sigma_{j-1})\right) \quad (5.11)$$

Hence, by taking the trace of A_{j-1} and applying Facts 5.3.1 and 5.3.2:

$$\begin{aligned}
\operatorname{Tr}(A_{j-1}) &\leq \operatorname{Tr}\left[A_j \exp(-\mu \Phi(\sigma_{j-1}))\right] && \text{(Fact 5.3.1)} \\
&\leq \operatorname{Tr}\left[A_j (\mathbb{1} - \mu \exp(-\mu) \Phi(\sigma_{j-1}))\right] && \text{(Fact 5.3.2)} \\
&= (\operatorname{Tr}(A_j))(1 - \mu \exp(-\mu) \operatorname{Tr}\left(\rho_j \Phi(\sigma_{j-1})\right) \\
&\leq (\operatorname{Tr}(A_j)) \exp(-\mu \exp(-\mu) \operatorname{Tr}(\rho_j \Phi(\sigma_{j-1}))). && (1 + x \leq \exp(x))
\end{aligned}$$

Similarly,

$$\begin{aligned}
\operatorname{Tr}(B_{j-1}) &\leq \operatorname{Tr}\left[B_j \exp\left(\mu \Phi^\dagger(\rho_{j-1})\right)\right] \\
&\leq \operatorname{Tr}\left[B_j (\mathbb{1} + \mu \exp(\mu) \Phi^\dagger(\rho_{j-1}))\right] \\
&= (\operatorname{Tr}(B_j))(1 + \mu \exp(\mu) \operatorname{Tr}\left(\rho_{j-1} \Phi(\sigma_j)\right)) \\
&\leq (\operatorname{Tr}(B_j)) \exp(\mu \exp(\mu) \operatorname{Tr}(\rho_{j-1} \Phi(\sigma_j))).
\end{aligned}$$

By rearranging and taking matrix logarithm on both sides, we obtain:

$$\begin{aligned}
\log\left(\frac{\operatorname{Tr} A_j}{\operatorname{Tr} A_{j-1}}\right) &\geq \mu \exp(-\mu) \operatorname{Tr}\left(\rho_j \Phi(\sigma_{j-1})\right) \\
\log\left(\frac{\operatorname{Tr} B_j}{\operatorname{Tr} B_{j-1}}\right) &\geq -\mu \exp(\mu) \operatorname{Tr}\left(\rho_{j-1} \Phi(\sigma_j)\right)
\end{aligned} \quad (5.12)$$

The inequality in Equation 5.9 follows by summing up the two inequalities obtained above and applying Lemma 5.3.1. Likewise, we can perform similar calculations on

$\mathrm{Tr}(A_j)$ and $\mathrm{Tr}(B_j)$ to obtain the statement of Equation 5.10.

$$\begin{aligned}
\mathrm{Tr}\, A_j &\leq \mathrm{Tr}\left[A_{j-1}\exp(\mu\Phi(\sigma_{j-1}))\right] \\
&\leq \mathrm{Tr}\left[A_{j-1}(\mathbb{1}+\mu\exp(\mu)\Phi(\sigma_{j-1}))\right] \\
&= (\mathrm{Tr}\, A_{j-1})(1+\mu\exp(\mu)\,\mathrm{Tr}\left(\rho_{j-1}\Phi(\sigma_{j-1})\right)) \\
&\leq (\mathrm{Tr}\, A_{j-1})\exp(\mu\exp(\mu)\,\mathrm{Tr}(\rho_{j-1}\Phi(\sigma_{j-1})))
\end{aligned}$$

Similarly,

$$\begin{aligned}
\mathrm{Tr}\, B_j &\leq \mathrm{Tr}\left[B_{j-1}\exp\Big(-\mu\Phi^\dagger(\rho_{j-1})\Big)\right] \\
&\leq \mathrm{Tr}\left[B_{j-1}(\mathbb{1}-\mu\exp(-\mu)\Phi^\dagger(\rho_{j-1}))\right] \\
&= (\mathrm{Tr}\, B_{j-1})(1-\mu\exp(-\mu)\,\mathrm{Tr}\left(\rho_{j-1}\Phi(\sigma_{j-1})\right)) \\
&\leq (\mathrm{Tr}\, B_{j-1})\exp(-\mu\exp(-\mu)\,\mathrm{Tr}(\rho_{j-1}\Phi(\sigma_{j-1})))
\end{aligned}$$

Again rearranging and taking matrix logarithm on both sides:

$$\begin{aligned}
\log\frac{\mathrm{Tr}\, A_j}{\mathrm{Tr}\, A_{j-1}} &\leq \mu\exp(\mu)\,\mathrm{Tr}\left(\rho_{j-1}\Phi(\sigma_{j-1})\right) \\
\log\frac{\mathrm{Tr}\, B_j}{\mathrm{Tr}\, B_{j-1}} &\leq -\mu\exp(-\mu)\,\mathrm{Tr}\left(\rho_{j-1}\Phi(\sigma_{j-1})\right)
\end{aligned} \tag{5.13}$$

Thus, an upper bound on $\Delta S(\rho^*\|\rho_j) + \Delta S(\sigma^*\|\sigma_j)$ is:

$$\Delta S(\rho^*\|\rho_j) + \Delta S(\sigma^*\|\sigma_j) \leq \mu\exp(\mu)\,\mathrm{Tr}\left(\rho_{j-1}\Phi(\sigma_{j-1})\right) - \mu\exp(-\mu)\,\mathrm{Tr}\left(\rho_{j-1}\Phi(\sigma_{j-1})\right)$$

□

In many practical scenarios, one would use decreasing step-sizes when running `MMWU`. As such, we utilize Corollary 5.3.2 and take the limit as step-size μ goes to 0 in order to show the following result:

Theorem 5.3.1. *The sum of quantum relative entropies between the player's strategies and a fully mixed Nash equilibrium in a two-player zero-sum quantum game tends to zero when step-size $\mu \to 0$. Specifically,*

$$\lim_{\mu\to 0}\frac{1}{\mu}\left(\Delta S(\rho^*\|\rho_j) + \Delta S(\sigma^*\|\sigma_j)\right) = 0 \tag{5.14}$$

Proof. Consider the `MMWU` update from time $j-1$ to j. As $\mu \to 0$, ρ_j and σ_j do not change more than $\mathcal{O}(\mu)$. Indeed, since all payoffs in the game are bounded in the `MMWU` update from time $j-1$ to j, all entries in the numerators increase by at most $\exp(\mathcal{O}(\mu)) = 1+\mathcal{O}(\mu)$. Likewise, the denominator is at least as large, but also upper bounded by the previous value of the denominator multiplied by $(1+\mathcal{O}(\mu))$. Hence, every entry in the outputs ρ_j and σ_j are at most $\mathcal{O}(\mu)$ from ρ_{j-1} and σ_{j-1}. Moreover, we have that $Tr(\rho_j\Phi(\sigma_{j-1}))$, $Tr(\rho_{j-1}\Phi(\sigma_j))$ and $Tr(\rho_{j-1}\Phi(\sigma_{j-1}))$ are all within $\mathcal{O}(\mu)$ of each other. Using a Taylor expansion, both the upper bounds and lower bounds

given in Equations 5.9,5.10 are of the order of $\mathcal{O}(\mu^2)$ and the statement of the theorem follows. □

Theorem 5.3.1 further motivates an investigation into the continuous time variant of MMWU, which we call quantum/matrix replicator dynamics. In particular, we show that in the continuous case, the sum of quantum relative entropies is invariant. We introduce and study matrix replicator dynamics in detail in Section 5.4.

5.4 Replicator Dynamics in Quantum Zero-Sum Games

We have seen that in the case of discrete-time dynamics (MMWU), as the step-size becomes infinitesimal, the total quantum relative entropy in the system remains invariant. A natural question would then be: does the same result hold in continuous time? In order to explore this question in greater detail, we first need to define the continuous analogue of MMWU, the matrix replicator dynamics (MRD), in the quantum setting.

We start by rewriting the MMWU update steps, but now defined over a continuous time interval $[0, t]$:

$$A(t) = \int_0^t \Phi(\sigma(\tau))d\tau \tag{5.15}$$

$$\rho(t) = \exp(A(t))/\mathrm{Tr}(\exp(A(t))) \tag{5.16}$$

$$B(t) = -\int_0^t \Phi^\dagger(\rho(\tau))d\tau \tag{5.17}$$

$$\sigma(t) = \exp(B(t))/\mathrm{Tr}(\exp(B(t))) \tag{5.18}$$

Note here that we shift the exponential terms from the definition of $A(t)$ and $B(t)$ to the corresponding $\rho(t)$ and $\sigma(t)$ terms. This will help simplify some of the proof techniques later on in this chapter. Furthermore, in the rest of this chapter, we will typically drop the explicit dependence on t to ease the notational burden in the proofs.

Observation 5.4.1. *As $\mu \to 0$, the discrete-time trajectories ρ_j and σ_j defined in Algorithm 1 are equal to the continuous-time trajectories $\rho(t)$ and $\sigma(t)$ defined in Equations 5.16 and 5.18.*

Proof. It suffices to show that at the limit, the integrals $A(t)$ and $B(t)$ defined in 5.15 and 5.17 can be written in the discrete summation form as seen in Algorithm 1. First we write the Riemann sum for $A(t)$ by taking j infinitesimal intervals in time interval $[0, t]$, each of width μ:

$$A(t) = \int_0^t \Phi(\sigma(\tau))d\tau = \sum_{i=0}^{j-1} \left(\frac{t}{j}\right) \Phi(\sigma_i)$$

However $t = \mu j$, so the above can be written as $\sum_{i=0}^{j-1} \mu\Phi(\sigma_i)$. Taking limit as $\mu \to 0$, it is clear to see that the matrix exponent of the continuous-time trajectory $A(t)$ is equal to

its discrete-time counterpart A_j. Hence, the trajectories $\rho(t)$ and ρ_j are equivalent at the limit. A similar argument holds for $\sigma(t)$ and σ_j at the limit. □

We now define the *matrix replicator dynamics* as:

$$d\rho/dt = \frac{d}{dt}\left(\frac{\exp(A)}{\mathrm{Tr}(\exp(A))}\right), \quad d\sigma/dt = \frac{d}{dt}\left(\frac{\exp(B)}{\mathrm{Tr}(\exp(B))}\right) \tag{MRD}$$

It is worth noting that typically, one can write the replicator equations in a form that describes the relative utility that one agent obtains as compared to the average utility overall. However, in the quantum case this is not possible in general, since it relies on the assumption that $\int_0^t \Phi(\sigma(\tau))d\tau$ and $\Phi(\sigma(t))$, and respectively $\int_0^t \Phi^\dagger(\rho(\tau))d\tau$ and $\Phi^\dagger(\rho(t))$ commute.

As a consequence of Observation 5.4.1, we can also conclude that the dynamical system described by the replicator dynamics defined in Equations MRD is a limit case of the dynamical system described by the MMWU algorithm as $\mu \to 0$.

We are now able to state the main theorem for quantum relative entropy in matrix replicator dynamics. The proof of this theorem utilizes Observation 5.4.1 to show that at the limit $\mu \to 0$, the rate of change of the quantum relative entropy is exactly the same whether we are in the discrete or continuous setting. We then apply Theorem 5.3.1 to complete the proof.

Theorem 5.4.1. *When applying matrix/quantum replicator dynamics in a two-player quantum zero-sum game with a fully-mixed Nash equilibrium (ρ^*, σ^*), the sum of quantum relative entropies between the fully-mixed Nash equilibrium and the state of the system $(\rho(t), \sigma(t))$ is invariant on every system trajectory, i.e.:*

$$\frac{d\big(S(\rho^*\|\rho(t)) + S(\sigma^*\|\sigma(t))\big)}{dt} = 0 \tag{5.19}$$

Proof. First we note that by expanding the limit definition of the derivative we obtain:

$$\begin{aligned}&\frac{d\big(S(\rho^*\|\rho(t)) + S(\sigma^*\|\sigma(t))\big)}{dt}\\ &= \lim_{h\to 0} \frac{\big(S(\rho^*\|\rho(t+h)) + S(\sigma^*\|\sigma(t+h))\big) - \big(S(\rho^*\|\rho(t)) + S(\sigma^*\|\sigma(t))\big)}{h}\end{aligned}$$

Here, we apply Observation 5.4.1 in the following manner: If we perform the substitution $j \to t+h$, $j-1 \to t$ and $\mu \to h$, we obtain a reformulation of the same limit in the language of step-sizes of duration μ. As $\mu \to 0$, the difference between the continuous time flow solution and its discrete-time Euler approximation is of order $O(\mu^2)$. Thus, we can compute this limit evaluated along the points of the discrete-time MMWU trajectory. This limit by Theorem 5.3.1 is equal to zero.

□

5.4.1 Connections between Matrix and Classical Replicator Dynamics

In this section we show that in the special case where commutativity holds, the matrix replicator dynamics (MRD) are equivalent to that of the classical replicator dynamics. Typically, in the classical case, one can write the replicator dynamics in a form that represents the utilities obtained by each player:

$$\dot{x} = x\left(Ay - \left(x^\top Ay\right)\mathrm{vec}\mathbb{1}\right)$$

where x and y are n-dimensional probability vectors representing the strategies of each player, A is the payoff matrix and $\mathrm{vec}\mathbb{1}$ is the n-dimensional all-one vector. In the matrix setting, we require a few additional results and assumptions in order to write (MRD) in the classical form. The following lemma arrives directly as a result of the series definition of the matrix exponential.

Lemma 5.4.1. *If matrices P and $\frac{dP}{dt}$ commute then*

$$\frac{d(\exp(P))}{dt} = \frac{dP}{dt}\cdot\exp(P) = \exp(P)\cdot\frac{dP}{dt}$$

Proof.

$$\begin{aligned}\frac{d}{dt}(\exp(P)) &= \frac{d}{dt}\left[\mathbb{1} + P + \frac{1}{2}P^2 + \frac{1}{3!}P^3 + \cdots\right]\\ &= \frac{dP}{dt}\left[\mathbb{1} + P + \frac{1}{2}P^2 + \cdots\right]\\ &= \frac{dP}{dt}\cdot\exp(P) = \exp(P)\cdot\frac{dP}{dt} \qquad \text{(by commutativity)}\end{aligned}$$

□

By leveraging Lemma 5.4.1 and assuming that commutativity holds, we can derive the following:

Lemma 5.4.2. *If $\int_0^t \Phi(\sigma(\tau))d\tau$ & $\Phi(\sigma(t))$, and $\int_0^t \Phi^\dagger(\rho(\tau))d\tau$ & $\Phi^\dagger(\rho(t))$ commute, then the replicator system when applied to a two-player zero-sum quantum game is equivalent to:*

$$d\rho/dt = \rho[\Phi(\sigma) - \mathrm{Tr}(\rho\Phi(\sigma))\mathbb{1}] \tag{5.20}$$

$$d\sigma/dt = \sigma[-\Phi^\dagger(\rho) + \mathrm{Tr}\left(\sigma\Phi^\dagger(\rho)\right)\mathbb{1}] \tag{5.21}$$

Proof. By definition we have that $\rho = \exp(A)/\mathrm{Tr}(\exp(A))$ where $A = \int_0^t \Phi(\sigma(\tau))d\tau$. By applying Lemma 5.4.1 we have that if $\int_0^t \Phi(\sigma(\tau))d\tau$ and $\Phi(\sigma(t))$ commute then $d\exp(A)/dt = \exp(A)\Phi(\sigma) = \Phi(\sigma)\exp(A)$.

$$\begin{aligned} d\rho/dt &= \frac{(d\exp(A)/dt)\operatorname{Tr}(\exp(A)) - \exp(A)d(\operatorname{Tr}(\exp(A)))/dt}{(\operatorname{Tr}\exp(A))^2} \\ &= \frac{(d\exp(A)/dt)\operatorname{Tr}(\exp(A)) - \exp(A)\operatorname{Tr}(d\exp(A)/dt)}{(\operatorname{Tr}\exp(A))^2} \\ &= \frac{\exp(A)\Phi(\sigma)\operatorname{Tr}(\exp(A)) - \exp(A)\operatorname{Tr}(\exp(A)\Phi(\sigma))}{(\operatorname{Tr}\exp(A))^2} \\ &= \rho[\Phi(\sigma) - \operatorname{Tr}(\rho\Phi(\sigma))\mathbb{1}] \end{aligned}$$

The proof in the case of $d\sigma/dt$ is similar. □

We can see that Equations 5.20 and 5.21 take a familiar form - indeed, the matrix replicator dynamics model the difference in utility obtained by each player compared to the average utility in much the same way as the classical replicator dynamics. In general, commutativity holds only in the case when all matrices $\Phi^\dagger(\rho(t))$ and $\Phi(\sigma(t))$ are diagonal, which is precisely the classical setting. The assumption that commutativity holds fails once we enter the realm of quantum information.

Reminiscent of similar results in classical evolutionary game theory (which we have leveraged in Chapters 3 and 4), one can write a proof for the invariance of total entropy (i.e. KL-divergence in the classical setting) directly using the formulation of replicator dynamics. However, given the commutativity assumption, this 'proof' only holds in the case where we are writing classical probability vectors and game payoff matrices using the quantum notation.

Proof of Theorem 5.4.1 (Classical case). We will prove that:

$$d\operatorname{Tr}\big(\rho^*(\log\rho^* - \log\rho(t)) + \sigma^*(\log\sigma^* - \log\sigma(t))\big)/dt = 0$$

To do this, we will focus on the terms related to the first player, i.e., $d\operatorname{Tr}\big(\rho^*(\log\rho^* - \log\rho(t))\big)/dt = -d\operatorname{Tr}\big(\rho^*\log\rho(t))\big)/dt$.

$$\begin{aligned} d\operatorname{Tr}\big(\rho^*\log\rho(t))\big)/dt &= \operatorname{Tr}\big(\rho^* d(\log\rho(t))/dt\big) \\ &= \operatorname{Tr}\big(\rho^*\rho(t)^{-1}d\rho(t)/dt\big) \quad \text{(by Lemma 5.4.1)} \\ &= \operatorname{Tr}\big(\rho^*\rho^{-1}\rho[\Phi(\sigma) - \operatorname{Tr}(\rho\Phi(\sigma))I]\big) \quad \text{(by Lemma 5.4.2)} \\ &= \operatorname{Tr}\big(\rho^*[\Phi(\sigma) - \operatorname{Tr}(\rho\Phi(\sigma))I]\big) \\ &= \operatorname{Tr}\big(\rho^*\Phi(\sigma)\big) - \operatorname{Tr}\big(\operatorname{Tr}(\rho\Phi(\sigma))\rho^*]\big) \\ &= \operatorname{Tr}\big(\rho^*\Phi(\sigma)\big) - \operatorname{Tr}(\rho\Phi(\sigma)) \quad (\text{Since } \operatorname{Tr}\left(\rho^*\right) = 1) \\ &= \operatorname{Tr}\big(\rho^*\Phi(\sigma^*)\big) - \operatorname{Tr}(\rho\Phi(\sigma)) \end{aligned}$$

Note in the second equality that $\rho(t)^{-1}$ exists since the exponential of a matrix is always an invertible matrix. The last equality comes from the assumption that the equilibrium (ρ^*, σ^*) is 'fully mixed', i.e. the payoff of a player when deviating from the Nash

equilibrium to any other strategy remains exactly equal to their equilibrium payoff (e.g., Rock-Paper-Scissors). A similar analysis for the second player results in:

$$d\mathrm{Tr}\big(\sigma^* \log \sigma(t))\big)/dt \quad = \quad -\mathrm{Tr}\left(\sigma^* \Phi^\dagger(\rho^*)\right) + \mathrm{Tr}\Big(\sigma \Phi^\dagger(\rho)\Big)$$

These terms cancel out and the theorem follows. □

In general, we cannot utilize the above argument to show the invariance of quantum relative entropy, but rather need an argument tailored to the quantum setting which is presented in the proof of Theorem 5.4.1.

5.5 Poincaré Recurrence in Quantum Zero-Sum Games

Now that we have described analytical results surrounding the day-to-day behaviour of matrix replicator dynamics, we seek to understand the *dynamics* of the trajectories. After all, invariance of quantum relative entropy does not fully describe how the system moves over time. We show that for any two-player zero-sum quantum game updated with matrix replicator dynamics, the system exhibits *Poincaré recurrence*, insofar as the game is zero-sum and has a fully-mixed Nash equilibria. As introduced in Section 5.2, the notion of Poincaré recurrence is a weaker version of periodicity. To be precise, for almost all initial conditions $\rho_0 \in \mathcal{A}$ and $\sigma_0 \in \mathcal{B}$, the matrix replicator dynamics return arbitrarily close to (ρ_0, σ_0) infinitely often.

Theorem 5.5.1. *The matrix replicator dynamics given in* (MRD) *are Poincaré recurrent in any two player quantum zero-sum game which has a fully-mixed Nash equilibrium.*

The proof of this main theorem involves carefully piecing together several auxiliary results, which we will describe in the rest of the section. Furthermore, we stress that due to the non-commutative nature of quantum systems, the standard (classical) approach of differentiating the discrete-time dynamics in the primal space of probability distributions does not apply directly unless we have the highly unlikely situation where $\int_0^t \Phi(\sigma(\tau))d\tau$ and $\Phi(\sigma(t))$ (resp. $\int_0^t \Phi^\dagger(\rho(\tau))d\tau$ and $\Phi^\dagger(\rho(t))$) commute.

For the proof in the quantum setting, we first define a *canonical transformation* on the space of the matrices $A(t)$ and $B(t)$, which will be crucial in proving the theorem.

Definition 5.5.1 (Canonical transformation)**.** *We define the canonical transformation of $A'(t)$ and $B'(t)$ to be a mapping of $A(t)$ and $B(t)$ as defined by Equations 5.15 and 5.17. In particular, we define*

$$\begin{aligned} A'(t) &= A(t) - (v^\dagger A(t) v)\mathbb{1} \\ B'(t) &= B(t) - (v^\dagger B(t) v)\mathbb{1} \end{aligned} \tag{5.22}$$

where v is a fixed vector defined as $v = [1, 0 \ldots 0]^\top$, such that the values of $v^\dagger A(t) v$ and $v^\dagger B(t) v$ are real numbers corresponding to the $(1,1)$-th element of matrices $A(t)$ and $B(t)$ for all t. Notice that this creates matrices $A'(t)$ and $B'(t)$ which have 0 as the $(1,1)$-th entry.

Under the transformation in Definition 5.5.1, the vector fields $\dot{A}'(t) = F(A')$ and $\dot{B}'(t) = F(B')$ are given by:

$$\begin{aligned}\dot{A}'(t) &= \Phi(\sigma(t)) - (v^\dagger \Phi(\sigma(t))v)\mathbb{1} \\ \dot{B}'(t) &= -\Phi^\dagger(\rho(t)) + (v^\dagger \Phi^\dagger(\rho(t))v)\mathbb{1}\end{aligned} \tag{5.23}$$

where $\frac{d}{dt}\left(v^\dagger A(t)v\right)$ is given by $v^\dagger \frac{dA(t)}{dt}v$.

Moreover, the values of $\rho'(t)$ and $\sigma'(t)$ are defined as:

$$\begin{aligned}\rho'(t) &= \exp\big(A'(t)\big)/\mathrm{Tr}(\exp\big(A'(t)\big)) \\ \sigma'(t) &= \exp\big(B'(t)\big)/\mathrm{Tr}(\exp\big(B'(t)\big))\end{aligned} \tag{5.24}$$

Proposition 5.5.1. *The dynamics of $\rho(t)$ and $\sigma(t)$ remain the same after undergoing the canonical transformation. Equivalently, $A'(t)$ and $A(t)$ (resp. $B'(t)$ and $B(t)$) admit the same strategy $\rho(t)$ (resp. $\sigma(t)$).*

Proof. We first consider the definition of $\rho'(t)$.

$$\begin{aligned}\rho'(t) &= \frac{\exp(A'(t))}{\mathrm{Tr}(\exp(A'(t)))} \\ &= \frac{\exp\left(\int_0^t \Phi(\sigma(\tau))d\tau - (v^\dagger A(t)v)\mathbb{1}\right)}{\mathrm{Tr}\left(\exp\left(\int_0^t \Phi(\sigma(\tau))d\tau - (v^\dagger A(t)v)\mathbb{1}\right)\right)} \\ &= \frac{\exp\left(\int_0^t \Phi(\sigma(\tau))d\tau\right) \cdot \exp\left(-(v^\dagger A(t)v)\mathbb{1}\right)}{\mathrm{Tr}\left(\exp\left(\int_0^t \Phi(\sigma(\tau))d\tau\right) \cdot \exp\left(-(v^\dagger A(t)v)\mathbb{1}\right)\right)}\end{aligned}$$

Note that the denominator can be written as the trace of a matrix where each diagonal entry is the corresponding value of A multiplied by $\exp\big(-v^\dagger A(t)v\big)$. Thus, the above can be rewritten as

$$\begin{aligned}\rho'(t) &= \frac{\exp(A(t)) \cdot \exp\left(-(v^\dagger A(t)v)\mathbb{1}\right)}{\exp(-v^\dagger A(t)v) \cdot \mathrm{Tr}\left(\exp(A(t))\right)} \\ &= \frac{\exp(A(t))}{\mathrm{Tr}(\exp(A(t)))} \cdot \mathbb{1} \\ &= \rho(t)\end{aligned}$$

We can perform similar computations to show the same holds true for $\sigma'(t)$ and $\sigma(t)$. □

Proposition 5.5.2. *The mappings $A'(t)$ and $\rho(t)$ (resp. $B'(t)$ and $\sigma(t)$) are diffeomorphic to one another.*

Proof. Consider the map f between A' and ρ. We have shown in Proposition 5.5.1 that the map in one direction is $\rho = f(A') = \frac{\exp(A')}{\mathrm{Tr}(\exp(A'))}$. This is continuously differentiable. Now let us consider the inverse map f^{-1}. In particular, we show that the inverse mapping exists and is equal to $A' = f^{-1}(\rho) = \log(\rho) - (v^\dagger \log(\rho)v)\mathbb{1}$.

Indeed,

$$\begin{aligned} f(f^{-1}(\rho)) &= \frac{\exp\big(\log(\rho) - (v^\dagger \log(\rho) v)\mathbb{1}\big)}{\mathrm{Tr}(\exp(\log(\rho) - (v^\dagger \log(\rho) v)\mathbb{1}))} \\ &= \frac{\exp\left(\log(\rho)\right) \cdot \exp\left(-(v^\dagger \log(\rho) v)\mathbb{1}\right)}{\mathrm{Tr}\left(\rho \cdot \exp\left(-(v^\dagger \log(\rho) v)\mathbb{1}\right)\right)} \\ &= \frac{\rho \cdot \exp\left(-(v^\dagger \log(\rho) v)\mathbb{1}\right)}{\exp\left(-(v^\dagger \log(\rho) v)\right)} \\ &= \rho \cdot \mathbb{1} = \rho \end{aligned}$$

where we have used the fact that $\mathrm{Tr}(\rho) = 1$. Furthermore, note that the inverse mapping $f^{-1}(\rho) = \log(\rho) - (v^\dagger \log(\rho) v)\mathbb{1}$ is smooth.

Hence, since f is a bijection and the inverse map is differentiable, $A'(t)$ and $\rho(t)$ are diffeomorphic to each other. Similarly, $B'(t)$ and $\sigma(t)$ are diffeomorphic to each other. □

Proposition 5.5.2 will be of crucial importance to our proof technique, since we first prove recurrence for the system described by $A'(t)$ and $B'(t)$, then recover recurrence in $\rho(t)$ and $\sigma(t)$.

To show Poincaré recurrence of (MRD), we first prove two key properties: (i) the flow of $\dot{A}'$ is volume preserving, meaning that the trace of the Jacobian of the respective vector fields $\dot{A}'(t) = F(A')$ and $\dot{B}'(t) = F(B')$ are zero, and (ii) A' and B' have bounded orbits from any interior initial condition. Then, Poincaré recurrence of A' and B' follows from Poincaré's theorem.

5.5.1 Volume Preservation

We introduce a lemma which shows that in two-player zero-sum matrix replicator dynamics, the canonical transformation produces a dynamical system which preserves volume.

Lemma 5.5.1. *For two-player zero-sum matrix replicator dynamics, the vector fields that arise as a result of the canonical transformation in Definition 5.5.1 are volume preserving.*

Proof. Let $\Psi_{A'}$ denote the flow of $A'(t)$ and $\Psi_{B'}$ denote the flow of $B'(t)$. First let us consider $A'(t)$.

$$\begin{aligned} \dot{A}' = \frac{dA'}{dt} &= \Phi(\sigma(t)) - (v^\dagger \Phi(\sigma(t)) v)\mathbb{1} \\ &= \Phi\left(\frac{\exp(B(t))}{\mathrm{Tr}(\exp(B(t)))}\right) - \left(v^\dagger \Phi\left(\frac{\exp(B(t))}{\mathrm{Tr}(\exp(B(t)))}\right) v\right)\mathbb{1} \end{aligned}$$

Now, we derive the first order partial-derivatives of $F(A')$ with respect to $A'(t)$:

$$\begin{aligned}\frac{\partial F(A')}{\partial A'} &= \frac{\partial}{\partial A'}\left(\Phi\left(\frac{\exp(B(t))}{\mathrm{Tr}(\exp(B(t)))}\right) - \left(v^\dagger \Phi\left(\frac{\exp(B(t))}{\mathrm{Tr}(\exp(B(t)))}\right) v\right)\mathbb{1}\right) \\ &= \frac{\partial}{\partial A'}\left(\Phi\left(\frac{\exp(B'(t))}{\mathrm{Tr}(\exp(B'(t)))}\right) - \left(v^\dagger \Phi\left(\frac{\exp(B'(t))}{\mathrm{Tr}(\exp(B'(t)))}\right) v\right)\mathbb{1}\right) \\ &= 0\end{aligned}$$

Clearly, the partial derivative $\frac{\partial F(A')}{\partial A'}$ only depends on the value of $B'(t)$ and not $A'(t)$. This implies that the vector field is separable, and so the first order partial derivative with respect to $A'(t)$ is zero. By a similar argument, the partial derivative $\frac{\partial F(B')}{\partial B'}$ is also zero. By definition, the diagonal of the Jacobian matrix describing the vector fields is zero, and hence the divergence (which is the trace of the Jacobian) is zero as well. We can then directly apply Liouville's theorem to conclude that the flows $\Psi_{A'}$ and $\Psi_{B'}$ are volume preserving. □

5.5.2 Bounded Orbits

We now show that the transformed dynamical system always has bounded orbits when initialized on the interior of the space of probability density matrices.

Lemma 5.5.2. *For any finite initial points $A(0)$ and $B(0)$, the dynamics mapped to $A'(t)$ and $B'(t)$ via the transformation in Definition 5.5.1 have bounded orbits.*

Proof. First, $\rho(0) = \frac{\exp(A(0))}{\mathrm{Tr}(\exp(A(0)))}$ (resp. $\sigma(0)$) is bounded away from the boundary, since we assumed that since $A(0)$ and $B(0)$ are finite matrices. Moreover, $A'(0)$ and $B'(0)$ are also finite. The sum of quantum relative entropies at time 0 is thus:

$$S(\rho^*\|\rho(0)) + S(\sigma^*\|\sigma(0)) = \mathrm{Tr}(\rho^*(\log\rho^* - \log\rho(0))) + \mathrm{Tr}(\sigma^*(\log\sigma^* - \log\sigma(0)))$$

This sum is finite since $\rho(0), \sigma(0)$ have full support. Indeed, the support of ρ^* and σ^* are contained in the support of $\rho(0)$ and $\sigma(0)$ respectively. By Theorem 5.4.1, the sum of quantum relative entropies $S(\rho^*\|\rho(t)) + S(\sigma^*\|\sigma(t))$ is bounded above by a fixed value C for all time t. Now, let $\lambda_{\min}(\rho^*)$ denote the minimum eigenvalue of ρ^*. We have that $\rho^* \geq \lambda_{\min}(\rho^*)\mathbb{1}$. Note the following:

$$\begin{aligned}\mathrm{Tr}(\rho^*(\log\rho^*)) - \mathrm{Tr}(\rho^*(\log\rho(t))) &< C \\ -\mathrm{Tr}(\rho^*(\log\rho(t))) &< D \\ -\mathrm{Tr}(\lambda_{\min}(\rho^*)\mathbb{1}(\log\rho(t))) &< D\end{aligned}$$

where C and D are positive and finite real numbers. Using the fact that the trace of a matrix is basis independent, we use a basis where $\rho(t)$ is diagonal. Here, the minimum eigenvalue of $\rho(t)$ cannot go to zero, otherwise $-\mathrm{Tr}(\lambda_{\min}(\rho^*)\mathbb{1}(\log\rho(t)))$ goes to $+\infty$, a clear contradiction to the inequality above. A similar argument holds for $\sigma(t)$. This is equivalent to saying that all the eigenvalues of $\rho(t)$ and $\sigma(t)$ lie in the interval $[\epsilon, 1-\epsilon]$ for some small $\epsilon > 0$.

Note that the value of $A'_{1,1}(t)$ is always zero by design. We also know by construction that $A'(t)$ is Hermitian, since $\sigma(t)$ is Hermitian, Φ constitutes a hermicity preserving map and the integral of Hermitian matrices remains Hermitian. Hence, the smallest eigenvalue of $A'(t)$, $\lambda_{\min}(A'(t))$ is upper bounded by the smallest element on the diagonal of $A'(t)$, namely $\lambda_{\min}(A'(t)) \leq 0$, since $A'_{1,1}(t) = 0$. If the largest eigenvalue $\lambda_{\max}(A'(t))$ goes to $+\infty$, then the corresponding smallest eigenvalue of $\rho(t) = \frac{\exp(A'(t))}{\mathrm{Tr}(\exp(A'(t)))}$ goes to 0. Likewise if the smallest eigenvalue $\lambda_{\min}(A'(t))$ goes to $-\infty$, the smallest eigenvalue of $\rho(t)$ goes to 0 as well. We have previously shown that this cannot occur, since the eigenvalues of $\rho(t)$ are all bounded away from zero. Hence, it follows that all eigenvalues of $A'(t)$ are bounded away from infinity.

Now, consider the scalar $v^\dagger A'(t)w$ for arbitrary bounded vectors v and w. $A'(t)$ is Hermitian, so by the spectral theorem it has an orthonormal eigenbasis. Now we express v and w in terms of this n-dimensional eigenbasis (which we denote by $\mathbb{e}$):

$$v = \sum_{i=1}^{n} a_i \mathbb{e}_i \quad \text{and} \quad w = \sum_{i=1}^{n} b_i \mathbb{e}_i$$

where a_i and b_i are scalars. Thus,

$$\begin{aligned} v^\dagger A'(t) w &= \left(\sum_{i=1}^{n} a_i \mathbb{e}_i \right)^\dagger A'(t) \left(\sum_{i=1}^{n} b_i \mathbb{e}_i \right) \\ &= \sum_{i=1}^{n} a_i b_i \lambda_i \end{aligned}$$

where λ_i are the eigenvalues of $A'(t)$ corresponding to $\mathbb{e}_i$. Here a_i, b_i and λ_i are all bounded, so it follows that all elements of matrix $A'(t)$ are bounded away from infinity as well. Likewise, the elements of matrix $B'(t)$ remain bounded from infinity and thus the dynamical system described by $A'(t)$ and $B'(t)$ have bounded orbits.

□

Finally, we are ready to prove Theorem 5.5.1 using Lemmas 5.5.1 (volume preservation) and 5.5.2 (bounded orbits).

Proof of Theorem 5.5.1. By Lemmas 5.5.1 and 5.5.2, as well as the Poincaré recurrence theorem introduced in Section 5.2, we immediately see that the system of replicator equations given by dA'/dt and dB'/dt are Poincaré recurrent since they are volume preserving and have bounded orbits. Since the flows of $A'(t)$ and $\rho(t)$ are diffeomorphic to one another (likewise for $B'(t)$ and $\sigma(t)$), $d\rho/dt$ and $d\sigma/dt$ are also Poincaré recurrent. This concludes the proof. □

5.6 Simulations

To corroborate the theoretical results presented in prior sections, we performed relevant simulations of quantum games using both discrete `MMWU` and matrix replicator dynamics.

In the rest of this section, we standardize the use of quantum zero-sum games obtained via basis transform. In particular, in order to formalize a method for simulating and understanding quantum games, we propose a method to generate complex payoff observables R with the same eigenvalues as a classical payoff matrix. This means that for any standard eigenbasis of a classical game, we create a new, complex eigenbasis where the eigenvalues are the same, but the eigenvectors are complex. Intuitively, this can be viewed as mapping a classical game matrix to the Hilbert space, effectively allowing generalizations of classically studied games to the semi-definite context. We introduce this methodology in Lemma 5.6.1, which describes a basis transformation to the Hilbert space.

Lemma 5.6.1. *For any $d \times d$ two player zero-sum game with classical payoff matrix P, let $\lambda_{i,j} = P_{i,j}$ and let V and W be unitary matrices of size $d \times d$. Then, the matrix R defined as*

$$R = \sum_{i,j} \lambda_{i,j}(V_i \otimes W_j)(V_i^\dagger \otimes W_j^\dagger) \tag{5.25}$$

produces a transformation from the classical payoff space to the d^2-dimensional Hilbert space. In particular, R is a complex $d^2 \times d^2$ matrix which satisfies the following properties:

- *The matrix R is Hermitian.*
- *The eigenvalues of R correspond to the elements of the classical payoff matrix P.*

Proof. First note that Equation 5.25 is equivalent to taking the following basis transformation:

$$\sum_{i,j} \lambda_{i,j}(V_i \otimes W_j)(V_i^\dagger \otimes W_j^\dagger) = (V \otimes W)\Lambda(V^\dagger \otimes W^\dagger)$$

where $\Lambda = \begin{bmatrix} \lambda_{1,1} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \lambda_{n,n} \end{bmatrix}$ is the $n^2 \times n^2$ diagonal matrix formed by placing the elements of classical payoff matrix P on the diagonal. Moreover, V and W are both unitary, so the tensor products $V \otimes W$ and $V^\dagger \otimes W^\dagger$ are also unitary. This holds because:

$$\begin{aligned}(V \otimes W)(V \otimes W)^\dagger &= (V \otimes W)(V^\dagger \otimes W^\dagger) \\ &= VV^\dagger \otimes WW^\dagger \\ &= \mathbb{1} \otimes \mathbb{1} = \mathbb{1}\end{aligned}$$

This implies that R can be written as a unitary diagonalization of the form $PDP^\dagger$, where $P = (V \otimes W)$ and $D = \Lambda$. Hence R is Hermitian and has real eigenvalues by the spectral theorem. In addition, by construction the eigenvalues of R are the elements of the classical payoff matrix P. □

Moreover, the initial conditions of a simulation of a two-player quantum zero-sum game can be similarly obtained by using the same orthonormal bases as in Lemma 5.6.1 and applying a change of basis to the initial conditions in the standard basis.

5.6.1 MMWU Simulations

First, we show the trajectories of the first eigenvalue of each player in a quantum Matching Pennies game, obtained using the discrete MMWU algorithm. To generate Figure 5.1, we used a modified version of Algorithm 1, where instead of constant step-size μ we define $\mu = \log\left(1 + \frac{1}{t^a}\right)$, where a is an exponent which determines the rate of decrease of the step-size, and t is the time-step of the simulation. Intuitively, this means that for higher values of a, the step-sizes go to zero at a faster rate. Empirically, we observe that if the exponent is greater than $\frac{1}{2}$, then we can clearly see the phenomenon described in Theorem 5.3.1, since the quantum relative entropy in the system increases when the trajectories move away from the fully mixed Nash equilibrium.

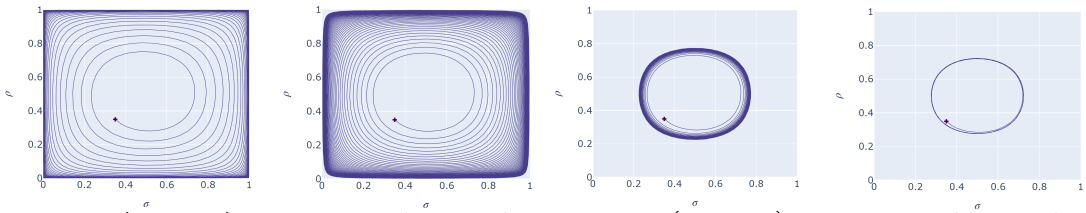


FIGURE 5.1: Trajectories of players using MMWU in a quantum Matching Pennies game with decreasing μ values.

5.6.2 MRD Simulations

In the case of matrix replicator dynamics, we show experimentally (Figure 5.2) that the sum of quantum relative entropies for both players is a constant of motion as shown in Theorem 5.4.1. We first use Lemma 5.6.1 to obtain the R matrix of a quantum game given some complex basis and classical payoff matrix (in this case, Matching Pennies). Then, we run the discrete-time MMWU update with decreasing step-size to compute the interior ϵ-approximate Nash of the quantum game. The replicator equations are then solved given a set of initial conditions for each player. Thereafter, we compute the quantum relative entropy between each player's strategy at each discretized time step and the ϵ-approximate Nash equilibrium.

Notice that although the quantum relative entropies of the first player (pink) and second player (purple) are oscillating over time, their sum remains approximately constant.

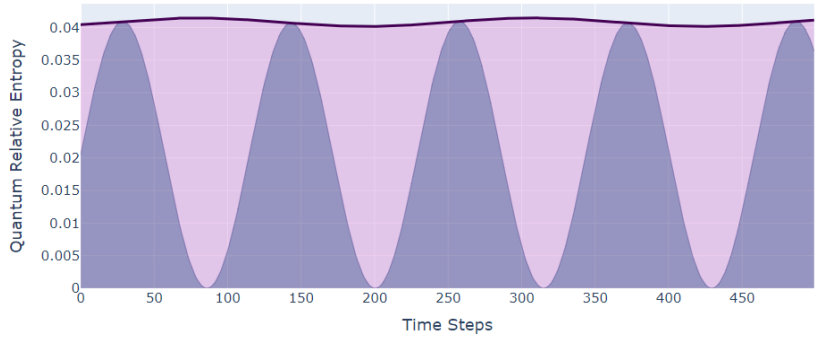


FIGURE 5.2: Approximately constant sum of quantum relative entropy values for players using MMWU in a quantum Matching Pennies game.

Finally, in order to generate Figure 5.3, we perform a similar simulation to the above. To show the dynamical behavior of the system, we utilize the Bloch sphere representation, which we now briefly describe. The *Pauli matrices* are given by

$$\sigma_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \ \sigma_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},$$
$$\sigma_2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

The (non-identity) Pauli matrices are Hermitian, their trace is equal to zero, they have ± 1 eigenvalues, pairwise anti-commute and form an orthogonal basis of the 2×2 complex matrices. Next, we show that any 2×2 density operator ρ can be represented as a real 3-dimensional vector with Euclidean norm at most 1, where vectors with norm equal to 1 correspond to pure states. These 3-dimensional vectors can be visualized as points on a sphere, known as the Bloch sphere. Specifically, we have that ρ can be written as

$$\rho = \frac{1}{2}(I_2 + \sum_{i=1}^{3} a_i \sigma_3), \text{ where } \|a\|_2 \leq 1,$$

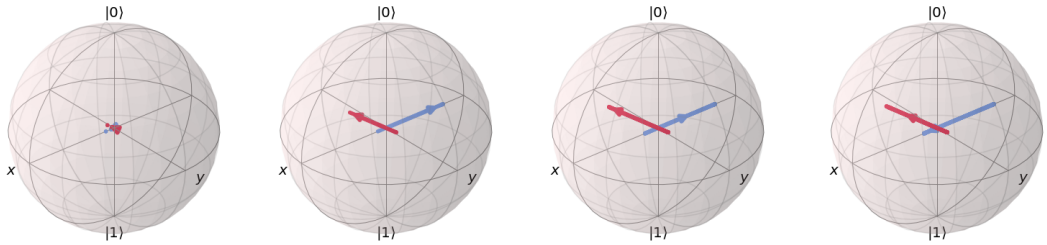
and moreover, ρ is pure iff $\|a\|_2 = 1$. To see this, we expand ρ in the Pauli basis, i.e., $\rho = \sum_{i=0}^{3} a_i \sigma_i$, where $a_i = \operatorname{Tr}(\rho\sigma_i)/\operatorname{Tr}(\sigma_i^2)$. Recalling that $\sigma_i^2 = I_2$ we finally get that

$$\rho = \frac{1}{2}(I + \sum_{i=1}^{3} a_i \sigma_3), \text{ where } a_i = \operatorname{Tr}(\rho\sigma_i).$$

Lastly, note that all a_i's are real (as the inner product of Hermitian matrices) and the eigenvalues of $\sum_{i=1}^{3} a_i \sigma_3$ are $\pm\|a\|$. Thus, for ρ to be PSD we need that $\|a\| \leq 1$. Lastly, ρ is pure iff $\operatorname{Tr}(\rho^2) = 1$, which is equivalent to $\|a\|_2 = 1$.

Utilizing the above representation, we present Bloch sphere visualizations of the trajectories of MRD in a quantum Matching Pennies game. In particular, the density matrix representing the strategy of each player at each time-step is given as a point within the sphere, and we plot the movement of these orbits over time. According to Theorems 5.4.1 and 5.5.1, we expect the trajectories of the replicator dynamics to stay on the interior

of the Bloch sphere, since the surface of the sphere corresponds to the pure states of the system. We see from Figure 5.3 that over time, the system never reaches the boundary of the sphere, which experimentally agrees with our results.



$t = 1$ $t = 50$ $t = 370$ $t = 1000$

FIGURE 5.3: Bloch sphere trajectories for quantum Matching Pennies game between Alice (blue) and Bob (red). The arrowheads represent the current state of each player at each time-step. Notice that over time, the orbits oscillate within the interior of the Bloch sphere, implying recurrent behavior.

5.6.3 Larger-Scale Experiments

In prior simulations, we studied a single-qubit system in the form of a Matching Pennies game. However, our results extend to games with larger strategy sets, some of which can be interpreted as multi-qubit systems. We study a 16×16 game with a payoff matrix with values in $[-1, 1]$ which can be played via 2 qubits. Moreoever, this game is selected because it has an interior (and uniform) Nash equilibrium. Similarly, we generate an 64×64 generalization of the 16×16 game to represent an example of a 3-qubit game with interior Nash. In Figures 5.4 and 5.5, we plot the Frobenius norm between the initial condition and the system state when updated using matrix replicator dynamics ((MRD)). Notice that despite the relatively erratic behaviour of the strategies, they eventually return to the initial condition, implying that recurrence holds (Theorem 5.5.1).

5.7 Conclusion

In this chapter, we departed from the setting of time-evolving games to focus on the interplay between continuous- and discrete-time learning dynamics in games. In particular, whilst Matrix Multiplicative Weights Update (MMWU) has been a well-studied algorithm in the literature, its continuous-time counterpart has not been studied from the perspective of learning in games.

We studied the quantum information theoretic properties of MMWU and its continuous analogue matrix replicator dynamics (MRD) in the context of two-player zero-sum quantum games. First, we provide a formulation of matrix replicator dynamics (MRD) which arises from taking a continuous-time limit of MMWU. Then, we show that the total quantum relative entropy within such a system is a constant of motion. Finally, we show that in matrix replicator systems with interior Nash equilibria, the dynamics exhibit Poincaré recurrence. This work constitutes an initial step towards analyzing learning

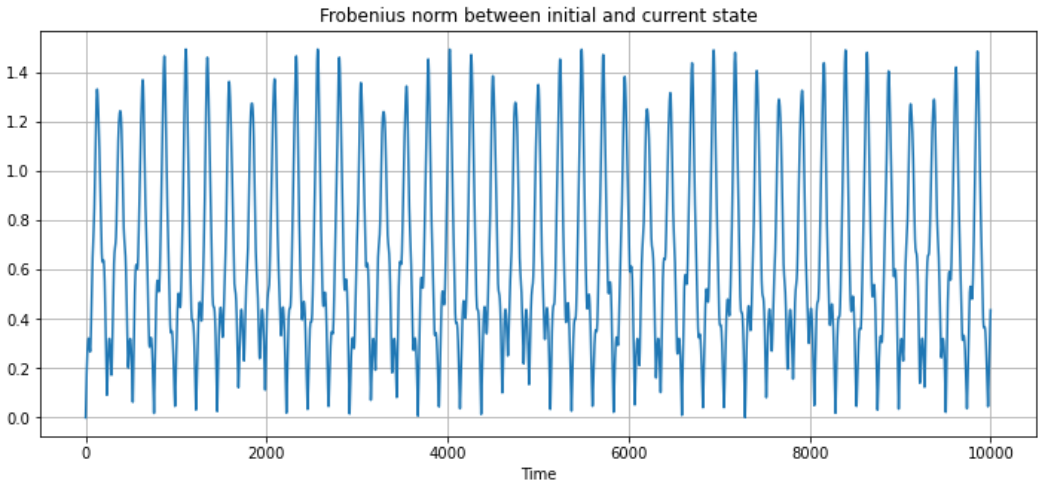


FIGURE 5.4: Frobenius norm between the initial condition and current system state for 2 qubit system with interior Nash equilibrium. Note that at approx. 7200 iterations the system returns arbitrarily close to the initial condition.

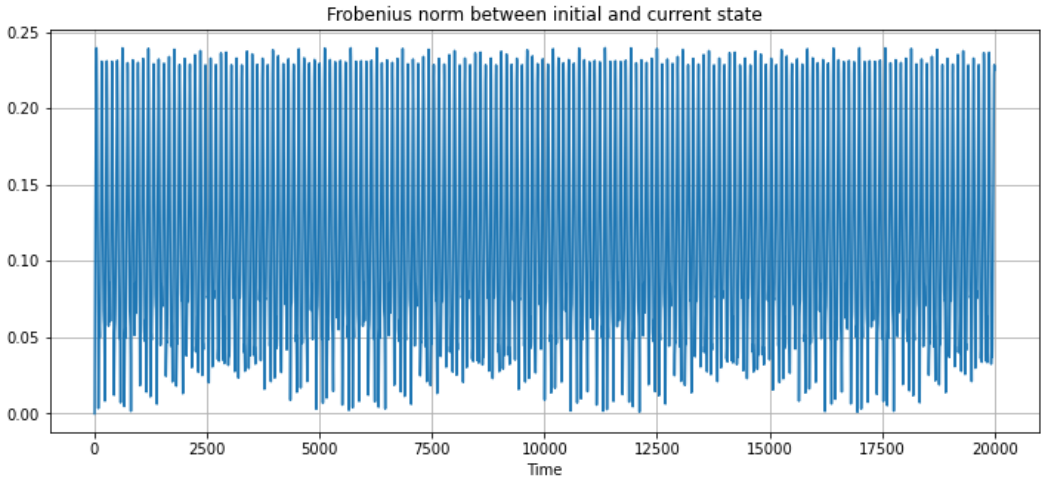


FIGURE 5.5: Frobenius norm between the initial condition and current system state for 3 qubit system with interior Nash equilibrium. Note that at approx. 12000 iterations the 3-qubit system returns arbitrarily close to the initial condition.

behaviour in games with quantum information. In the classical world, showing that conservation laws and recurrence holds has led to a better understanding of game dynamics in increasingly complex settings [117, 127, 135]. Similar work has now been on the rise in the realm of quantum games [108, 109, 111, 112, 171].

Part II

Discrete-Time Dynamics: Optimism and Clairvoyance

It is far better to foresee even without certainty than not to foresee at all.

Henri Poincaré

In the remainder of this dissertation, we switch our focus to discrete-time dynamics. In many multi-agent settings, a key question is whether one can design algorithms that guarantee convergence to equilibria, either in time-average or in the last-iterate sense. Moreover, the rate of convergence of these dynamics is an integral aspect of the analysis.

In Chapter 6, we study a class of multiplayer extensive-form games with a network structure, which we call network zero-sum extensive-form games. For these games, we utilize *Optimistic* gradient ascent to establish fast time-average and last-iterate convergence to Nash equilibria. Finally, in Chapter 7, we introduce a variant of `MWU` for learning in general-sum games called Clairvoyant `MWU` (`CMWU`), where players select their strategies based on everyone's strategies in the next time-step (i.e., the players are clairvoyant). Even though implementing `CMWU` is computationally difficult, we introduce an uncoupled learning dynamic based on `CMWU` which achieves a new state-of-the-art rate of convergence to coarse correlated equilibria in general normal-form games.

Chapter 6

Optimism: Fast Convergence to Nash in Network Zero-Sum Extensive-Form Games

This chapter is taken from (with minor modifications) our paper *'Fast Convergence of Optimistic Gradient Ascent in Network Zero-Sum Extensive Form Games'* [159].

6.1 Introduction

In Chapters 3 and 4, we studied the *dynamical properties* of continuous-time dynamics in time-evolving zero-sum game variants. While these intrinsic modifications of the games studied were certainly interesting, there is another element of the formulation which demands further investigation. In the realm of normal-form zero-sum games, [31] showed that (multiplayer) polymatrix/network zero-sum games exhibit the same Nash convergence properties that two-player zero-sum games enjoy. This is a particularly fascinating result, because it implies that there exist multiplayer games with a certain structure for which one can prove convergence to Nash equilibria using simple discrete-time online update rules.

Thus far, that line of inquiry has (quite reasonably) focused on polymatrix *normal-form games*. However, learning in games beyond the normal-form regime is also imperative. Many of the success stories in game theory comes from *extensive-form games* (EFGs), an important class of games which have been studied for more than 50 years [98]. EFGs capture various settings where several selfish agents sequentially perform actions which change the *state of nature*, with the action-sequence finally leading to a *terminal state*, and the players then receive a payoff. The most ubiquitous examples of EFGs are real-life games such as Chess, Poker, Go etc. Indeed, this area is of key interest for the *online learning in games* framework, even leading to modern algorithms which can defeat the best human players in real-life games [28, 165]. In addition, online learning in EFGs has other applications in economics, machine learning and sequential decision-making that extend far beyond the design of game-solvers [4, 135].

Despite its numerous applications, online learning in EFGs, particularly multiplayer EFGs, is far from well understood. From a practical point of view, testing and experimenting with various online learning algorithms in EFGs often requires massive amounts of computational resources due to the large number of states in EFGs of interest [146, 188]. Even from a theoretical perspective, many complications may arise. As mentioned in prior chapters, online learning dynamics may oscillate, cycle or even admit chaotic behavior even in simple settings [106, 117, 134]. In essence, solving extensive-form games is difficult and we cannot assume much, if anything, about the convergence properties of online learning in this setting.

This is not to say that all hope is lost in the realm of EFGs. Much like in normal-form games, two-player EFGs turn out to be quite a bit more well-behaved than their multiplayer counterparts. [188] and [101] proposed no-regret algorithms for extensive-form games with $\mathcal{O}(1/\sqrt{T})$ average regret and polynomial running time in the size of the game. More recently, regret-based algorithms achieved $\mathcal{O}(1/T)$ time-average convergence to the min-max equilibrium [59, 82, 96] for two-player zero-sum EFGs. Finally, recent work has shown that in two-player zero-sum EFGs, the time-average strategy vector produced by online learning dynamics converges to the Nash Equilibrium (NE), while there exist dynamics that exhibit last-iterate convergence [59, 104].

In particular, while standard online learning dynamics such as `MWU` and `GDA` fail to converge to the Nash equilibrium in the last-iterate sense, a modification known as 'optimism' can actually allow the day-to-day behavior of these dynamics to converge to the Nash. In the game theoretic context, optimistic learning dynamics exploit the fact that both players are using the same learning dynamic and take into account the players' payoffs in the previous timestep. Essentially, the payoff gradient of the previous timestep is used as a predictor for the next timestep's payoff gradient. This idea has been used in various settings [47, 142, 177, 178], and in two-player EFGs an optimistic variant of the standard Mirror Descent framework has seen success in recent years, showing fast convergence to Nash equilibria in the last-iterate sense [59, 104].

Considering that many interesting real-world settings consist of multiple agents, the above line of research then motivates a few natural questions:

> *Are there natural classes of multi-agent extensive-form games for which online learning dynamics converge to a Nash Equilibrium? Furthermore, what type of convergence is possible? Can we only guarantee time-average convergence, or can we also prove last-iterate convergence of the dynamics via optimism?*

Throughout this chapter, we will answer the above questions in the positive for an interesting class of multiplayer EFGs called *Network Zero-Sum Extensive-Form Games*. Reminiscent of the definitions of polymatrix games from Chapters 3 and 4, a Network EFG consists of a graph $\mathcal{G} = (V, E)$ where each vertex $u \in V$ represents a utility-maximizing player and each edge $(u, v) \in E$ corresponds to an extensive-form game Γ^{uv} played between the players $u, v \in V$. Each player $u \in V$ selects their strategy so as to maximize the overall payoff from the games corresponding to their incident edges. The game is additionally called *zero-sum* if the sum of the players' payoffs is equal to zero no matter the selected strategies. In essence, this formulation is precisely that of [32, 51], but instead of normal-form games on the edges of the graph, we consider EFGs. Moreover,

note that we call these games *network* games and not *polymatrix* games, because we study EFGs instead of bimatrix games on the edges of the graph.

Specifically, we analyze the convergence properties of the online learning dynamics produced when all players of a Network Zero-Sum EFG update their strategies according to *Optimistic Gradient Ascent*, and show the following result:

Informal Theorem. *When the players in a network zero-sum extensive-form game update their strategies using Optimistic Gradient Ascent (*OGA*), their time-average strategies converge at rate $\mathcal{O}(1/T)$ to a Nash Equilibrium, while the last-iterate mixed strategies converge to a Nash Equilibrium at rate $\mathcal{O}(c^{-t})$ for some game-dependent constant $c > 0$.*

Network Zero-Sum EFGs are an interesting class of multi-agent EFGs for much the same reasons that network zero-sum normal form games are interesting, with several additional challenges. Indeed, due to the prevalence of networks in computing systems, there has been increased interest in network formulations of normal form games [92], which have been applied to multi-agent reinforcement learning [184] and social networks [93].

Network Zero-Sum EFGs can be seen as a natural model of closed systems in which selfish agents compete over a fixed set of resources [32, 51], thanks to their global constant-sum property[1] (the edge games do not necessarily have to be zero-sum). As an example of this model, consider the users of an online poker platform playing *Heads-up Poker*, which is a two-player EFG. Each user can be thought of as a node in a graph and two users are connected by an edge (corresponding to a poker game) if they play against each other. Note that in Heads-up Poker games, each player puts either a small or large number of chips (also known as blinds) into a common pot, and the game differs depending on which player has which blind. Hence, each edge/game could differ from another due to the differences in blinds. Each user u selects a strategy to play against the other players, with the goal of maximizing their overall payoff. This is an indicative example which can clearly be modeled as a Network Zero-Sum EFG.

In addition, Network Zero-Sum EFGs are also attractive to study because their descriptive complexity scales *polynomially* with the number of agents. Multiplayer EFGs that cannot be decomposed into pairwise interactions (i.e., do not have a network structure) admit an exponentially large description with respect to the number of the agents [92]. Hence, by considering this class of games, we are able to exploit the decomposition to extend results that are known for network normal form games to the extensive form setting.

6.1.1 Our Contributions

To the best of our knowledge, this is the first work establishing convergence to Nash Equilibria of online learning dynamics in network extensive-form games with more than two agents. As already mentioned, there has been a stream of recent works establishing the convergence to Nash Equilibria of online learning dynamics in two-player zero-sum EFGs. However, there are several key differences between the two-player and network cases. All previous works concerning the two-player case follow a *bilinear saddle point*

[1] this is equivalent to the global zero-sum property.

approach. Specifically, these works leverage the fact that any Nash Equilibrium of this class of games is a min-max equilibrium. This means that any NE coincides with the solutions of the following bilinear saddle-point problem:

$$\min_{x\in\mathcal{X}}\max_{y\in\mathcal{Y}} x^\top \cdot A \cdot y = \max_{y\in\mathcal{Y}}\min_{x\in\mathcal{X}} x^\top \cdot A \cdot y$$

Any online learning dynamic that converges to the solution of the above saddle-point problem thus also converges to the Nash equilibrium in the two-player setting.

However, in the network setting, we lose the safe haven of min-max optimization. The games no longer have min-max equilibria and hence there is no immediate connection between Nash Equilibria of the game and saddle-point optimization. To overcome this difficulty, we establish that Optimistic Gradient Ascent (OGA) in a class of EFGs we call *consistent* Network Zero-Sum EFGs (see Section 6.3) can be equivalently described as Optimistic Gradient Descent in a *two-player symmetric game* (R, R) over a *treeplex polytope* $\mathcal{X}$. Note the difference in update rule – gradient ascent over the original game is thus shown to be equivalent to gradient descent in a new, constructed game.

To be clear, both the matrix R that constitutes the new symmetric game and the treeplex polytope $\mathcal{X}$ are constructed from the original Network Zero-Sum EFG. Using the zero-sum property of Network EFGs, we show that the constructed matrix R satisfies a 'restricted' zero-sum property, in the following sense:

$$x^\top \cdot R \cdot y + y^\top \cdot R \cdot x = 0 \text{ for all } x, y \in \mathcal{X} \tag{6.1}$$

Intuitively, one can think of Property (6.1) as a generalization of the *classical zero-sum property* i.e. $A = -A^\top$. In general, the matrix we construct, R, does not satisfy the classical zero-sum property $R = -R^\top$. Property (6.1) simply ensures that the sum of payoffs is equal to zero only when $x, y \in \mathcal{X}$. Our primary technical contribution consists of generalizing the analysis of [178] (which holds for classical two-player zero-sum games) to symmetric games which satisfy Property (6.1).

6.2 Preliminaries and Definitions

Due to their potentially complicated structure, a key part of studying extensive-form games is describing them mathematically while avoiding unwieldy notation. For the purposes of this chapter, we repeat the full description of two-player EFGs, as the notation becomes helpful for a thorough understanding of the network generalization which we derive. Moreover, our simulations are primarily based on an extensive-form version of Matching Pennies, and we also do our best to provide intuition and explanations for the various components of the definition.

6.2.1 Two-Player Extensive-Form Games

We will employ the indicative example of Matching Pennies throughout this section. In the first example, imagine an MP game where Player 1 selects their action first, Player 2 observes their action, then selects their action (Figure 6.1).

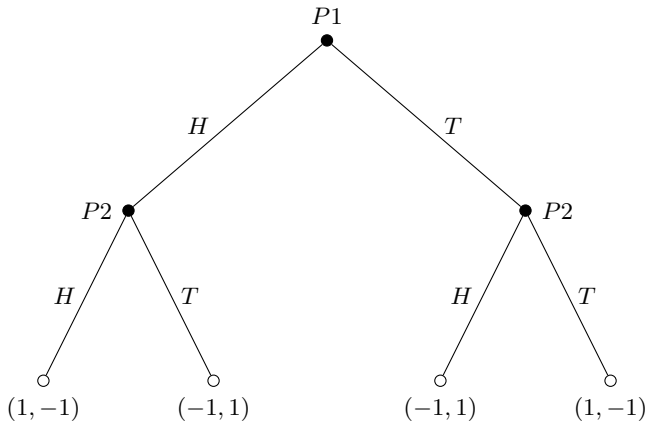


FIGURE 6.1: Matching Pennies game with perfect information.

Definition 6.2.1. *A two-player extensive-form game* Γ *is a tuple* $\Gamma := \langle \mathcal{H}, \mathcal{A}, \mathcal{Z}, p, I \rangle$ *which consists of:*

- *A finite set* $\mathcal{H}$ *of nodes which are decision points for the players. The states* $h \in \mathcal{H}$ *form a tree rooted at an initial state* $r \in \mathcal{H}$*.*
- *A set of available actions* $\mathcal{A}(h)$ *at each state* $h \in \mathcal{H}$*.*
- *A label* $\mathrm{Label}(h) \in \{1, 2, c\}$ *that denotes the acting player at each state* $h \in \mathcal{H}$*. Moreover, there is a special 'chance player' denoted by the letter* c*, which represents randomness in nature. Each state* $h \in \mathcal{H}$ *with* $\mathrm{Label}(h) = c$ *is additionally associated with a function* $\sigma_h : \mathcal{A}(h) \mapsto [0, 1]$ *where* $\sigma_h(\alpha)$ *denotes the probability that the chance player selects action* $\alpha \in \mathcal{A}(h)$ *at state* h*, and* $\sum_{\alpha \in \mathcal{A}(h)} \sigma_h(\alpha) = 1$*.*
- *A successor node* $\mathrm{Next}(\alpha, \mathrm{h}, \mathrm{i})$ *which denotes the state* $h' := \mathrm{Next}(\alpha, \mathrm{h}, \mathrm{i})$ *which is reached when player* $i := \mathrm{Label}(h)$ *takes action* $\alpha \in \mathcal{A}(h)$ *at state* h*. For clarity,* $\mathcal{H}_i \subseteq \mathcal{H}$ *denotes the states* $h \in \mathcal{H}$ *with* $\mathrm{Label}(h) = i$*.*
- *Terminal states* $\mathcal{Z}$ *corresponding to the leaves of the game tree. At each* $z \in \mathcal{Z}$ *no further action can be chosen, so* $\mathcal{A}(z) = \varnothing$*. Each terminal state* $z \in \mathcal{Z}$ *is associated with a payoff function* $p_i : \mathcal{Z} \to \mathbb{R}$*, which defines the payoff* $p_i(z)$ *received by player* i *upon reaching leaf node* z*.*
- *Information sets for each player*i *denoted by* $(I_1, \ldots, I_k)$*, where* $I(h)$ *is the information set of state* $h \in \mathcal{H}_i$*. If* $I(h_1) = I(h_2)$ *for some* $h_1, h_2 \in \mathcal{H}_1$*, then* $\mathcal{A}(h_1) = \mathcal{A}(h_2)$*.*

In the MP game shown in Figure 6.1, the decision points for the players are labeled with solid black circles. At each of these nodes, the label determines who acts. For instance, Player 1 starts the game and Player 2 follows in the second node. The action set at each node is $\{H, T\}$ – each player can choose either Heads or Tails at their respective decision point. Moreover, Player 1's choice determines $\mathrm{Next}(\alpha, h, 1)$, intuitively the node at which Player 2 has to play and its corresponding action set. Terminal/leaf nodes are denoted by empty circles, and payoffs $(p_1(z), p_2(z))$ are assigned to Players 1 and 2 respectively.

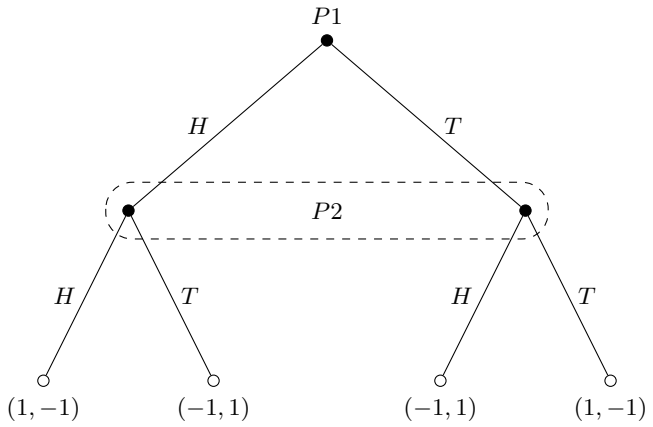


FIGURE 6.2: Matching Pennies game with imperfect information. Information set for Player 2 is denoted by dotted box.

The Case of Imperfect Information. There is one more element of Definition 6.2.1 which has not been fully explained. Consider a different implementation of the Matching Pennies game where Player 2 does not get to observe Player 1's action. To account for this lack of information, nodes are grouped together into what we call *information sets*. Nodes that belong to the same information set are effectively indistinguishable – the available actions at states h_1, h_2 in the same information set ($I(h_1) = I(h_2)$) must coincide, in particular $\mathcal{A}(h_1) = \mathcal{A}(h_2)$. Otherwise, Player 2 could very easily determine which state they are in by simply comparing their action sets! We present this example of an imperfect information game in Figure 6.2.

We also need a way of writing down the players' strategies in a two player EFG. We introduce a one such 'natural representation' known as a *behavioral plan*.

Definition 6.2.2. *A behavioral plan σ_i for player i is a function such that for each state $h \in \mathcal{H}_i$, $\sigma_i(h)$ is a probability distribution over $\mathcal{A}(h)$. In particular, $\sigma_i(h, \alpha)$ denotes the probability that player i takes action $\alpha \in \mathcal{A}(h)$ at state $h \in \mathcal{H}_i$. Furthermore, it is required that $\sigma_i(h_1) = \sigma_i(h_2)$ for each $h_1, h_2 \in \mathcal{H}_i$ with $I(h_1) = I(h_2)$. The set of all behavioral plans for player i is denoted by Σ_i.*

Due to imperfect information, the constraint $\sigma_i(h_1) = \sigma_i(h_2)$ for all $h_1, h_2 \in \mathcal{H}_i$ with $I(h_1) = I(h_2)$ models the fact that since player i cannot differentiate between states h_1, h_2, they must act in the exact same way at states $h_1, h_2 \in \mathcal{H}_i$.

Next, we turn our attention to the payoffs that each player receives. Intuitively, the expected payoff for any given player can be described with respect to the payoff obtained at each terminal state and the probability that each terminal state is reached. This idea can be expressed given the players' behavioral plans as follows:

Definition 6.2.3. *For a collection of behavioral plans $\sigma = (\sigma_1, \sigma_2) \in \Sigma_1 \times \Sigma_2$, the expected payoff of player i, denoted by $U_i(\sigma)$, is defined as:*

$$U_i(\sigma) := \sum_{z \in \mathcal{Z}} p_i(z) \cdot \underbrace{\Pi_{(h,h') \in \mathcal{P}(z)} \, \sigma_{\mathrm{Label(h)}}(h, \alpha_{h'})}_{\textit{probability that state } z \textit{ is reached}}$$

where $\mathcal{P}(z)$ denotes the path from the root state r to the terminal state z and $\alpha_{h'}$ denotes the action $\alpha \in \mathcal{H}_i$ such that $h' = \text{Next}(h, \alpha)$.

Next, we need a definition of what a Nash equilibrium looks like in this class of games.

Definition 6.2.4. *A collection of behavioral plans $\sigma^* = (\sigma_1^*, \sigma_2^*)$ is called a Nash Equilibrium if for all players $i = \{1, 2\}$,*

$$U_i(\sigma_i^*, \sigma_{-i}^*) \geq U_i(\sigma_i, \sigma_{-i}^*) \quad \textit{for all } \sigma_i \in \Sigma_i$$

As we have seen, the classical result of [128] proves the existence of Nash Equilibrium in normal form games. This result also generalizes to a wide class of extensive-form games which satisfy a property called *perfect recall* [97, 152], which intuitively means that players never forget their past actions and observations.

Consider two information sets $I(h_1)$ and $I(h_2)$. If $I(h_1) = I(h_2)$. By definition, player i cannot differentiate between states $h_1, h_2 \in \mathcal{H}_i$. Moreover, define the sets $\mathcal{P}(h_1) \cap \mathcal{H}_i := (p_1, \ldots, p_k, h_1)$ and $\mathcal{P}(h_2) \cap \mathcal{H}_i := (q_1, \ldots, q_m, h_2)$, where we overload notation slightly and use p_ℓ and q_ℓ to denote the states along paths $\mathcal{P}(h_1)$ and $\mathcal{P}(h_2)$ respectively. In order for state h_1 to be reached in gameplay, player i must take a specific sequence of actions along the path $\mathcal{P}(h_1) \cap \mathcal{H}_i := (p_1, \ldots, p_k, h_1)$. The same logic holds for $\mathcal{P}(h_2) \cap \mathcal{H}_i := (q_1, \ldots, q_m, h_2)$. Now, if player i could distinguish $\mathcal{P}(h_1) \cap \mathcal{H}_i$ from $\mathcal{P}(h_2) \cap \mathcal{H}_i$ (i.e., if the paths that led to their current state are different), then they could immediately distinguish state h_1 from h_2 by recalling the previous states in $\mathcal{H}_i$. This means that the number of nodes in both paths has to be the same, and the information sets of each state in both paths also have to be the same. However, even if $I(p_\ell) = I(q_\ell)$ for all $\ell \in \{1, \ldots, k\}$, player i could still distinguish h_1 from h_2 if their successors are different, i.e. if $p_{\ell+1} \in \text{Next}(p_\ell, \alpha, i)$ and $q_{\ell+1} \in \text{Next}(q_\ell, \alpha', i)$ are ever different. In such a case, player i can distinguish h_1 from h_2 by recalling the actions that they previously played, and checking if the ℓ-th action was α or α'. These constraints to ensure perfect recall are summarized in Definition 6.2.5.

Definition 6.2.5. *A two-player extensive-form game $\Gamma := \langle \mathcal{H}, \mathcal{A}, \mathcal{Z}, p, I \rangle$ has **perfect recall** if and only if for all states $h_1, h_2 \in \mathcal{H}_i$ with $I(h_1) = I(h_2)$ the following holds:*

The sets $\mathcal{P}(h_1) \cap \mathcal{H}_i := (p_1, \ldots, p_k, h_1)$ and $\mathcal{P}(h_2) \cap \mathcal{H}_i := (q_1, \ldots, q_m, h_2)$ satisfy:

(1) $k = m$.

(2) $I(p_\ell) = I(q_\ell)$ *for all* $\ell \in \{1, \ldots, k\}$.

(3) $p_{\ell+1} \in \text{Next}(p_\ell, \alpha, i)$ *and* $q_{\ell+1} \in \text{Next}(q_\ell, \alpha, i)$ *for some action* $\alpha \in \mathcal{A}(p_\ell)$ *(since* $\mathcal{A}(p_\ell) = \mathcal{A}(q_\ell)$*).*

It is important to note that Kuhn's Theorem [98] states that in games with perfect recall, behavioral strategies are equivalent to the more familiar *mixed* strategies (i.e. probability distribution over pure strategies). Furthermore, perfect recall is typically satisfied for most extensive-form games of interest.

6.2.2 Two-Player Extensive-Form Games in Sequence Form

A key aspect of our analysis has to do with how strategies of each player are represented. Because EFGs often have large numbers of strategies, the behavioral plans described in the previous section are not efficient, since the expected payoff of each player requires a product over potentially many information sets. To deal with this, an alternative formulation called the *sequence form* was introduced by [175], which allows for a more convenient representation of strategies in EFGs. In the sequence form representation, any two-player EFG Γ can be captured by a two-player bilinear game where the action spaces of the players are a specific kind of polytope, commonly known as a *treeplex* [82]. Intuitively, rather than selecting a behavioral plan for every information set, players instead look at the terminal/leaf nodes of the game tree and consider the sequence of strategies needed to arrive at that leaf. In order to formally define the notion of a treeplex and the sequence form, we first need to introduce some additional notation.

Definition 6.2.6. *For a two-player extensive-form game Γ, we define the following:*

- $\mathcal{P}(h)$ *denotes the path from the root state $r \in \mathcal{H}$ to the state $h \in \mathcal{H}$.*
- $\mathrm{Level}(h)$ *denotes the distance from the root state $r \in \mathcal{H}$ to state $h \in \mathcal{H}$.*
- $\mathrm{Prev}(h, i)$ *denotes the lowest ancestor of h in the set $\mathcal{H}_i$. In particular,*

$$\mathrm{Prev}(h, i) = \mathrm{argmax}_{h' \in \mathcal{P}(h) \cap \mathcal{H}_i} \mathrm{Level}(h').$$

- *The set of states $\mathrm{Next}(h, \alpha, i) \subseteq \mathcal{H}$ denotes the highest descendants $h' \in \mathcal{H}_i$ once action $\alpha \in \mathcal{A}(h)$ has been taken at state h. More formally, $h' \in \mathrm{Next}(h, \alpha, i)$ if and only if in the path $\mathcal{P}(h, h') = (h, h_1, \ldots, h_k, h')$, all states $h_\ell \notin \mathcal{H}_i$ and $h_1 = \mathrm{Next}(h, \alpha)$.*

Definition 6.2.7. *Given a two-player extensive-form game Γ, the 'treeplex' set $\mathcal{X}_i^\Gamma$ is composed by all vectors $x_i \in [0, 1]^{|\mathcal{H}_i| + |\mathcal{Z}|}$ which satisfy the following constraints:*

(1) $x_i(h) = 1$ *for all $h \in \mathcal{H}_i$ with $\mathrm{Prev}(h, i) = \varnothing$.*

(2) $x_i(h_1) = x_i(h_2)$ *if there exists $h_1', h_2' \in \mathcal{H}_i$ such that $h_1 \in \mathrm{Next}(h_1', \alpha, i)$, $h_2 \in \mathrm{Next}(h_2', \alpha, i)$ and $I(h_1') = I(h_2')$.*

(3) $\sum_{\alpha \in \mathcal{A}(h)} x_i(\mathrm{Next}(h, \alpha, i)) = x_i(h)$ *for all $h \in \mathcal{H}_i$.*

A vector $x_i \in \mathcal{X}_i^\Gamma$ is typically referred to as a player i's *strategy in sequence form*. Strategies in sequence form come as an alternative to the behavioral plans of Definition 6.2.2. What Definition 6.2.7 establishes is that for every probability vector specifying the players' strategies over each information set, players can instead select vectors x which are of size equal to all possible actions throughout the game. A treeplex thus is a tree with nodes that alternate between sequences and information sets. Importantly, as we establish in Lemma 6.2.1, there exists an equivalence between a behavioral plan $\sigma_i \in \Sigma_i$ and a strategy in sequence form $x_i \in \mathcal{X}_i^\Gamma$ for games with perfect recall.

Lemma 6.2.1. *Consider a two-player extensive-form game Γ with perfect recall and the $(|\mathcal{H}_1| + |\mathcal{Z}|) \times (|\mathcal{H}_2| + |\mathcal{Z}|)$ dimensional matrices A_1^Γ, A_2^Γ with $[A_i^\Gamma]_{zz} = p_i(z)$ for all terminal nodes $z \in \mathcal{Z}$ and 0 otherwise. There exists a polynomial-time algorithm transforming any behavioral*

plan $\sigma_i \in \Sigma_i$ to a vector $x_{\sigma_i} \in \mathcal{X}_i^\Gamma$ such that

$$U_1(\sigma_1, \sigma_2) = x_{\sigma_1}^\top \cdot A_1^\Gamma \cdot x_{\sigma_2} \quad \textit{and} \quad U_2(\sigma_1, \sigma_2) = x_{\sigma_2}^\top \cdot A_2^\Gamma \cdot x_{\sigma_1}$$

Conversely, there exists a polynomial-time algorithm transforming any vector $x_i \in \mathcal{X}_i^\Gamma$ to a vector $\sigma_{x_i} \in \Sigma_i$ such that

$$x_1^\top \cdot A_1^\Gamma \cdot x_2 = U_1(\sigma_{x_1}, \sigma_{x_2}) \quad \textit{and} \quad x_2^\top \cdot A_2^\Gamma \cdot x_1 = U_2(\sigma_{x_1}, \sigma_{x_2})$$

Because of this result, it is clear why strategies in sequence form are of great use. Assume that player 2 selects a behavioral plan $\sigma_2 \in \Sigma_2$. Then, player 1 wants to compute a behavioral plan $\sigma_1^* \in \Sigma_1$ which is a *best response* to σ_2, namely $\sigma_1^* := \text{argmax}_{\sigma_1 \in \Sigma_1} U_1(\sigma_1, \sigma_2)$. This computation can be done in polynomial-time in the following manner: Player 1 initially converts (in polynomial time) the behavioral plan σ_2 to $x_{\sigma_2} \in \mathcal{X}_2^\Gamma$, which is the sequence form representation of the original strategy. Then, they can obtain a vector $x_1^* = \text{argmax}_{x_1 \in \mathcal{X}_1^\Gamma} x_1^\top \cdot A_1^\Gamma \cdot x_2$. This step can be performed in polynomial-time by computing the solution of an appropriate linear program[175]. Finally, they can convert the vector x_1^* to a behavioral plan $\sigma_{x_1^*} \in \Sigma_1$ in polynomial-time. Lemma 6.2.1 ensures that $\sigma_{x_1^*} = \text{argmax}_{\sigma_1 \in \Sigma_1} U_1(\sigma_1, \sigma_2)$.

The above reasoning can be used to establish an equivalence between the Nash Equilibrium (σ_1^*, σ_2^*) of an EFG $\Gamma := \langle \mathcal{H}, \mathcal{A}, \mathcal{Z}, p, I \rangle$ with the Nash Equilibrium in its sequence form.

Definition 6.2.8. *A Nash Equilibrium of a two-player EFG Γ in sequence form is a vector $(x_1^*, x_2^*) \in \mathcal{X}_1^\Gamma \times \mathcal{X}_2^\Gamma$ such that*

- $(x_1^*)^\top \cdot A_1^\Gamma \cdot x_2^* \geq (x_1)^\top \cdot A_1^\Gamma \cdot x_2^*$ *for all* $x_1 \in \mathcal{X}_1^\Gamma$
- $(x_2^*)^\top \cdot A_2^\Gamma \cdot x_1^* \geq (x_2)^\top \cdot A_2^\Gamma \cdot x_1^*$ *for all* $x_2 \in \mathcal{X}_2^\Gamma$

Hence, Lemma 6.2.1 directly implies that any Nash Equilibrium of an EFG $(\sigma_1^*, \sigma_2^*) \in \Sigma_1 \times \Sigma_2$ as per Definition 6.2.4 can be converted in polynomial-time to a Nash Equilibrium in sequence form $(x_1^*, x_2^*) \in \mathcal{X}_1^\Gamma \times \mathcal{X}_2^\Gamma$ and vice versa.

6.2.3 Optimistic Mirror Descent

In this section we introduce an extension of the Mirror Descent framework defined in Chapter 2.2, known as *Optimistic Mirror Descent* [141]. First, let us define `OMD` in its standard form, from the perspective of cost minimization. In the following chapters, we will transform this into a utility maximization update instead.

For a convex function $h : \mathbb{R}^d \mapsto \mathbb{R}$, the corresponding *Bregman divergence* is defined as

$$D_h(x, y) := h(x) - h(y) - \langle \nabla h(y), x - y \rangle$$

If h is γ-strongly convex, then $D_h(x, y) \geq \frac{\gamma}{2}\|x - y\|$, where $\|\cdot\|$ is shorthand for the L_2-norm.

Now consider a game played by n players, where the action of each player i is a vector x_i from a convex set $\mathcal{X}_i$. Each player selects their action $x_i \in \mathcal{X}_i$ so as to minimize their

individual cost (denoted by $C_i(x_i, x_{-i})$), which is continuous, differentiable and convex with respect to x_i. Specifically,

$$C_i(\lambda \cdot x_i + (1-\lambda)\cdot x_i', x_{-i}) \leq \lambda \cdot C_i(x_i, x_{-i}) + (1-\lambda)\cdot C_i(x_i', x_{-i}) \text{ for all } \lambda \in [0,1]$$

Given a step size $\eta > 0$ and a regularizer $\psi(\cdot)$, *Optimistic Mirror Descent* (OMD) sequentially performs the following update step for $t = 1, 2, \ldots$:

$$\begin{aligned} x_i^t &= \text{argmin}_{x\in\mathcal{X}_i} \left\{ \eta \left\langle x, F_i^{t-1}(x) \right\rangle + D_\psi \left(x, \hat{x}_i^t\right) \right\} \\ \hat{x}_i^{t+1} &= \text{argmin}_{x\in\mathcal{X}_i} \left\{ \eta \left\langle x, F_i^{t}(x) \right\rangle + D_\psi (x, \hat{x}_i^t) \right\} \end{aligned} \tag{OMD}$$

where $F_i^t(x_i) = \nabla_{x_i} C_i(x_i, x_{-i}^t)$ and $D_h(x, y)$ is the *Bregman divergence* with respect to $h(\cdot)$. If the step-size η selected is sufficiently small, then OMD exhibits the no-regret property [141], making it a natural update algorithm for selfish players in the game-theoretic setting [80]. To simplify notation in the rest of the chapter, we will denote the projection operator of a convex set $\mathcal{X}^*$ as $\Pi_{\mathcal{X}^*}(x) := \text{argmax}_{x^*\in\mathcal{X}^*} \|x - x^*\|$ and the squared distance of vector x from a convex set $\mathcal{X}^*$ as $\text{dist}^2(x, \mathcal{X}^*) := \|x - \Pi_{\mathcal{X}^*}(x)\|^2$.

6.3 Our Setting

In this section, we introduce the concept of Network Zero-Sum Extensive-Form Games, which are a network extension of the two player EFGs introduced in Section 6.2.

6.3.1 Network Zero-Sum Extensive-Form Games

A *network extensive-form game* is defined with respect to an undirected graph $\mathcal{G} = (V, E)$ where nodes V ($|V| = n$) correspond to the set of players and each edge $(u, v) \in E$ represents a *two-player extensive-form game* Γ^{uv} played between players u, v. Each node/player $u \in V$ selects a single behavioral plan $\sigma_u \in \Sigma_u$ which they use to play all the two-player EFGs on its outgoing edges.

Definition 6.3.1 (Network Extensive-Form Games)**.** *A network extensive-form game is a tuple $\Gamma := \langle \mathcal{G}, \mathcal{H}, \mathcal{A}, \mathcal{Z}, I \rangle$ where*

- *$\mathcal{G} = (V, E)$ is an undirected graph where the nodes V represents the players.*
- *Each player $u \in V$ has a set of states $\mathcal{H}_u$ at which they play. Each such state $h \in \mathcal{H}_u$ has a set $\mathcal{A}(h)$ of possible actions that player u can take.*
- *$\mathrm{I}(h)$ denotes the information set of $h \in \mathcal{H}_u$. If $I(h) = I(h')$ for some $h, h' \in \mathcal{H}_u$, then $\mathcal{A}(h) = \mathcal{A}(h')$.*
- *Each edge $(u, v) \in E$, Γ^{uv} is a two-player extensive-form game with **perfect recall**. The states of Γ^{uv} are denoted by $\mathcal{H}^{uv} \subseteq \mathcal{H}_u \cup \mathcal{H}_v$.*
- *$\mathcal{Z}^{uv}$ is the set of terminal states of the two-player EFG Γ^{uv}, and $p_u^{\Gamma^{uv}}(z)$ denotes the payoffs of u at the terminal state $z \in \mathcal{Z}^{uv}$ (likewise for player v). The overall set of terminal states of the network extensive-form game is the set $\mathcal{Z} := \cup_{(u,v)\in E} \mathcal{Z}^{uv}$.*

In a network extensive-form game, each player $u \in V$ selects a behavioral plan $\sigma_u \in \Sigma_u$ (see Definition 6.2.2) that they use to play the two-player EFG's Γ^{uv} with $(u, v) \in E$. Each player selects their behavioral plan so as to maximize the sum of the payoffs of the two-player EFGs in their outgoing edges.

Definition 6.3.2. *Given a collection of behavioral plans $\sigma = (\sigma_1, \ldots, \sigma_n) \in \Sigma_1 \times \ldots \times \Sigma_n$ the payoff of player u, denoted by $U_u(\sigma)$, is given by:*

$$U_u(\sigma) := \sum_{v:(u,v)\in E} p_u^{\Gamma^{uv}}(\sigma_u, \sigma_v)$$

Moreover a collection $\sigma^* = (\sigma_1^*, \ldots, \sigma_n^*) \in \Sigma_1 \times \ldots \times \Sigma_n$ is called a Nash Equilibrium if and only if

$$U_u(\sigma_u^*, \sigma_{-u}^*) \geq U_u(\sigma_u, \sigma_{-u}^*) \quad \text{for all } \sigma_u \in \Sigma_u$$

As already mentioned, each player $u \in V$ plays all the two-player EFGs Γ^{uv} for $(u, v) \in E$ with the same behavioral plan $\sigma_u \in \Sigma_u$. This is due to the fact that the player cannot distinguish between a state $h_1, h_2 \in \mathcal{H}_u$ with $I(h_1) = I(h_2)$, even if h_1, h_2 are states of different EFGs Γ^{uv} and $\Gamma^{uv'}$. Because all the games have *perfect recall*, this implies that player u cannot differentiate between states h_1, h_2 due to their perfect memory of the states $\mathcal{H}_u$ visited in the past and their past actions.

In Definition 6.3.3, we introduce the notion of *consistency* in network EFGs, an extension to perfect recall in two-player EFGs. Much like perfect recall, we will assume that all the network EFGs we study are *consistent*.

Definition 6.3.3. *A network extensive-form game $\Gamma := \langle \mathcal{G}, \mathcal{H}, \mathcal{A}, \mathcal{Z}, I \rangle$ is called **consistent** if and only if for all players $u \in V$ and states $h_1, h_2 \in \mathcal{H}_u$ with $I(h_1) = I(h_2)$ the following holds:*

For any $(u, v), (u, v') \in E$, the sets $\mathcal{P}^{uv}(h_1) \cap \mathcal{H}_u := (p_1, \ldots, p_k, h_1)$ and $\mathcal{P}^{uv'}(h_2) \cap \mathcal{H}_u := (q_1, \ldots, q_m, h_2)$ satisfy:

(1) $k = m$.

(2) $I(p_\ell) = I(q_\ell)$ for all $\ell \in \{1, k\}$.

(3) $p_{\ell+1} \in \text{Next}^{\Gamma_{uv}}(p_\ell, \alpha, u)$ and $q_{\ell+1} \in \text{Next}^{\Gamma_{uv'}}(q_\ell, \alpha, u)$ for some action $\alpha \in \mathcal{A}(p_\ell)$.

where $\mathcal{P}^{uv}(h)$ denotes the path from the root state to state h in the two-player extensive-form game Γ^{uv}.

This is a direct extension of the perfect recall property, except that now the property extends to all the games which player u is involved in.

In this work we study the special class of network *zero-sum* extensive-form games. This class of games is a generalization of the network zero-sum normal form games studied in [32].

Definition 6.3.4. *A behavioral plan $\sigma_u \in \Sigma_u$ of Definition 6.2.2 is called pure if and only if $\sigma_u(h, \alpha)$ either equals 0 or 1 for all actions $\alpha \in \mathcal{A}(h)$. A network extensive-form game is*

called ***zero-sum*** *if and only if for any collection* $\sigma := (\sigma_1, \ldots, \sigma_n)$ *of pure behavioral plans,* $U_u(\sigma) = 0$ *for all* $u \in V$.

6.3.2 Network Extensive-Form Games in Sequence Form

As in the case of two-player EFGs, there exists an equivalence between behavioral plans $\sigma_u \in \Sigma_u$ and strategies in sequence form x_u. As we shall later see, this equivalence is of great importance since it allows for the design of natural and computationally efficient learning dynamics that converge to Nash Equilibria both in terms of behavioral plans and strategies in sequence form.

Definition 6.3.5. *Given a network extensive-form game* $\Gamma := \langle \mathcal{G}, \mathcal{H}, \mathcal{A}, \mathcal{Z}, I \rangle$*, the treeplex polytope* $\mathcal{X}_u \subseteq [0,1]^{|\mathcal{H}_u|+|\mathcal{Z}_u|}$ *is the set defined as follows:*

$x_u \in \mathcal{X}_u$ *if and only if*

(1) $x_u \in \mathcal{X}_u^{\Gamma_{uv}}$ *for all* $(u,v) \in E$.

(2) $x_u(h_1) = x_u(h_2)$ *if there exists* $(u,v), (u,v') \in E$ *and* $h_1', h_2' \in \mathcal{H}_u$ *with* $I(h_1') = I(h_2')$ *such that* $h_1 \in \mathrm{Next}^{\Gamma_{uv}}(h_1', \alpha, u)$*,* $h_2 \in \mathrm{Next}^{\Gamma_{uv'}}(h_2', \alpha, u)$ *and* $I(h_1') = I(h_2')$.

The second constraint in Definition 6.3.5 is the equivalent of the second constraint in Definition 6.2.7. Similarly to the two-player case, the treeplex polytope allows us to express players' strategies in terms of a collection of paths along a tree, rather than a collection of behavioral plans, which simplifies the expected payoff representation.

We remark that much like in the two-player case, the linear equations describing the treeplex polytope $\mathcal{X}_u$ can be derived in polynomial-time with respect to the description of the network extensive-form game. In Lemma 6.3.1 we formally state and prove the equivalence between behavioral plans and strategies in sequence form.

Lemma 6.3.1. *Consider the matrix* A^{uv} *of dimensions* $(|\mathcal{H}_u| + |\mathcal{Z}^u|) \times (|\mathcal{H}_v| + |\mathcal{Z}^v|)$ *such that*

$$[A^{uv}]_{h_1 h_2} = \begin{cases} p_u^{\Gamma_{uv}}(h) & \text{if } h_1 = h_2 = h \in \mathcal{Z}^{uv} \\ 0 & \text{otherwise} \end{cases}$$

There exists a polynomial time algorithm converting any collection of behavioral plans $(\sigma_1, \ldots, \sigma_n) \in \Sigma_1 \times \ldots \times \Sigma_n$ *into a collection of vectors* $(x_1, \ldots, x_n) \in \mathcal{X}_1 \times \ldots \times \mathcal{X}_n$ *such that for any* $u \in V$*,*

$$U_u(\sigma) = x_u^\top \cdot \sum_{v:(u,v)\in E} A^{uv} \cdot x_v$$

In the opposite direction, there exists a polynomial time algorithm converting any collection of vectors $(x_1, \ldots, x_n) \in \mathcal{X}_1 \times \ldots \times \mathcal{X}_n$ *into a collection of behavioral plans* $(\sigma_1, \ldots, \sigma_n) \in \Sigma_1 \times \ldots \times \Sigma_n$ *such that for any* $u \in V$*,*

$$x_u^\top \cdot \sum_{v:(u,v)\in E} A^{uv} \cdot x_v = U_u(\sigma)$$

Definition 6.3.6. *A Nash Equilibrium of a network extensive-form game $\mathcal{G}$ in sequence form is a vector $(x_1^*, \ldots, x_n^*) \in \mathcal{X}_1 \times \ldots \times \mathcal{X}_n$ such that for all $u \in V$:*

$$(x_u^*)^\top \cdot \sum_{v:(u,v)\in E} A^{uv} \cdot x_v^* \geq x_u^\top \cdot \sum_{v:(u,v)\in E} A^{uv} \cdot x_v^* \quad \text{for all } x_u \in \mathcal{X}_u$$

Corollary 6.3.1. *Given a network extensive-form game, any Nash Equilibrium $(\sigma_1^*, \ldots, \sigma_n^*) \in \Sigma_1 \times \ldots \times \Sigma_n$ (as per Definition 6.2.4) can be converted in polynomial-time to a Nash Equilibrium $(x_1^*, \ldots, x_n^*) \in \mathcal{X}_1 \times \ldots \times \mathcal{X}_n$ (as per Definition 6.3.6) and vice versa.*

Due to the simpler representation of expected payoffs, the sequence form representation gives us a perspective with which we can analyze the theoretical properties of online learning algorithms when applied to network zero-sum EFGs. In the following section, we utilize the sequence form representation to study a special case of Optimistic Mirror Descent known as Optimistic Gradient Ascent (OGA) in network zero-sum EFGs.

6.4 Convergence Results for OGA

In this section, we study the convergence properties of *Optimistic Gradient Ascent* (OGA) when applied to network zero-sum EFGs. To do so, we utilize the Optimistic Mirror Descent (OMD) framework described earlier. Similar to the case of FTRL in previous chapters, one can select the regularizer of the OMD update in order to recover specific learning dynamics. Of particular interest to us is OGA, which is a special case of OMD where the regularizer is $h(a) = \frac{1}{2}\|a\|^2$, which means that the Bregman divergence $D_h(x, y)$ equals $\frac{1}{2}\|x - y\|^2$. Note that in network zero-sum EFGs, players are trying to maximize their payoffs/utilities, so we specify that the update is a form of gradient *ascent* rather than *descent*. Hence, the minimization at every time-step now becomes maximization instead. In particular, OGA takes the following form:

$$\begin{aligned} x_u^t &= \text{argmax}_{x\in\mathcal{X}_u} \left\{ \eta \left\langle x, \sum_{v:(u,v)\in E} A^{uv} \cdot x_v^{t-1} \right\rangle - D_h\left(x, \hat{x}_u^t\right) \right\} \\ \hat{x}_u^{t+1} &= \text{argmax}_{x\in\mathcal{X}_u} \left\{ \eta \left\langle x, \sum_{v:(u,v)\in E} A^{uv} \cdot x_v^{t} \right\rangle - D_h\left(x, \hat{x}_u^t\right) \right\} \end{aligned} \tag{OGA}$$

Note that in the discrete-time case, we use T to denote the time horizon of the dynamics (i.e., the total number of timesteps that the system is run for). In Theorem 6.4.1 we describe the $\Theta(1/T)$ convergence rate to NE for the time-average strategies for any player using OGA.

Theorem 6.4.1. *Let $\{x^1, x^2, \ldots x^T\}$ be the vectors produced by* (OGA) *for some initial strategies $x^0 := (x_1^0, \ldots, x_n^0)$ in a network zero-sum EFG. Then, there exist game-dependent constants $c_1, c_2 > 0$ such that if $\eta \leq 1/c_1$ then for any $u \in V$:*

$$\hat{x}_u^\top \cdot \sum_{v:(u,v)\in E} A^{uv} \cdot \hat{x}_v \geq x^\top \cdot \sum_{v:(u,v)\in E} A^{uv} \cdot \hat{x}_v - \Theta\left(\frac{c_1 \cdot c_2}{T}\right) \quad \text{for all } x \in \mathcal{X}_u$$

where $\hat{x}_u = \sum_{s=1}^{T} x_u^s/T$ *is the time-average strategy of player* u.

Applying the polynomial-time transformation of Lemma 6.3.1 to the time-average strategy vector $\hat{x} = (\hat{x}_1, \ldots, \hat{x}_n)$ produced by Optimistic Gradient Ascent, we immediately get that for any player $u \in V$,

$$U_u(\hat{\sigma_u}, \hat{\sigma}_{-u}) \geq U_u(\sigma_u, \hat{\sigma}_{-u}) - \Theta\left(c_1 \cdot c_2/T\right) \quad \text{for all } \sigma_u \in \Sigma_u$$

Next, in Theorem 6.4.2, we establish the fact that OGA enjoys last-iterate convergence with linear rate to the set of Nash equilibria in network zero-sum EFGs.

Theorem 6.4.2. *Let* $\{x^1, x^2, \ldots, x^t, \ldots, x^T\}$ *be the vectors produced by* (OGA) *for* $\eta \leq 1/c_3$ *when applied to a network zero-sum EFG. Then, the following inequality holds:*

$$\mathrm{dist}^2(x^t, \mathcal{X}^*) \leq 64\mathrm{dist}^2(x^1, \mathcal{X}^*) \cdot (1 + c_1)^{-t}$$

where $\mathcal{X}^*$ *denotes the set of Nash Equilibria,* $c_1 := \min\left\{\frac{16\eta^2 c^2}{81}, \frac{1}{2}\right\}$ *and* c_3, c *are positive game-dependent constants.*

This rather surprising result states that not only does OGA converge in a last-iterate sense to the set of Nash equilibria, it also does so at a linear rate! The process of showing these results requires us to take a detour via a transformation of the extensive-form game payoffs. To be precise, we formulate the zero-sum multiplayer extensive-form game as a two-player symmetric game, which greatly simplifies the notation in our proofs.

6.4.1 Proof Outline

We conclude this section by providing the key ideas towards proving Theorems 6.4.1 and 6.4.2. For the rest of the section, we assume that the network extensive-form games we study are consistent and zero-sum. Before proceeding, we need to introduce a few more necessary definitions and notations. We denote as $\mathcal{X} := \mathcal{X}_1 \times \ldots \times \mathcal{X}_n$ the product of treeplexes of Definition 6.3.5 and define the $|\mathcal{X}| \times |\mathcal{X}|$ matrix R as follows:

$$R_{(u:h_1),(v:h_2)} = \begin{cases} -[A^{uv}]_{h_1 h_2} & \text{if } (u,v) \in E \\ 0 & \text{otherwise} \end{cases}$$

The matrix R can be used to derive a more concrete/simple form of the OGA update:

Lemma 6.4.1. *Let* $\{x^1, x^2, \ldots, x^t, \ldots, x^T\}$ *be the collection of strategy vectors produced by* (OGA) *initialized with* $x^0 := (x_1^0, \ldots, x_n^0) \in \mathcal{X}$. *The equations*

$$\begin{aligned} x^t &= \mathit{argmin}_{x \in \mathcal{X}} \left\{\eta \left\langle x, R \cdot x^{t-1} \right\rangle + D_\psi\left(x, \hat{x}^t\right)\right\} \\ \hat{x}^{t+1} &= \mathit{argmin}_{x \in \mathcal{X}} \left\{\eta \left\langle x, R \cdot x^{t} \right\rangle + D_\psi\left(x, \hat{x}^t\right)\right\} \end{aligned} \qquad \text{(s-OGD)}$$

produce the exact same collection of strategy vectors $\{x^1, x^2, \ldots, x^t, \ldots, x^T\}$ *when initialized with* $x^0 \in \mathcal{X}$, *and are a simplified version of the* (OGA) *update (the* s *in the equation label stands for Simplified).*

Proof. First, since (s−OGD) is defined on the product of treeplexes $\mathcal{X}$, let us decompose the equations from the perspective of an arbitrary player u. Specifically, for some x_u^t, $u \in \{1, \ldots, n\}$ it holds that the inner product $\langle x, R \cdot x^{t-1} \rangle$, $x \in \mathcal{X}$ can be decomposed into inner products of the form $\langle x, R \cdot x_u^{t-1} \rangle$, where x is now in the individual treeplex $\mathcal{X}_u$. Moreover, by the definition of matrix R, we can substitute:

$$R_{(u:h_1),(v:h_2)} = -[A^{uv}]_{h_1 h_2}$$

for all $(u, v) \in E$ and 0 otherwise. Effectively, from the perspective of player u, the product of R and x^t gives us $\sum_{(u,v)\in E} A^{u,v} \cdot x_v^t$. This gives us the following:

$$x_u^t = \text{argmin}_{x \in \mathcal{X}_u} \left\{ \eta \left\langle x, - \sum_{v:(u,v)\in E} A^{uv} \cdot x_v^{t-1} \right\rangle + D_\psi \left(x, \hat{x}_u^t \right) \right\} \tag{6.2}$$

$$\hat{x}_u^{t+1} = \text{argmin}_{x \in \mathcal{X}_u} \left\{ \eta \left\langle x, - \sum_{v:(u,v)\in E} A^{uv} \cdot x_v^{t} \right\rangle + D_\psi \left(x, \hat{x}_u^t \right) \right\} \tag{6.3}$$

Finally, we can take the negative of the terms inside the braces to obtain (OGA). Hence, for every strategy vector x updated using (s−OGD), the constituent strategy vectors for each player u are exactly the same as (OGA). Thus if the initial conditions x^0 are the same, for all time t the collection of strategy vectors $\{x^1, \ldots, x^T\}$ are the same between both formulations. □

Using the above definition of R, we can derive a *two-player symmetric game* (R, R) defined over the polytope $\mathcal{X}$. More precisely, the x-player selects $x \in \mathcal{X}$ so as to minimize $x^\top Ry$ while the y-player selects $y \in \mathcal{X}$ so as to minimize $y^\top Rx$. Now, consider the Optimistic Mirror Descent algorithm (described in (OMD)) applied to the above symmetric game. Notice that if $x^0 = y^0$, then by the symmetry of the game, the strategy vector (x^t, y^t) at time t will be of the form (x^t, x^t) and indeed, $(x^t, \hat{x}^t)$ will satisfy (s−OGD). We prove that the produced vector sequence $\{x^t\}_{t\geq 1}$ converges to a *symmetric Nash Equilibrium*.

Lemma 6.4.2. *A strategy vector x^* is an ϵ-symmetric Nash Equilibrium for the symmetric game (R, R) if the following holds:*

$$(x^*)^\top \cdot R \cdot x^* \leq x^\top \cdot R \cdot x^* + \epsilon \quad \textit{for all } x \in \mathcal{X}$$

Any ϵ-symmetric Nash Equilibrium $x^* \in \mathcal{X}$ is also an ϵ-Nash Equilibrium for the network zero-sum EFG.

Time-average convergence proof idea. With all that out of the way, we can start proving time-average convergence. A key property of the constructed matrix R is stated and proven in Lemma 6.4.3. Its proof follows the steps of the proof of Lemma B.3 in [32] and is presented in Appendix C.4.

Lemma 6.4.3. $x^\top \cdot R \cdot y + y^\top \cdot R \cdot x = 0$ *for all* $x, y \in \mathcal{X}$.

Once Lemma 6.4.3 is established, we can use to it to prove that the time-average strategy vector converges to an ϵ-symmetric Nash Equilibrium in a two-player symmetric game.

Lemma 6.4.4. *Let $(x^1, x^2, \ldots, x^T)$ be the sequence of strategy vectors produced by* (s−OGD) *for $\eta \leq \min\{1/8\|R\|^2, 1\}$. Then,*

$$min_{x \in \mathcal{X}} x^\top \cdot R \cdot \hat{x} \geq -\Theta\left(\frac{\mathcal{D}^2 \|R\|^2}{T}\right)$$

where $\hat{x} = \sum_{s=1}^{T} x^s/T$ and $\mathcal{D}$ is the diameter of the treeplex polytope $\mathcal{X}$.

Combining Lemma 6.4.3 with Lemma 6.4.4, we get that that the time-average vector $\hat{x}$ is a $\Theta\left(\frac{\mathcal{D}^2\|R\|^2}{T}\right)$-symmetric Nash Equilibrium. This follows directly from the fact that $\hat{x}^\top \cdot R \cdot \hat{x} = 0$. Then, Theorem 6.4.1 follows via a direct application of Lemma 6.4.2. For completeness, we present the complete proof of Theorem 6.4.1 in Appendix C.6.

Last-iterate convergence proof idea. Finally, the last-iterate convergence proof utilizes Lemma 6.4.3 in the following manner: Lemma 6.4.3 directly implies that the set of symmetric Nash Equilibria can be written as:

$$\mathcal{X}^* = \{x^* \in \mathcal{X} : \min_{x \in \mathcal{X}} x^\top \cdot R \cdot x^* = 0\}.$$

Using this, we further establish that Optimistic Gradient *Descent* admits last-iterate convergence to the symmetric NE of the (R, R) game. This result is formally stated and proven in Theorem 6.4.3, the proof of which is adapted from the analysis of [178], with modifications to apply the steps to our setting. For completeness, the full proof of Theorem 6.4.3 is given in Appendix C.7.

Theorem 6.4.3. *Let $\{x^1, x^2, \ldots, x^t, \ldots, x^T\}$ be the vectors produced by* (s−OGD) *for $\eta \leq \min(1/8\|R\|^2, 1)$. Then:*

$$\text{dist}^2(x^t, \mathcal{X}^*) \leq 64\text{dist}^2(x^1, \mathcal{X}^*) \cdot (1 + C_2)^{-t}$$

where $C_2 \coloneqq \min\left\{\frac{16\eta^2 C^2}{81}, \frac{1}{2}\right\}$ with C being a positive game-dependent constant.

The statement of Theorem 6.4.2 then follows directly by combining Theorem 6.4.3 and Lemma 6.4.2. This result generalizes previous last-iterate convergence results for the setting of two-player zero-sum EFGs, even for games without a unique Nash Equilibrium.

6.5 Simulations

In order to better visualize our theoretical results, we experimentally evaluate OGA when applied to various network extensive-form games (which are zero-sum and consistent). In the rest of this section, we present the experimental setup followed by the empirical observations obtained for our convergence results.

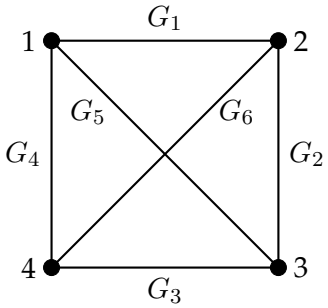


FIGURE 6.3: 4-node graph for randomized EFGs. Each node represents a player and each edge represents a game G_i between the corresponding players.

6.5.1 Experimental Setups

Random Network Extensive-Form Games. In our simulations, we first generate random zero-sum extensive-form games on both a 3-node graph where every player plays against the other two players, as well as a dense 4-node graph (shown in Figure 6.3). Specifically, each game is characterized by a 3×3 symmetric matrix which represents the sequence form of an extensive-form game written as a matrix. For each run of the simulation, we first create the games which are to be played, randomly generating matrices with elements in $[0, 1]$. Then, we optimize for the choice of step-size η, selecting the value that gives the fastest convergence rate to the Nash Equilibrium. In the plots, in order to reduce visual clutter, we present the squared distance from the Nash for only one of the players. In addition, in order to more clearly show the fast rate of convergence, we compute the logarithm of $\mathrm{dist}^2(x^t, \mathcal{X}^*)$ in the plots. It is worth noting that the 4-node graph takes significantly longer to arrive at the last iterate compared to the 3-node graph.

Kuhn Poker. Kuhn poker is a simplified version of poker proposed by [99]. The deck contains only three cards, namely Jack, Queen and King. Each player is dealt one card, and the third is left unseen. Player 1 can either check or bet, and subsequently Player 2 can also either check or bet. Finally, if Player 1 checks in round 1 and Player 2 bets in round 2, Player 1 gets another round to fold or call. Eventually, the player with the highest card wins the pot. The Kuhn poker game is presented in Figure 6.4. In the sequence form representation of the game, Kuhn poker has dimension $|\mathcal{X}| \times |\mathcal{X}| = 13 \times 13$ and the corresponding payoff matrix can be easily computed by hand. We run an experiment with 5 players on a graph where each player plays in exactly two Kuhn poker games with randomized initial conditions.

Time-average Convergence. Our theoretical results guarantee time-average convergence to the Nash Equilibrium set (Theorem 6.4.1). We experimentally confirm this by running OGA on a network version of the ubiquitous Matching Pennies game with 20 nodes (Figure 6.5 (a)), followed by a 4-node network zero-sum EFG (Figure 6.5 (b)). Next, we ran OGA on a Kuhn poker network. Emulating the illustrative example of a competitive online Poker lobby as described in Section 6.1, we modeled a situation whereby each player is playing against multiple other players, and ran simulations for

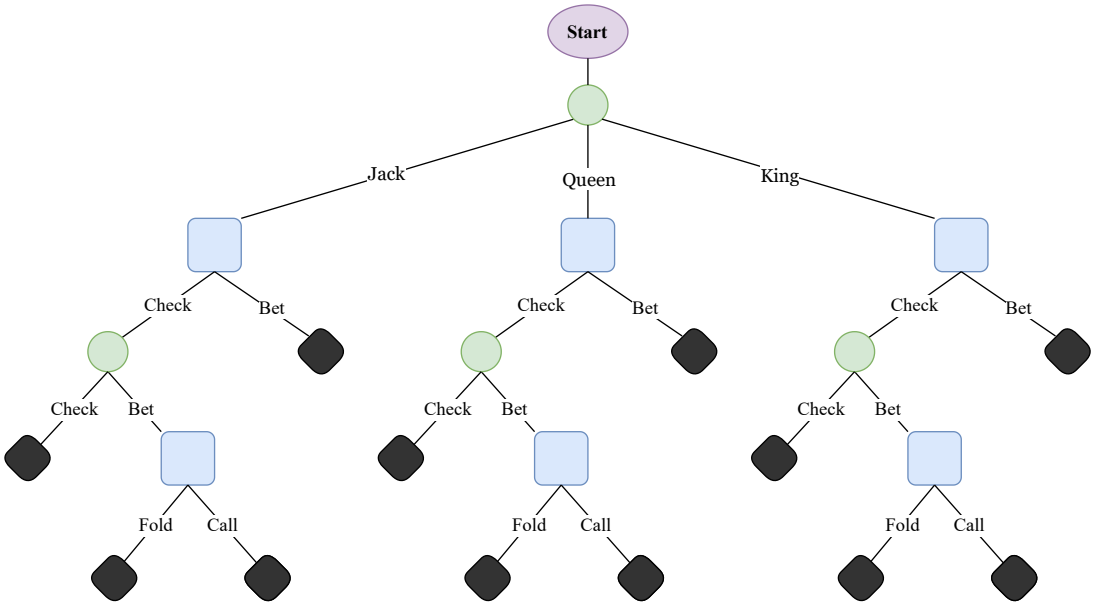


FIGURE 6.4: Extensive-form representation of Kuhn poker from the perspective of one player. The blue nodes represent decision points for the player, green nodes represent observation points (either the player observes their card or the other player takes an action) and finally the black nodes denote the terminal states of the game.

such a game with 5 agents (Figure 6.5 (c)).

In the plots, we show on the y-axis the difference between the cumulative averages of the strategy probabilities and the Nash Equilibrium value calculated from the game. In each of the plots, we see that these time-average values go to 0, implying convergence to the NE set.

Last-iterate Convergence. Theorem 6.4.2 guarantees $\mathcal{O}(c^{-t})$ convergence in the last-iterate sense to a Nash Equlibrium for OGA. Similar to the time-average case, we ran simulations for randomly generated 3 and 4-node network extensive-form games, where each bilinear game between two agents is a randomly generated matrix with values in $[0, 1]$ (Figure 6.6 (a-b)). Moreover, we also simulated a 5-node game of Kuhn poker in order to generate Figure 6.6 (c). In order to generate the plots, we measured the log of the distance between each agent's strategy at time t and the set of Nash Equilibria (computed *a priori*), given by $\log\big(\text{dist}^2(x^t, \mathcal{X}^*)\big)$. As can be seen in Figure 6.6, OGA indeed obtains fast convergence in the last-iterate sense to a Nash Equilibrium in each of our experiments.

A point worth noting is that when the number of nodes increases, the empirical last-iterate convergence time also increases drastically. For example, in the 5-player Kuhn poker game we see that each players' convergence time is significantly greater compared to the smaller scale experiments. However, with a careful choice of η, we can still guarantee convergence to the set of Nash Equilibria for all players.

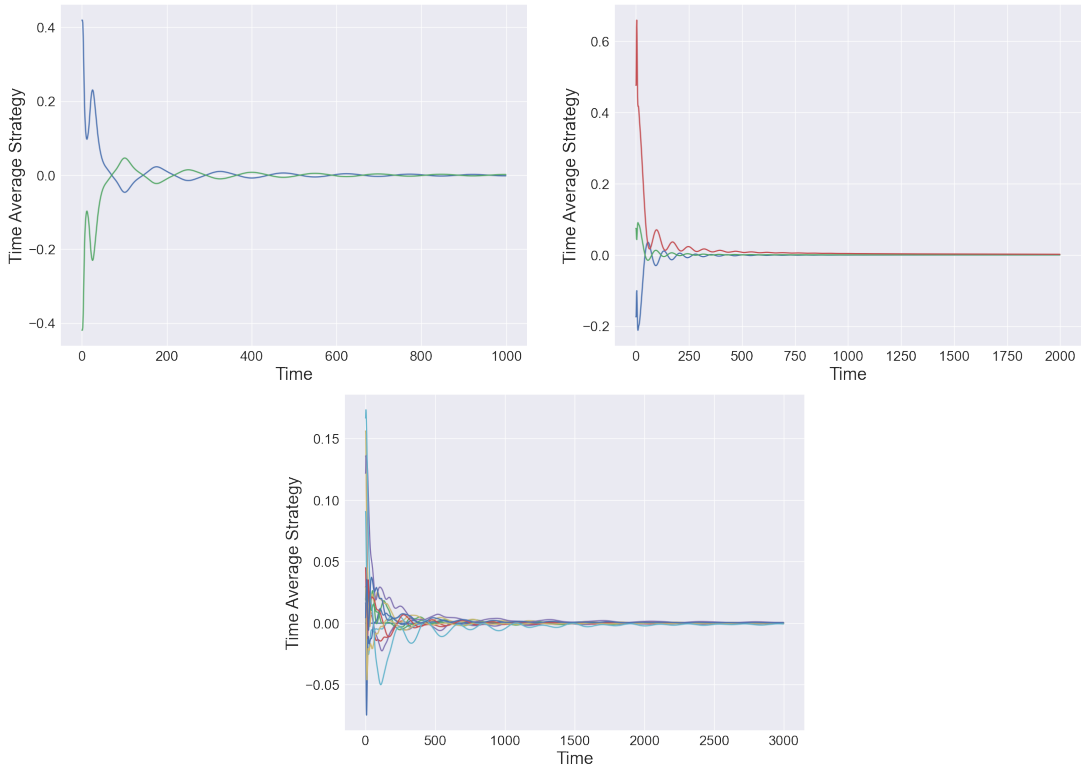


FIGURE 6.5: Time-average convergence of OGA in network zero-sum extensive-form games, where each player is involved in 2 or more different games and must select their strategy accordingly. (a) 20-node Matching Pennies game. (b) 4-node random extensive-form game. (c) 5-node *Kuhn poker* game.

A note on large-scale games. An empirical observation from our simulations is that the number of nodes in the network as well as the sparsity of the graph plays a major role in convergence times, particularly the *last-iterate* convergence times. This intuitive observation presents an interesting challenge when modeling truly large-scale problems. For instance, a setting such as Texas Hold'em poker admits a huge number of parameters (of order 10^{18}). Even in the two-player case this is prohibitively large, and this issue is compounded if we are in the multiplayer setting.

As an illustrative example, consider a network game where every player plays the ubiquitous zero-sum game, Matching Pennies, against two other players. Figure 6.7 shows that the convergence times drastically increase when we go from a 4-node graph to a 20-node graph. Similarly, in our experiments with extensive-form games in sequence form, it becomes difficult to simulate larger games (such as Leduc poker, which has dimension $|\mathcal{X}| \times |\mathcal{X}| = 337$) once there are multiple players playing in several games. This is a practical limitation which represents an interesting divide between our theoretical results and the reality of many large-scale, real world games. It is certainly a fascinating research direction to find ways to bridge this gap in future research.

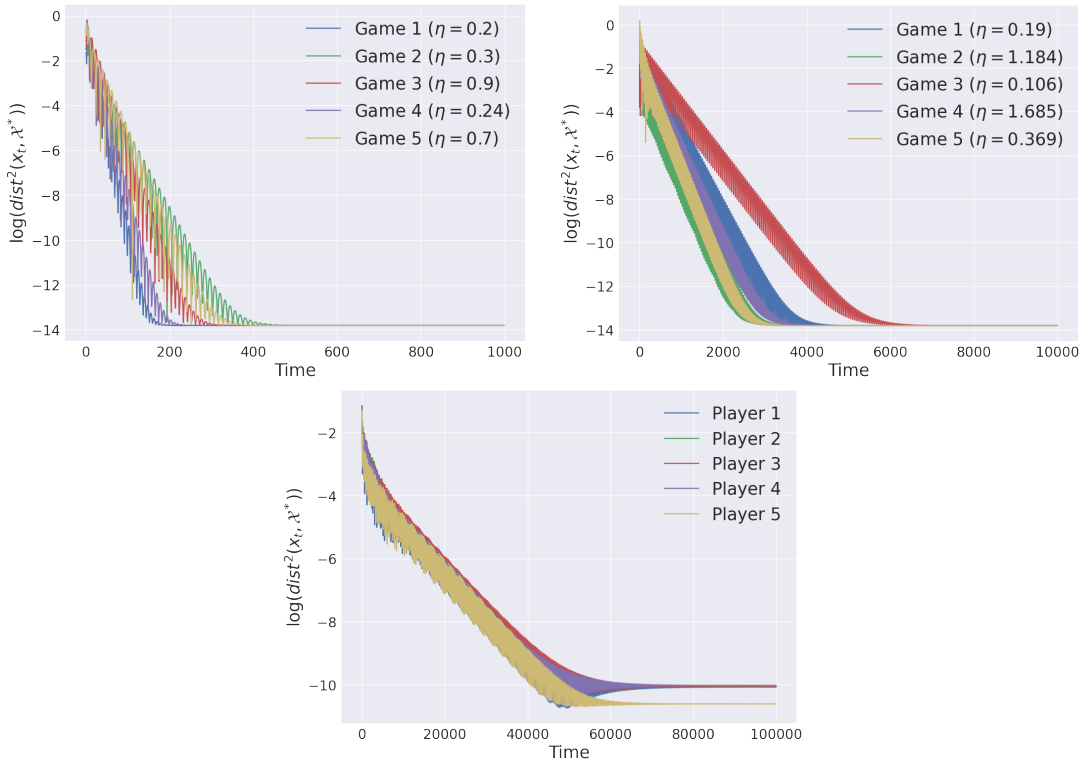


FIGURE 6.6: Last-iterate convergence of OGA to the NE in network zero-sum extensive-form games. The plots shown are: (a) 3-node randomly generated network zero-sum extensive-form game. (b) 4-node random network zero-sum extensive-form game. Note the significantly longer time needed to achieve convergence compared to the 3-node experiment. (c) 5-node *Kuhn poker* game.

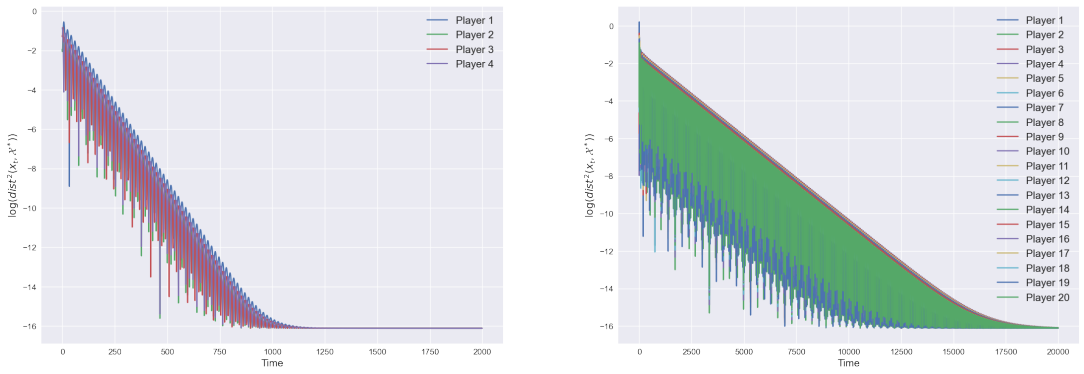


FIGURE 6.7: Simulations using OGA in network Matching Pennies games. (Left) Convergence times for 4-player game; (Right) Convergence times for 20-player game.

6.6 Conclusion

In this chapter, we formulated a class of games called *Network Zero-Sum Extensive-Form Games*, which encode the setting where multiple players compete in pairwise games over a set of resources, defined on a graph. We analyze the convergence properties of *Optimistic Gradient Ascent* in this setting, proving that `OGA` results in both time-average and last-iterate convergence to the set of Nash Equilibria. In order to show this, we utilize a transformation from network zero-sum extensive-form games to two-player symmetric games and subsequently show the convergence results in the symmetric game setting.

This work represents an initial foray into the world of online learning dynamics in network extensive-form games. As the empirical results suggest, however, there remains a wide gap between convergence results in theory and practice. With the recent success stories of extensive-form game solving using modern AI techniques, our hope is that theoretical analysis of this class of games will continue, and provide intuition and insight into the behavior of more complex algorithms in the literature. Moreover, we hope to study how the scalability of online learning algorithms can be improved in the network extensive-form game setting and beyond.

Chapter 7

Clairvoyance: A New Paradigm for Learning in General-Sum Games

This chapter is taken from (with minor modifications) our paper *'Beyond Time-Average Convergence: Near-Optimal Uncoupled Online Learning via Clairvoyant Multiplicative Weights Update'* [139].

7.1 Introduction

In previous chapters, we have focused on modifying well-studied online learning algorithms such as `MWU` and `GDA` to utilize in interesting and new settings. However, there is a deeper question about what it means to guarantee low regret using such techniques. Typically, when players use uncoupled online learning dynamics, they have a simple update rule and directly take the updates at each timestep and choose their next strategy accordingly. This is considered almost a fundamental property of online learning – that the values which are used to derive time-averaged CCEs are taken at each timestep. In the final chapter of this dissertation, we present a work that shifts this paradigm towards a broader understanding of *deduction rules* that go beyond simply taking the time average of all updates. Our core idea is simple: what if instead of immediately playing the computed update at each timestep, the players instead projected forward in their minds and ran a short 'simulation' of sorts before selecting their strategy at the next timestep? We call this concept 'clairvoyance', since players are able to update their strategies with a prediction of what all other players will do next.

In principle, this concept is akin to players sharing a mental model of the world, and updating their strategies only after they have performed a comprehensive internal analysis of the state of the world (i.e., what the other players have played in the previous timestep, the payoff obtained in the previous timestep and so on). At first glance, this might seem to violate the uncoupled nature of standard online learning dynamics such as `MWU`. However, in this chapter we will show that clairvoyant algorithms can be modified into uncoupled/decentralized learning dynamics, which then allows us to explore the notion of an uncoupled modification to `MWU` which can exhibits strong performance guarantees.

The connection between online learning and game theory has been extensively studied for decades. The emergence of online learning dynamics guaranteeing comparable payoffs with the highest-rewarding strategy has provided a landmark method of understanding how selfish agents can act in an adversarial environment, while the notion of *Coarse Correlated Equilibrium* (CCE) has provided a game-theoretic characterization of the limiting behavior of such *online learning dynamics* [19, 76, 78, 145].

An important aspect of online learning dynamics is that players can collectively learn a CCE without having explicit knowledge of the full game description. By definition, these dynamics take as input the sequence of vectors corresponding to the payoff of each possible action, making no assumption about how these payoff vectors are derived [80]. This idea of information exchange can be concisely described with the following decentralized/distributed protocol, as we have seen in prior chapters. This framework is known in the literature as *uncoupled online learning dynamics* [46]:

(1) No player is aware of their payoff matrix or the payoff matrix of any other player.

(2) At each round t, each player i announces their mixed strategy x_i^t.

(3) Based on the announced strategy vector $x^t := (x_1^t, \ldots, x_n^t)$, each player i **learns only** their payoff vector $u_i(x_{-i}^t)$.

(4) Each player i uses $(u_i(x_{-i}^0), \ldots, u_i(x_{-i}^t))$ to update their mixed strategy at round $t+1$.

Uncoupled online learning dynamics exhibit several interesting algorithmic properties: $i)$ their decentralized nature permits efficient distributed implementation, $ii)$ the implicit access to the game via payoff vectors permits game-abstractions with vastly reduced size, and $iii)$ their iterative guarantees can be preferable over the all-or-nothing guarantees of linear programs. A notable example of their algorithmic success is the design of state-of-the art AI poker programs based on iterative self-play that outperform previous LP-based approaches, and are able to compete with human professionals [28, 165, 188]. We remark that this result has motivated a parallel line of research studying the convergence rates of uncoupled online learning dynamics for *extensive form games* [34, 57, 59, 60, 96], and for which we also have shown results in the *network* extensive-form game setting (see Chapter 6).

To reiterate standard results from the literature, if players use a classical *no-regret* online learning algorithm (e.g. Hedge, Regret Matching, `MWU` etc), then the time-average behavior of the resulting uncoupled dynamics will converge to the set of CCE with rate $\Theta(1/\sqrt{T})$ [35]. To be precise, $\Theta(1/\sqrt{T})$ has been shown to be the optimal time-average regret that a player can achieve in the standard adversarial setting, where an adversary can select any sequence of actions/strategies so as to maximize the player's losses.

However, the question of determining which update rule gives the fastest convergence to CCE in a more general setting where adversarial no-regret is not necessary has remained open. Over the years, a series of works has proposed update rules with better and better rates [35, 38, 46, 86, 142, 164], up until the recent seminal work of [47], which established that *Optimistic Multiplicative Weights Update* (`OMWU`) admits $\mathcal{O}(\log^4 T/T)$ rate of convergence to CCE in general-sum games. This rate matches (up to logarithmic

factors) the lower bound of $\Omega(1/T)$ on the convergence rate of any uncoupled online learning dynamic [46].

7.1.1 Our Contributions

All previous works in this area couple the goals of minimizing adversarial regret and fast time-average convergence to CCE [35, 38, 46, 49, 86, 142, 164]. Since many of the aforementioned applications are in the realm of self-play where all agents (equivalently, players) are programmed to follow the update rule, guaranteeing adversarial no-regret is not a necessity. Indeed, it is not clear why time-average behavior should be the only way to deduce a CCE. In fact, any simple and efficient deduction rule would serve the algorithmic benefits of *uncoupled online learning dynamics* [60, 96].

Motivated by the above, we introduce a novel update rule called Clairvoyant Multiplicative Weights Update (`CMWU`) that produces sequences of strategy profiles with constant regret in general normal-form games. In its generic form, `CMWU` is a centralized update rule and does *not* fit in the online learning framework. However, based on `CMWU`, we design an *uncoupled* online learning dynamic called `CMWU` dynamics which gives fast convergence to CCE beyond the time-average sense. More precisely, we establish that given any trajectory of length T of `CMWU` dynamics, its $\log(T)$-sparse sub-trajectory always admits constant regret for all players. As a result, the time-average behavior of the $\log T$-sparse sub-trajectory converges to CCE with rate $\Theta\left(\log(T)/T\right)$, improving on the previous state-of-the-art $\Theta\left(\log^4 T/T\right)$ achieved by [47]. The update rule of `CMWU` dynamics (presented in Algorithm 3) admits a simple form and is an efficient implementation (requiring only a single step of the MWU algorithm at each round). Moreover, the proof of convergence of `CMWU` dynamics is far simpler than previous proofs of convergence of uncoupled dynamics such as `OMWU` [47].

To provide a comparison with the existing learning dynamics in the literature, in Table 7.1, we summarize the most important results concerning the convergence to CCE of uncoupled online learning dynamics alongside `CMWU` dynamics.

Remark 7.1.1. *All the results mentioned in Table 7.1 additionally admit $\tilde{\mathcal{O}}(\sqrt{T})$ adversarial guarantees, where the $\tilde{\mathcal{O}}$ notation ignores logarithmic factors. As mentioned above, this means that once a subset of the players act adversarially by not following the update rule of the online learning dynamic, the players who do follow the update rule are guaranteed to experience at most $\tilde{O}(\sqrt{T})$ regret. The reason why `CMWU` is able to achieve better guarantees with simpler analysis comes from the fact that it neglects the adversarial no-regret guarantees which are irrelevant in self-play/training settings and focuses on minimizing the regret in specific parts of the sequence.*

Remark 7.1.2. *In another related work, [2] established an online learning dynamic that converges with rate $\Theta(n\log(m)\log^4 T/T)$ to Correlated Equilibria (CE), a subset of Coarse Correlated Equilibria. Finally, following our work, [3] produced time-average convergence to CE via online learning at a rate $\Theta(nm^{5/2}\log T/T)$. In comparison, our dependency on the number of actions m is exponentially smaller at $\Theta(\log(m))$. Furthermore, their update rule is based on self-concordant regularization and thus admits high per-iteration complexity, a point which is mentioned by the authors.*

TABLE 7.1: Prior results for convergence to CCE in uncoupled online learning dynamics. n denotes the number of players, m denotes the number of actions per player and V denotes the maximum value in the game payoff matrix/tensor.

Update Rule	Rate of Convergence	Game Type
MWU	$\mathcal{O}\left(V\sqrt{\log m/T}\right)$ [35]	General-sum
Excessive Gap Technique	$\mathcal{O}\left(V\log m(\log T+\log^{3/2} m)/T\right)$ [46]	2-player zero-sum
DS-OptMD, OptDA	$\log^{\mathcal{O}(V)}(m)/T$ [86]	2-player zero-sum
OMWU	$\mathcal{O}\left(V\log m\sqrt{n}/T^{3/4}\right)$ [142, 164]	General-sum
OMWU	$\mathcal{O}\left(V\log^{5/6} m/T^{5/6}\right)$ [38]	General-sum
OMWU	$\mathcal{O}\left(nV\log m\log^4 T/T\right)$ [47]	General-sum
CMWU	$\mathcal{O}\left(nV\log m\log T/T\right)$ (**Theorem** 7.5.1)	General-sum

7.2 The Philosophy and Design of CMWU

At its core, our result relies upon a fundamental but underappreciated difference between online learning (as a general concept) and online learning in games – while online learning relates to the study of open loop systems (i.e., algorithms), learning in games relates to the study of closed loop systems (i.e., dynamical systems). In the former, nothing can be inferred about the future states of the system, which may evolve arbitrarily based on exogenous factors. In the latter, the future cannot evolve entirely arbitrarily, as it is a function of its current state. In principle, this makes online learning in games more predictable.

This realization has recently been explored in a class of learning dynamics that are known as 'optimistic' [49, 142]. This class of dynamics makes the optimistic assumption that the state of the system in the next timestep will be identical to the current timestep, resulting in a slight recency bias in the learning behavior. In Chapter 6, we studied Optimistic Gradient Descent-Ascent in the context of multiplayer extensive-form games. Moreover, optimism can apply to many other standard learning dynamics for games. For instance, Optimistic Multiplicative Weights Update (OMWU) (also referred to as Optimistic Hedge in the literature), is a variant of MWU with the sole difference that the payoff contributions of the last timestep are taken into account twice. Such dynamics and numerous variants thereof have enjoyed widespread adoption in recent years, showing strong performance gains over their non-optimistic counterparts in various settings (e.g. [2, 9, 47, 49, 58, 59, 68, 72, 86, 119, 123, 177, 178]).

Unfortunately, although optimistic dynamics can effectively fuse together predictability and exploitability, they suffer from an intrinsic weakness. The assumption made by optimistic dynamics about the state of the system in the next period is only correct if there

is *no* adaptation/exploitation on the part of the agents. Thus, a tension is introduced between predictability (which decreases with the step-size) and exploitability (which increases with the step-size), resulting in growing performance losses over time.

To deal with this weakness, we introduce a different philosophy in the design of online learning dynamics in normal-form games. We shift away from the prevailing paradigm by defining a novel algorithm that we call *Clairvoyant Multiplicative Weights Update* (CMWU). CMWU is MWU equipped with a mental/simulated model (jointly shared across all players) about the state of the system in its next period. Each player records their mixed strategy, i.e., their belief about what they expect to play in the next period in this shared mental model, which is then internally updated using MWU without any changes to the real-world behavior up until the internal system equilibrates, thus indicating its consistency with the real-world outcome in the next time-step. It is then, and only then, that players take action in the real-world, effectively doing so with "full knowledge" of the system state on the next day, i.e., they are clairvoyant. CMWU acts like MWU with one day look-ahead, achieving bounded (i.e. constant) time-average regret. CMWU is closely related with the *Proximal Point Method* (PPM) [115, 125, 131, 144] which is an implicit (and therefore unimplementable) method that admits arbitrarily fast convergence in the context of convex minimization. Similarly to PPM, in order to implement the update rule of CMWU one would require access to the explicit description of the game and to additionally solve a *fixed-point problem*.

Our main technical contribution thus consists of establishing that for sufficiently small step-sizes, the update rule of CMWU can be computed via the iterations of a contraction map. This not only provides a way to compute the CMWU update rule using a simple and efficient process, but also opens a pathway for *decentralized computation* in the context of uncoupled online learning dynamics. The basic idea behind CMWU dynamics (Algorithm 3) is to punctuate the history of play with *anchor points* which are equally spaced in distance $\Theta(\log(T))$. The key idea at play is that any given anchor point is the CMWU update of the previous anchor point. In this way, the regret in the anchor sequence is constant for any player. Moreover, the intermediate points between two anchor points correspond to the iterations of the contraction map. Surprisingly enough, all of the above can be implemented with just an *if condition* and an MWU-esque exponentiation (see Algorithm 3).

7.3 Preliminaries and Definitions

In addition to the basic definitions of game theory and online learning from Chapters 2 and 2.2, in this chapter we clarify some notation. A general finite normal-form game $\Gamma \equiv \Gamma(\mathcal{N}, \mathcal{S}, u)$ consists of a set of players $\mathcal{N} = \{1, ..., n\}$ where player i may select from a finite set of actions or pure strategies $\mathcal{S}_i$, and each player has a payoff function $u_i : \mathcal{S} \equiv \prod_i \mathcal{S}_i \to \mathbb{R}$ assigning reward $u_i(s)$ to player i. It is common to describe u_i with a payoff tensor $A^{(i)}$ where $u_i(s) = A^{(i)}_s$. Let $m = \max_i |\mathcal{S}_i|$ denote the maximum number of pure strategies in Γ, and let $V = \max_{i,s} |A^{(i)}_s|$ denote the maximum payoff value of any strategy in Γ.

Players are also allowed to use mixed strategies $x_i = (x_{is_i})_{s_i \in \mathcal{S}_i} \in \Delta(\mathcal{S}_i) \equiv \mathcal{X}_i$. The set of mixed strategy profiles is $\mathcal{X} = \prod_i \mathcal{X}_i$. A strategy is fully mixed if $x_{is_i} > 0$ for all $s_i \in \mathcal{S}_i$ and $i \in \mathcal{N}$. Individuals compute the payoff of a mixed strategy linearly using expectation. Formally,

$$u_i(x) = \sum_{s \in \mathcal{S}} u_i(s) \prod_{i \in \mathcal{N}} x_{is_i}. \tag{7.1}$$

We also use the following notation to express player payouts for brevity in our analysis. Let $v_{is_i}(x) = u_i(s_i; x_{-i})$[1] denote the reward i receives if i opts to play pure strategy s_i when everyone else commits to their strategies described by x. This results in $u_i(x) = \langle v_i(x_{-i}), x_i \rangle$. Next, the CCE as defined earlier is given by:

Definition 7.3.1. *A probability distribution μ over pure strategy profiles $s = (s_1, \ldots, s_n) \in \mathcal{S}$ is called an ϵ-approximate Coarse Correlated Equilibrium if for each agent $i \in [n]$,*

$$\mathrm{E}_{s \sim \mu}\left[u_i(s)\right] \geq \mathrm{E}_{s \sim \mu}\left[u_i(s'_i, s_{-i})\right] - \epsilon \quad \textit{for all actions } s_i \in \mathcal{S}_i$$

Definition 7.3.2. *Given a strategy profile $x := (x_1, \ldots, x_n) \in \mathcal{X}$, μ_x denotes the product probability distribution over strategy profiles $s = (s_1, \ldots, s_n) \in \mathcal{S}$ induced by x, $\mu_x(s) := \Pi_{s_i \in s} x_{is_i}$*

We study games from a learning perspective where players iteratively update their mixed strategies over time based on the performance of pure strategies in prior iterations via an uncoupled, online adaptive algorithm. In Algorithm 2, we present a generic description of uncoupled online learning dynamics.

Algorithm 2 Uncoupled Online Learning Dynamics

1: **for** round $t = 0, \cdots, T-1$ **do**
2: Each player $i \in [n]$, **broadcasts** their mixed strategy $x_i^t \in \mathcal{X}_i$.
3: Each player $i \in [n]$, **learns only** their reward vector $u_i(x_{-i}^t)$.
4: Each player $i \in [n]$, **updates** $x_i^{t+1} \in \mathcal{X}_i$ based only on $u_i(x_{-i}^0), \ldots, u_i(x_{-i}^t)$.
5: **end for**

For example, if the update rule of `MWU` is implemented at Step 4 of Algorithm 2, then the distribution $\hat{\mu} := \sum_{t=0}^{T-1} \mu_{x_t}/T$ is an $(1/\sqrt{T})$-approximate CCE.

7.4 Clairvoyant Multiplicative Weights Update

In this section, we introduce a novel learning algorithm for games that we call Clairvoyant Multiplicative Weights Updates (`CMWU`). Critically, `CMWU`, unlike `MWU`, forms self-confirming predictions/beliefs about what all opponents will play in the next time instance. Namely, all players will form the same belief about what player i will play in the next period $t+1$ (i.e. mixed strategy x_i^{t+1}). These beliefs/estimates are such

[1] $(s_i; x_{-i})$ denotes the strategy x after replacing x_i with s_i.

that when players simulate an extra period of play in their mind and update their current strategies using MWU, the resulting strategy for each player i is x_i^{t+1}. All players accurately *predict* the behavior of all other player in the next timestep, in other words they are *clairvoyant*!

The update rule for (CMWU) is as follows:

$$x_{is_i}^{t+1} = \frac{x_{is_i}^t \exp\big\{(\eta_i \cdot v_{is_i}(x^{t+1})\big\})}{\sum_{\bar{s}_i \in \mathcal{S}_i} x_{i\bar{s}_i}^t \exp\{(\eta_i \cdot v_{i\bar{s}_i}(x^{t+1}))\}} \tag{CMWU}$$

Notice that CMWU is an implicit method – the new strategy x^{t+1} appears on both sides of the equation, and thus the method needs to solve an algebraic/fixed point equation for the unknown x^{t+1}.

Theorem 7.4.1. *The algebraic system of equations in* CMWU *defined by an arbitrary game* Γ*, an arbitrary tuple of learning rates* η_i*, and any state* x^t*, always admits a solution.*

Theorem 7.4.1 follows directly by Brouwer's fixed-point theorem, which states that any continuous function that maps a nonempty compact convex set to itself always has a solution. Having established the fact that the players can always *collectively compute* a next step of the CMWU update rule, we present the remarkable property of CMWU stated in Theorem 7.4.2.

Theorem 7.4.2. *Let* $x_0, \ldots, x_{T-1}$ *be a sequence of mixed strategies such that all pairs of consecutive mixed strategies* (x_t, x_{t+1}) *satisfy Equation* (CMWU)*. Then, each player* i *has bounded regret* $\leq \log |S_i|/\eta_i$*.*

The proof of Theorem 7.4.2 follows by standard arguments in online learning literature (e.g. Lemma 5.4 in [80]). We include a version of the proof below for completeness.

Proof. By its construction, (CMWU) is equivalent to FTRL adapted to the fact that on day $\tau = 1, \ldots, t$ it receives as input the utilities of the following day, $\tau + 1 = 2, \ldots, t + 1$, and with entropic regularizer. Astute readers will note that this essentially makes it a variant of the 'Be-The-Leader' algorithm which is known to have non-positive regret. Indeed, the only source of regret of CMWU is due to the additional period's regularization costs. In other words, (CMWU) solves the problem:

$$x_i^{t+1} = \operatorname*{argmax}_{x_i \in \mathcal{X}_i} \left\{ -\frac{h_i(x_i)}{\eta_i} + \left\langle \sum_{\tau=2}^{t+1} v_i(x^\tau), x_i \right\rangle \right\}$$

where $h_i(x_i) = -\sum_i x_i \log x_i$ is the negative entropy regularizer.

By defining the concave payoff $-\frac{h_i}{\eta_i}(x_i)$ as the day 1 payoff function of player i, we have that via a straightforward induction argument that their regret is non-positive. (CMWU), however, has to account for the fact that it does actually include the regularization payoffs in its regret calculations.

$$
\begin{aligned}
\mathit{Regret}_{\text{CMWU}}(t+1) &= \max_{x_i \in \mathcal{X}_i} \left\{ \left\langle \sum_{\tau=2}^{t+1} v_i(x^\tau), x_i \right\rangle \right\} - \sum_{\tau=2}^{t+1} x_i^\tau \cdot v_i(x^\tau) \\
&\leq \max_{x_i \in \mathcal{X}_i} \left\{ -\frac{h_i(x_i)}{\eta_i} + \left\langle \sum_{\tau=2}^{t+1} v_i(x^\tau), x_i \right\rangle \right\} \\
&\quad - \left(-\frac{h_i(x_i^1)}{\eta_i} + \sum_{\tau=2}^{t+1} x_i^\tau \cdot v_i(x^\tau) \right) \\
&\quad + \frac{\max_{x_i \in \mathcal{X}_i} h_i(x_i) - \min_{x_i \in \mathcal{X}_i} h_i(x_i)}{\eta_i} \\
&\leq \frac{\max_{x_i \in \mathcal{X}_i} h_i(x_i) - \min_{x_i \in \mathcal{X}_i} h_i(x_i)}{\eta_i}
\end{aligned}
$$

Then, setting $h_i(x_i) = -\sum_i x_i \log x_i$, we obtain that the regret for each player is bounded by $\frac{\log |S_i|}{\eta_i}$, completing the proof. □

Using the above result, we directly obtain the following corollary:

Corollary 7.4.1. *Let $x_0, \ldots, x_{T-1}$ be a sequence of mixed strategies such that all pairs of consecutive mixed strategies (x_t, x_{t+1}) satisfy Equation* (CMWU)*. Then, the probability distribution $\hat{\mu} := \sum_{t=0}^{T-1} \mu_{x^t}/T$ is a $(\eta \log m/T)$-approximate CCE where $\eta := \min_{i \in n} \eta_i$.*

From a computational complexity perspective, solving the algebraic/fixed point equation (CMWU) is generally a formally hard problem. For instance, the computation of a Nash Equilibrium [128] which has proven to be PPAD-complete [48], reduces to the computation of a solution for equation (CMWU) when $\eta \to \infty$. Moreover, even if we assume that the players possess unlimited computational power, it is not clear how they can efficiently compute a solution to Equation (CMWU) by utilizing an *uncoupled online learning dynamic*.

In Section 7.4.1, we show that if each η_i is upper-bounded by some game-dependent parameters, (CMWU) is a contraction map (Theorem 7.4.3, Corollary 7.4.2). This not only implies that there exists a unique fixed-point solution that can be computed efficiently, but also that the CMWU update rule can be simulated via an uncoupled online learning dynamic.

7.4.1 Uniqueness of Fixed Point via Map Contraction

We will establish uniqueness of the fixed point in CMWU for a specific range of step-sizes. The proof is based on an application of the Banach fixed-point theorem (mapping theorem or contraction mapping theorem) [15]. Thus, we also simultaneously provide a constructive method to compute these fixed points with linear convergence rate.

In what follows, it will be useful to consider vanilla MWU (see Chapter 2.2.2) as a map from a vector of payoffs $v_i = (v_{i1}, \ldots, v_{i|S_i|})$ to mixed strategies, parameterized by the current strategies x_i^t:

$$f_{x_i^t}(v_i) \coloneqq \left(\frac{x_{i1}^t \exp\{(\eta_i \cdot v_{i1})\})}{\sum_{\bar{s}_i \in \mathcal{S}_i} x_{i\bar{s}_i}^t \exp\{(\eta_i \cdot v_{i\bar{s}_i})\}}, \ldots, \frac{x_{i|\mathcal{S}_i|}^t \exp\{(\eta_i \cdot v_{i|\mathcal{S}_i|}\})}{\sum_{\bar{s}_i \in \mathcal{S}_i} x_{i\bar{s}_i}^t \exp\{(\eta_i \cdot v_{i\bar{s}_i})\}} \right) \qquad (\texttt{MWU}_f)$$

We first establish the fact that $f_{x_i^t}(v_i)$ is a $2\eta_i$-continuous mapping.

Lemma 7.4.1. *For any choice of $x_i^t \in \Delta(\mathcal{S}_i)$, the ($\texttt{MWU}_f$) map $f_{x_i^t} : \mathbb{R}^{|S_i|} \to \Delta(\mathcal{S}_i)$ satisfies that for any utility vectors $v_i, v_i' \in \mathbb{R}^{|S_i|}$,*

$$\left\| f_{x_i^t}(v_i) - f_{x_i^t}(v_i') \right\|_1 \leq 2\eta_i \left\| v_i - v_i' \right\|_\infty$$

Proof. To simplify notation we drop the dependence on x_i^t and denote with $f_{s_i}(v_i)$ the s_i coordinate of $f_{x_i^t}(v_i)$. Notice that for any $s_i, s_i' \in S_i$ with $s_i \neq s_i'$ we have:

$$\begin{aligned} \frac{\partial f_{s_i}}{\partial v_{is_i}} &= \eta_i \frac{x_{is_i}^t \exp\{(\eta_i \cdot v_{is_i}\})\Big(\sum_{\bar{s}_i \in \mathcal{S}_i} x_{i\bar{s}_i}^t \exp\{(\eta_i \cdot v_{i\bar{s}_i})\}\Big)}{\Big(\sum_{\bar{s}_i \in \mathcal{S}_i} x_{i\bar{s}_i}^t \exp\{(\eta_i \cdot v_{i\bar{s}_i})\}\Big)^2} - \frac{\Big(x_{is_i}^t \exp\{(\eta_i \cdot v_{is_i}\})\Big)^2}{\Big(\sum_{\bar{s}_i \in \mathcal{S}_i} x_{i\bar{s}_i}^t \exp\{(\eta_i \cdot v_{i\bar{s}_i})\}\Big)^2} \\ &= \eta_i x_{is_i}^{t+1}(1 - x_{is_i}^{t+1}) \end{aligned}$$

Moreover,

$$\begin{aligned} \frac{\partial f_{s_i}}{\partial v_{is_i'}} &= -\eta_i \frac{x_{is_i}^t \exp\{(\eta_i \cdot v_{is_i}\}) x_{is_i'}^t \exp\Big\{(\eta_i \cdot v_{is_i'}\Big\})}{\Big(\sum_{\bar{s}_i \in \mathcal{S}_i} x_{i\bar{s}_i}^t \exp\{(\eta_i \cdot v_{i\bar{s}_i})\}\Big)^2} \\ &= -\eta_i x_{is_i}^{t+1} x_{is_i'}^{t+1} \end{aligned}$$

It follows that $\|\nabla f_{s_i}(v_i)\|_1 = 2\eta_i x_{is_i}^{t+1} \cdot (1 - x_{is_i}^{t+1}) \leq 2\eta_i x_{is_i}^{t+1}$ and thus $\sum_{s_i \in \mathcal{S}_i} \|\nabla f_{s_i}(v_i)\|_1 \leq 2\eta_i$.

$$\begin{aligned} \left\| f_{x_i^t}(v_i) - f_{x_i^t}(v_i') \right\|_1 &= \sum_{s_i \in \mathcal{S}_i} |f_{s_i}(v_i) - f_{s_i}(v_i')| \\ &= \sum_{s_i \in \mathcal{S}_i} \Big| \int_{t=0}^1 \left\langle \nabla f_{s_i}\left((1-t)v_i + tv_i'\right), v_i - v_i' \right\rangle \partial t \Big| \\ &\leq \sum_{s_i \in \mathcal{S}_i} \int_{t=0}^1 \left| \left\langle \nabla f_{s_i}\left((1-t)v_i + tv_i'\right), v_i - v_i' \right\rangle \right| \partial t \\ &\leq \int_{t=0}^1 \Big(\sum_{s_i \in \mathcal{S}_i} \left\| \nabla f_{s_i}\left((1-t)v_i + tv_i'\right) \right\|_1 \Big) \cdot \left\| v_i - v_i' \right\|_\infty \partial t \\ &\leq 2\eta_i \left\| v_i - v_i' \right\|_\infty \end{aligned}$$

□

In the rest of the section we establish that once all η_i are selected sufficiently small then Equation $\texttt{MWU}_f$ admits a unique fixed point and is in fact a contraction map.

Definition 7.4.1. *The distance between the strategy profiles $x = (x_1, \ldots, x_n) \in \mathcal{X}$ and $x' = (x'_1, \ldots, x'_n) \in \mathcal{X}$ is defined as $\delta(x, x') := \max_{1 \le i \le n} \|x_i - x'_i\|_1$.*

In mathematics and geometry, a mapping $f : \mathcal{X} \to \mathcal{X}$ is called a contraction mapping if there exists a constant $0 \le c < 1$ such that $\delta(f(x), f(x')) \le c \cdot \delta(x, x')$ for all $x, x' \in \mathcal{X}$. Intuitively, a contraction mapping brings points closer together over time. Moreover, the contraction mapping theorem states that a strict contraction on a complete metric space has a unique fixed point.

In Theorem 7.4.3 we show that the computation of CMWU is a contraction map if all step-sizes η_i do not exceed a game-dependent constant.

Theorem 7.4.3. *Consider the mixed strategy profile $(x_1^t, x_2^t, \ldots, x_n^t)$ and the map $G : \mathcal{X} \mapsto \mathcal{X}$ defined as follows:*

$$G(x) \coloneqq \left(f_{x_1^t}(v_1(x)), \ldots, f_{x_n^t}(v_n(x)) \right)$$

Then for any $x, x' \in \mathcal{X}$,

$$\delta(G(x), G(x')) \le \eta V (n-1) \cdot \delta(x, x')$$

where η is the maximum step-size over all players, $[0, V]$ is the payoff range of the game and n is the number of players.

Proof. Let us denote by $d_{\mathrm{TV}}(x, x')$ the total variation distance between product distributions x, x'. Then, following the definition of distance in Definition 7.4.1:

$$\begin{aligned}
\delta(G(x), G(x')) &= \left\| f_{x_i^t}(v_i(x)) - f_{x_i^t}(v_i(x')) \right\|_1 \\
&\le 2\eta \cdot \left\| v_i(x) - v_i(x') \right\|_\infty && (7.2) \\
&\le 2\eta V \cdot d_{\mathrm{TV}}(x_{-i}, x'_{-i}) \\
&\le 2\eta V \cdot \textstyle\sum_{j \ne i} d_{\mathrm{TV}}(x_j, x'_j) && (7.3) \\
&= \eta V \cdot \textstyle\sum_{j \ne i} \left\| x_j - x'_j \right\|_1 \\
&\le \eta V (n-1) \cdot \delta(x, x')
\end{aligned}$$

The first inequality (7.2) is obtained via Lemma (7.4.1). The inequality (7.3) can be derived using known properties of total variation (e.g., [83]). □

Corollary 7.4.2. *The (MWU_f) map with maximum step-size $\eta < \frac{1}{(n-1)V}$ is a contraction and thus converges to its unique fixed point at a linear rate.*

To summarize, we have described several results that elucidate the effectiveness of CMWU in converging to approximate CCEs, by first showing that the regret of each player is bounded, and that the CMWU update always admits a unique fixed point via a contraction mapping argument.

7.5 Uncoupled CMWU Online Learning Dynamics

In this section we present an uncoupled online learning dynamic based on the CMWU update rule, which we call CMWU dynamics. In order to ensure that the algorithm is uncoupled/decentralized, the players follow the distributed protocol described in Algorithm 2. Additionally, each player i additionally runs Algorithm 3 as an internal subroutine to update their strategy x_i^t at each round. To simplify notation in Algorithm 3, we set the number of actions of any player i to be $m := |\mathcal{S}_i|$.

Algorithm 3 Internal Update Rule of CMWU Dynamics

Input: $\eta > 0$, $k \in \mathcal{N}$
$x_i^{-1} \leftarrow (1/m, \dots, 1/m)$ and $z_i^{-1} \leftarrow (1/m, \dots, 1/m)$
for each round $t = 0, \cdots, T-1$ **do**
  if $t \bmod k == 0$ **then**
    $x_i^t \leftarrow x_i^{t-1}$
    Player i **broadcasts** the mixed strategy x_i^t and then **receives** the payoff vector $v_i(x_{-i}^t)$.
    Update z_i^t such that for all $s_i \in \mathcal{S}_i$,

$$z_{is_i}^t \leftarrow \frac{z_{is_i}^{t-1} e^{\eta \cdot v_{is_i}(x_{-i}^t)}}{\sum_{\bar{s}_i \in \mathcal{S}_i} z_{i\bar{s}_i}^{t-1} e^{\eta \cdot v_{i\bar{s}_i}(x_{-i}^t)}}$$

  else
    $z_i^t \leftarrow z_i^{t-1}$
    Update x_i^t such that for all $s_i \in \mathcal{S}_i$,

$$x_{is_i}^t \leftarrow \frac{z_{is_i}^t e^{\eta \cdot v_{is_i}(x_{-i}^{t-1})}}{\sum_{\bar{s}_i \in \mathcal{S}_i} z_{i\bar{s}_i}^t e^{\eta \cdot v_{i\bar{s}_i}(x_{-i}^{t-1})}}$$

    Player i **broadcasts** the mixed strategy x_i^t and then **receives** the payoff vector $v_i(x_{-i}^t)$.
  end if
end for

The primary object of note in Algorithm 3 is the parameter k, which partitions the time horizon into sub-trajectories of size k. Each sub-trajectories is designed so that the final point is the CMWU update of the initial point (and that there are $k-1$ points in between). We call these endpoints 'anchor points'. Moreover, as we have established in Corollary 7.4.2, the updates of the $\texttt{MWU}_f$ mapping constitute a contraction mapping with linear convergence rate to a unique fixed point of the CMWU update. Hence, the idea of Algorithm 3 is that in the intermediate points between each pair of anchor points, players run vanilla MWU in order to arrive at the strategy vector which constitutes the next CMWU update. Then, the update at the subsequent anchor point only requires a vanilla MWU update using this 'clairvoyant' strategy vector as an input.

From a theoretical perspective, we know that the regret of the CMWU update (i.e. at the anchor points) is constant. In Theorem 7.5.1, we establish the regret bound for Algorithm 3 as a whole with an appropriate choice of k, incorporating the regret from the intermediate MWU steps. We show that by selecting $k = \log(T)$, we can guarantee that the $\log T$-sparse sub-trajectory generated by Algorithm 3 converges to the set of CCE with a state-of-the-art rate.

Theorem 7.5.1. *Let $x_0, \ldots, x_{T-1}$ be the strategy vector once each agent internally adopts Algorithm 3 with $\eta = 1/2nV$ and $k = \lceil \log T \rceil$. Then for each agent i,*

$$\sum_{\tau=0}^{T'} \left\langle v_i(x_{-i}^{k\cdot\tau}), x_i^{k\cdot\tau} \right\rangle - \max_{x_i \in \mathcal{X}_i} \sum_{\tau=0}^{T'} \left\langle v_i(x_{-i}^{k\cdot\tau}), x_i \right\rangle \geq -12nV \log m$$

where $T' = \left\lfloor \frac{T-1}{k} \right\rfloor$. Thus, the distribution $\hat{\mu} := \sum_{\tau=0}^{T'} \mu_{x^{k\tau}}/T'$ is a $\Theta\left(nV \log m \log T/T\right)$-approximate CCE.

Proof. Notice that $z_i^t = z_i^{\lfloor t/k \rfloor}$ when $(t \bmod k) \neq 0$ and that

$$z_{is_i}^t \leftarrow \frac{z_{is_i}^{t-k} \cdot e^{\eta v_{is_i}(x_{-i}^t)}}{\sum_{\bar{s}_i \in \mathcal{S}_i} z_{i\bar{s}_i}^{t-k} \cdot e^{\eta v_{i\bar{s}_i}(x_{-i}^t)}} \quad \text{when } (t \bmod k) = 0 \tag{7.4}$$

As a result, the sequence $z_i^0, z_i^k, \ldots, z_i^{k\tau}, \ldots$ is the sequence produced by MWU with *look-ahead* applied to the sequence of reward vectors $v_i(x_{-i}^0), v_i(x_{-i}^k), \ldots, v_i(x_{-i}^{k\tau}), \ldots$. MWU with look-ahead is effectively Be-The-Regularized-Leader, which admits the following regret guarantee (of a similar flavor to Lemma 5.4 in [80]),

$$\sum_{\tau=0}^{T'} \left\langle v_i(x_{-i}^{k\cdot\tau}), z_i^{k\cdot\tau} \right\rangle - \max_{x_i \in \mathcal{X}_i} \sum_{\tau=0}^{T'} \left\langle v_i(x_{-i}^{k\cdot\tau}), x_i \right\rangle \geq -\frac{\log m}{\eta} = -2nV \log m$$

For completeness, we include the full statement of the regret bound for our setting in Lemma 7.5.1:

Lemma 7.5.1. *Consider $z^{-k} = (1/m, \ldots, 1/m)$, the sequence of mixed strategies $z_i^0, z_i^k, \ldots, z_i^{kT'}$ and the sequence of reward vectors $v_i(x_{-i}^0), v_i(x_{-i}^k), \ldots, v_i(x_{-i}^{kT'})$ such that*

$$z_{is_i}^{k\cdot\tau} \leftarrow \frac{z_{is_i}^{k\cdot\tau-k} \cdot e^{\eta v_{is_i}(x_{-i}^{k\cdot\tau})}}{\sum_{\bar{s}_i \in \mathcal{S}_i} z_{i\bar{s}_i}^{k\cdot\tau-k} \cdot e^{\eta v_{i\bar{s}_i}(x_{-i}^{k\cdot\tau})}} \quad \text{for all } \tau \geq 0$$

Then, the following guarantee holds,

$$\sum_{\tau=0}^{T'} \left\langle v_i(x_{-i}^{k\cdot\tau}), z_i^{k\cdot\tau} \right\rangle - \max_{x_i \in \mathcal{X}_i} \sum_{\tau=0}^{T'} \left\langle v_i(x_{-i}^{k\cdot\tau}), x_i \right\rangle \geq -\frac{\log m}{\eta}.$$

Proof of Lemma 7.5.1. To simplify notation, let us define $t := k \cdot \tau$ and $v_i^t := v_i(x_{-i}^{k\cdot\tau})$. It is known that the z_i^t can be equivalently described as:

$$z_i^t = \underset{z_i \in \mathcal{X}_i}{\operatorname{argmax}} \left[\gamma \left\langle \sum_{s=0}^{t} u_i^s, z_i \right\rangle - h(z_i) \right] \text{ for all } t \geq -1$$

where $h(z_i) = -\sum_{s_i} z_{is_i} \log z_{is_i}$ (see Section 5.4.1 in [80]). Now let $g_t(z_i) := \gamma \left\langle \sum_{s=0}^{t} u_i^s, z_i \right\rangle - h(z_i)$ which means that $z_i^t = \operatorname{argmax}_{z_i \in \mathcal{X}_i} g_t(z_i)$ and let $x_i^* := \operatorname{argmax}_{x_i \in \mathcal{X}_i} \sum_{t=0}^{T'} \left\langle v_i^t, x_i \right\rangle$. Using a simple induction argument (identical to that of Lemma 5.4 in [80]), one can easily show that

$$\sum_{t=-1}^{T'} g_t(z_i^t) \geq \sum_{t=-1}^{T'} g_t(x_i^*)$$

which implies that

$$\sum_{\tau=0}^{T'} \left\langle v_i(x_{-i}^t), z_i^t \right\rangle - \sum_{\tau=0}^{T'} \left\langle v_i(x_{-i}^t), x_i^* \right\rangle \geq \frac{h(z^{-1})}{\gamma} - \frac{h(x_i^*)}{\gamma} \geq \frac{\log m}{\gamma}$$

□

Up next, we establish that $\left\| z_i^{k\cdot\tau} - x_i^{k\cdot\tau} \right\|_1 \leq 1/T$ for all $\tau \geq 1$. Let $\tau_m := (\tau - 1) \cdot k + m$. By the definition of Algorithm 3, $x_i^{k\cdot\tau} = x_i^{\tau_k - 1}$ and $z_i^{\tau_m} = z_i^{\tau_0}$ for all $m = 0, \ldots, k-1$. Thus,

$$x_{is_i}^{\tau_m} \leftarrow \frac{z_{is_i}^{\tau_0} e^{\eta v_{is_i}(x_{-i}^{\tau_m - 1})}}{\sum_{\bar{s}_i \in \mathcal{S}_i} z_{i\bar{s}_i}^{\tau_0} e^{\eta v_{i\bar{s}_i}(x_{-i}^{\tau_m - 1})}} \quad \text{for all } s_i \in \mathcal{S}_i.$$

Using Equation $\texttt{MWU}_f$ of Section 7.4.1, the above system of equations can be concisely written as,

$$x_i^{\tau_m} = f_{z_i^{\tau_0}} \left(u_i(x_{-i}^{\tau_m - 1}) \right)$$

and since all players follow Algorithm 3, $x^{\tau_m} = G\left(x^{\tau_m - 1}\right)$ where $G(x) := (f_{z_1^{\tau_0}}(x), \ldots, f_{z_n^{\tau_0}}(x))$. Since $\eta := 1/2nV$ by Theorem 7.4.3, $G(x)$ is a contraction map with constant $1/2$. Thus,

$$\delta\left(G\left(x^{\tau_k - 1}\right), x^{\tau_k - 1}\right) \leq \frac{1}{2^{k-2}} \cdot \delta\left(x^{\tau_1}, x^{\tau_0}\right) \leq \frac{8}{2^k}.$$

The second inequality follows by $\delta\left(x^{\tau_1}, x^{\tau_0}\right) \leq 2$ (see Definition 7.4.1). Recall that $x^{k\cdot\tau} = x^{\tau_k - 1}$ and that $z^{k\cdot\tau} = G\left(x^{\tau_k - 1}\right)$ (Equation 7.5.1). As a result, for each player i

$$\left\| x_i^{k\cdot\tau} - z_i^{k\cdot\tau} \right\|_1 \leq \delta\left(x^{k\cdot\tau}, z^{k\cdot\tau}\right) = \delta\left(x^{\tau_k - 1}, G\left(x^{\tau_k - 1}\right)\right) \leq \frac{8}{2^k}$$

We are now ready to complete the proof of Theorem 7.5.1 as follows:

$$\begin{aligned}
\sum_{\tau=0}^{T'}\langle v_i(x_{-i}^{k\cdot\tau}), x_i^{k\cdot\tau}\rangle &\geq \sum_{\tau=0}^{T'}\langle v_i(x_{-i}^{k\cdot\tau}), z_i^{k\cdot\tau}\rangle - |\langle v_i(x_{-i}^{k\cdot\tau}), x_i^{k\cdot\tau} - z_i^{k\cdot\tau}\rangle| \\
&\geq \sum_{\tau=0}^{T'}\langle v_i(x_{-i}^{k\cdot\tau}), z_i^{k\cdot\tau}\rangle - \sum_{\tau=0}^{T'}\left\|v_i(x_{-i}^{k\cdot\tau})\right\|_\infty \cdot \left\|x_i^{k\cdot\tau} - z_i^{k\cdot\tau})\right\|_1 \\
&\geq \sum_{\tau=0}^{T'}\langle v_i(x_{-i}^{k\cdot\tau}), z_i^{k\cdot\tau}\rangle - 8V\frac{T}{k2^k} - \left\|v_i(x_{-i}^{0})\right\|_\infty \cdot \left\|x_i^{0} - z_i^{0})\right\|_1 \\
&\geq \sum_{\tau=0}^{T'}\langle v_i(x_{-i}^{k\cdot\tau}), z_i^{k\cdot\tau}\rangle - 8V\frac{T}{k2^k} - \left\|v_i(x_{-i}^{0})\right\|_\infty \cdot \left\|x_i^{0} - z_i^{0})\right\|_1 \\
&\geq \max_{x_i\in\mathcal{X}_i}\sum_{\tau=0}^{T'}\langle v_i(x_{-i}^{k\cdot\tau}), x_i\rangle - 2nV\log m - 10V \\
&\geq \max_{x_i\in\mathcal{X}_i}\sum_{\tau=0}^{T'}\langle v_i(x_{-i}^{k\cdot\tau}), x_i\rangle - 12nV\log m
\end{aligned}$$

The statement of the theorem follows. □

7.6 Experimental Results

In this section we present an experimental result that shows the fast convergence of CMWU to CCE. In Figure 7.1 we compare the performance of CMWU dynamics (Algorithm 3) to the current state-of-the-art OMWU with step-sizes selected according to [47]. The game we study is a randomly generated 4-player, 10-strategy normal-form game and in each run, the players' initial conditions are randomly generated. The update rule for OMWU, also referred to as Optimistic Hedge, can be written as

$$x_{is_i}^{t+1} = \frac{x_{is_i}^{t}\exp\{(\eta_i\cdot\left(2v_{is_i}(x^t) - v_{is_i}(x^{t-1})\right)\})}{\sum_{\bar{s}_i\in\mathcal{S}_i} x_{i\bar{s}_i}^{t}\exp\{(\eta_i\cdot\left(2v_{i\bar{s}_i}(x^t) - v_{i\bar{s}_i}(x^{t-1})\right)\}} \tag{OMWU}$$

In order to account for the internal update rule of CMWU dynamics, we run the OMWU experiment for a longer time and compute the regret of the OMWU dynamic only at each $\log(T)$-th iterate. We observe that CMWU dynamics allow for faster computation of CCE than OMWU.

In Figure 7.2, we plot the cumulative regret of CMWU dynamics in a 4-player 10-strategy game when run with various fixed step-size values. As the step-size increases, we note empirically that the time required to compute a CCE decreases.

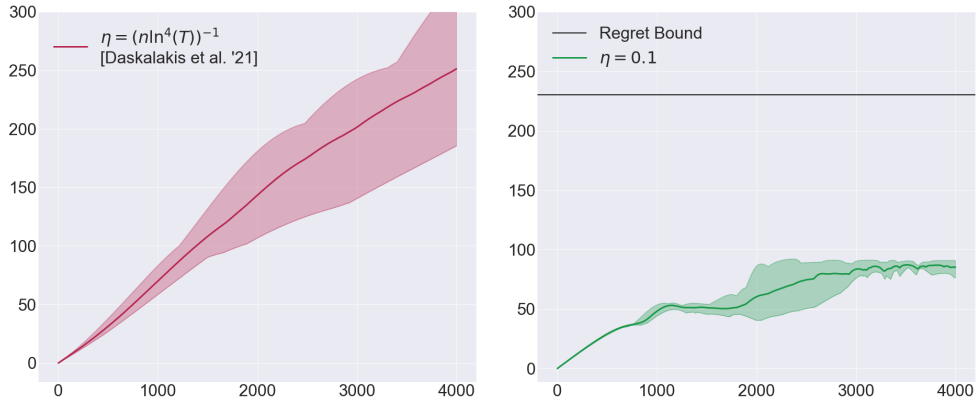


FIGURE 7.1: State-of-the-art OMWU [47] vs. CMWU dynamics in a 4-player 10-strategy game. We plot the max over players' cumulative regret for several common random initializations. The shaded region represents the max/min regret range across runs. CMWU dynamics allow for significantly faster computation of approximate coarse correlated equilibria than OMWU, i.e., it needs significantly fewer oracle calls for the same accuracy level. For a zoom-in on the cumulative regret of CMWU dynamics for larger step-sizes η see Fig. 7.2.

7.7 CMWU Dynamics as an Anytime Algorithm

Our formulation of the internal update rule of CMWU dynamics in Algorithm 3 can also be framed as an anytime algorithm. In computer science, most algorithms are designed to run to completion (i.e., a time horizon is predetermined and the computation is performed until the end of the fixed time horizon). In that sense, the bounded regret property which we have established is valid only if the time horizon is known. However, an anytime (or interruptible) algorithm allows for a relaxation of that requirement - it is able to return a valid solution even if interrupted before the time horizon is reached.

In our setting, we can provide a formulation for an anytime algorithm in the case where the time horizon T is not known in advance and thus the algorithm has to have bounded regret for all T. Typically one can obtain such an anytime algorithm via a doubling trick, but we propose a simple modification of the internal update rule which achieves the same effect in Algorithm 4. Our convergence result of Theorem 7.5.1 can also be extended to the anytime setting, as we show in Theorem 7.7.1.

Theorem 7.7.1. *Let $x_0, \dots, x_{T-1}$ be the strategy vector produced when each player internally adopts Algorithm 4 with $\eta = 1/2nV$. Then for each player i,*

$$\sum_{\tau \in T'} \left\langle v_i(x^\tau_{-i}), x^\tau_i \right\rangle - \max_{x_i \in \mathcal{X}_i} \sum_{\tau \in T'} \left\langle v_i(x^\tau_{-i}), x_i \right\rangle \geq -\mathcal{O}(nV \log m)$$

Moreover $|T'| = \Omega(T/\log T)$ and thus the distribution $\hat{\mu} := \sum_{\tau \in T'} \mu_{x^\tau}/T'$ is a $\mathcal{O}\left(nV \log m \log T/T\right)$-approximate CCE.

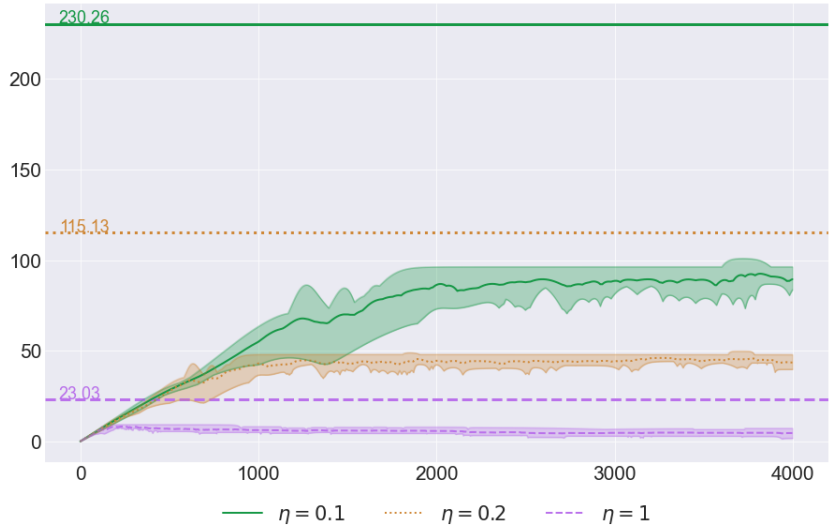


FIGURE 7.2: Zoomed-in cumulative regret over time for the CMWU plot in Figure 7.1 with larger η values. As the learning rate increases, antithetical with other approaches, the speed of convergence of CMWU to CCE increases. Colored horizontal lines represent the respective theoretical regret bounds for each value of η.

Proof. Notice that the set T' is the same for any player i. In order to simplify notation, let us define $T' = \{1, \dots, \tau_{k-1}, \tau_k, \dots, \tau_K\}$. At the same time, note that if Algorithm 4 is run for T time-steps, then $K = \Omega(T/\log T)$. As in the proof of Theorem 7.4.2 we have that for any $\tau_k \in T'$, by definition of Algorithm 4,

$$\left| x_{is_i}^{\tau_k} - \frac{x_{is_i}^{\tau_{k-1}} \cdot e^{\eta v_{is_i}(x_{-i}^{\tau_{k-1}})}}{\sum_{\bar{s}_i \in \mathcal{S}_i} x_{i\bar{s}_i}^{\tau_{k-1}} \cdot e^{\eta v_{i\bar{s}_i}(x_{-i}^{\tau_{k-1}})}} \right| \leq \frac{1}{2^{\tau_k - \tau_{k-1}}} \leq \frac{1}{k^2} \tag{7.5}$$

To simplify notation we rewrite the above inequality as

$$\|x^{\tau_k} - y^{\tau_k}\| \leq 1/k^2$$

where

$$y_{is_i}^{\tau_k} \leftarrow \frac{x_{is_i}^{\tau_{k-1}} \cdot e^{\eta v_{is_i}(x_{-i}^{\tau_{k-1}})}}{\sum_{\bar{s}_i \in \mathcal{S}_i} x_{i\bar{s}_i}^{\tau_{k-1}} \cdot e^{\eta v_{i\bar{s}_i}(x_{-i}^{\tau_{k-1}})}}$$

Algorithm 4 Anytime Internal Update Rule of `CMWU` Dynamics

Input: $\eta > 0$
$x_i^0 \leftarrow (1/m, \ldots, 1/m)$ and $z_i^0 \leftarrow (1/m, \ldots, 1/m)$
$T' \leftarrow \{1\}$ and $\tau \leftarrow 0$
for each round $t = 1, \cdots, T-1$ **do**
  if $t == \tau + \log\left(|T'|^2\right)$ **then**
    $x_i^t \leftarrow x_i^{t-1}$
    Player i **broadcasts** the mixed strategy x_i^t and then **receives** the payoff vector $v_i(x_{-i}^t)$.
    Update z_i^t such that for all $s_i \in \mathcal{S}_i$,

$$z_{is_i}^t \leftarrow \frac{z_{is_i}^{t-1} e^{\eta \cdot v_{is_i}(x_{-i}^t)}}{\sum_{\bar{s}_i \in \mathcal{S}_i} z_{i\bar{s}_i}^{t-1} e^{\eta \cdot v_{i\bar{s}_i}(x_{-i}^t)}}$$

    $T' \leftarrow T' \cup \{t\}$ and $\tau \leftarrow t$
  else
    $z_i^t \leftarrow z_i^{t-1}$
    Update x_i^t such that for all $s_i \in \mathcal{S}_i$,

$$x_{is_i}^t \leftarrow \frac{z_{is_i}^t e^{\eta \cdot v_{is_i}(x_{-i}^{t-1})}}{\sum_{\bar{s}_i \in \mathcal{S}_i} z_{i\bar{s}_i}^t e^{\eta \cdot v_{i\bar{s}_i}(x_{-i}^{t-1})}}$$

    Player i **broadcasts** the mixed strategy x_i^t and then **receives** the payoff vector $v_i(x_{-i}^t)$.
  end if
end for

The proof is completed with the exact same argument as in Theorem 7.4.2. More precisely at the final step of the proof we obtain:

$$\begin{aligned}
\sum_{k=1}^{K} \langle v_i(x_{-i}^{\tau_k}), x_i^{\tau_k} \rangle &\geq \sum_{k=1}^{K} \langle v_i(x_{-i}^{\tau_k}), y_i^{\tau_k} \rangle - |\langle v_i(x_{-i}^{\tau_k}), x_i^{\tau_k} - y_i^{\tau_k} \rangle| \\
&\geq \sum_{k=1}^{K} \langle v_i(x_{-i}^{\tau_k}), y_i^{\tau_k} \rangle - \sum_{k=1}^{K} \left\| v_i(x_{-i}^{\tau_k}) \right\|_\infty / k^2 \\
&\geq \sum_{k=1}^{K} \langle v_i(x_{-i}^{\tau_k}), y_i^{\tau_k} \rangle - \mathcal{O}(V) \\
&\geq \max_{x_i \in \mathcal{X}_i} \sum_{k=1}^{K} \langle v_i(x_{-i}^{\tau_k}), x_i \rangle - \mathcal{O}(nV \log m)
\end{aligned}$$

□

7.8 Conclusion

In this chapter, we have introduced and analyzed a novel learning dynamic that achieves the strongest to-date results for time-average convergence to coarse correlated equilibria (CCE) in general normal-form games. We call this method Clairvoyant Multiplicative Weights Update (`CMWU`). Although the definition of `CMWU` requires a fixed-point computation and effectively enables the players to accurately predict and exploit the future behavior of their opponents, we show that it can be implemented efficiently in an online and uncoupled manner, allowing it to be easily incorporated in existing ML systems. A critical conceptual innovation of the uncoupled implementation of `CMWU` is that the CCE computation is performed not via the standard techniques (uniform time averaging or last-iterate implementation) but an averaging of a pre-specified and game independent $\mathcal{O}(\log(T))$-sparse subsequence of the whole history of play. This type of implementation, as far as we are aware, is novel not only from a theoretical but also a practical perspective.

Learning in games has become a fundamental tool for many machine learning applications, ranging from generative adversarial networks to poker and beyond. Despite this wide range of settings, there is a rough pattern in achieving these results - design an optimization-driven learning dynamic and have it compete against itself in self-play. In poker style applications, the typical technique is to approximately solve the zero-sum game via low-regret *time-averaging* (typically uniform, sometimes with recency bias) over the entire history of play. However, in other applications such as Go [158], the policy/strategy encoding is achieved via deep neural networks with many parameters, so any notion of averaging over the history of play becomes impractical. In these settings, the hope is that self-play leads to ever improving players with the solution being the last-iterate of the self-play process.

The introduction of `CMWU` shows that exploring different averaging techniques beyond the time-average and last-iterate regimes enables efficient, uncoupled and state-of-the-art algorithms for solving general normal-form games. This points to a rather under-explored hyper-parameter of online algorithm design, the *averaging policy*, and raises tantalizing questions about extending our results both in theory as well as experimentally to more complex settings.

Chapter 8

Concluding Remarks and Future Work

There's no going back. Nobody goes back the same way they came.

Andrei Tarkovsky, *Stalker*

8.1 Retrospective

While each chapter is accompanied with a discussion of the impact of our results, in this final retrospective we summarize and describe higher-level implications of our body of work. This dissertation explores the behavior of continuous- and discrete-time dynamics in game-theoretic settings beyond the standard, normal-form regime. Our results encompass time-evolving games, quantum games, extensive-form games and general-sum games, showcasing the flexibility of simple classes of online learning dynamics such as `FTRL` as models for the evolution and learning of rational agents. Our results in the case of discrete-time dynamics show that converging quickly in either time-average or last-iterate sense is possible even in settings with multiple competing players. However, our results for continuous-time dynamics also additionally show that the recurrent behavior extends even to time-evolving and quantum games.

If game theory is to be a reasonable model for the intrinsic payoff/cost structures in real-world systems, then equilibrium computation and learning needs to be robust to potential changes to the system over time. We have explored two forms of time-evolution within this dissertation, which capture different models of endogenous and exogenous evolution. However, even in the conceptually simplistic model of periodically-evolving games, the safe haven of time-average equilibrium convergence can fail. This exhibits a fragility of the standard game-theoretic model that requires further investigation. Perhaps a more important line of inquiry has to do with the notion of equilibration studied – indeed, our results only apply to settings where the Nash equilibrium of each period game is invariant. Recent works that are able to achieve efficient learning in various time-evolving game settings suggest that more *dynamic* notions of equilibrium and performance measures are needed to improve the robustness of the theory [1, 54, 118, 185].

Instead of selecting strategies on a simplex as in the classical normal-form setting, quantum game players instead select strategies on a spectraplex. This increased complexity makes the time-average equilibrium convergence and recurrence results particularly interesting. It tells us that learning in games, while fragile in some ways, is also extensible to realms beyond the classically studied regime. While the results for recurrence in quantum games might seem like a mathematical curiosity, there is no doubt the multi-agent quantum computing systems are on the horizon. As such, our results indicate that our intuition for learning behavior in classical games can carry over to quantum games, an observation which has been corroborated by several recent works [108, 109, 111, 112, 171].

Let us go back to the primary motivation of studying online learning in games: game theory emerges as a model of interactions between multiple agents in the real world, and the simple learning rules we study function as a model for how agents might evolve and adapt by repeatedly interacting with each other and the world. The online nature of this adaptation captures situations where agents might be entirely unaware of what other agents are doing, or even what the rules of interaction might be. In view of this, our results elucidate a pertinent tension between theory and practice. Many modern machine learning applications have had a flavor of learning in games. Consider the recent success of reinforcement-learning algorithms based on the counterfactual regret minimization (CFR) framework which are able to outperform even the best human players in games such as Poker, Go, Chess and Shogi [27–29, 157]. Moreover, recently [56] managed to train an AI which can play the cooperative game of strategy Diplomacy at a human level by combining strategic reasoning and natural language processing. These technological advancements arose from the advent of deep learning and the increasing ability to train models on vast amounts of data. Meanwhile, even though our work on network EFGs shows that some convergence results can be shown in multi-player EFGs, proving results for highly complex multi-stage algorithms with massive strategy spaces such as Chess proves to be far less tractable.

Despite this divide between theory and practice, it is clear that the framework of online learning in games provides a good test-bed for computer scientists to i) better understand why certain algorithms may or may not work in practice, and ii) improve individual aspects of the overall training process. As a concrete example, fictitious play, which we discussed in the very first chapter of this dissertation, has been successfully combined with neural network function approximation, resulting in an algorithm that can converge to approximate Nash equilibria [81]. In this sense, one could think of optimistic gradient descent, clairvoyant multiplicative weights and other online learning dynamics as a baseline *conceptual* learning rule which can be used in tandem with deep methods to create truly state-of-the-art algorithms.

The implementation of clairvoyance in an uncoupled learning dynamic raises questions about further modifications to standard online learning dynamics that can outperform standard methods. There is also a strong connection to proximal algorithms in min-max optimization which arise from the proximal point method [129, 144]. Indeed, one can derive a 'clairvoyant' version of the extra-gradient method which achieves near-optimal convergence rates for min-max problems with a much simpler analysis than other algorithms in the literature [36]. With the increasingly complex designs

of state-of-the-art algorithms in machine learning and AI, clairvoyance can thus be thought of as method to modify training methods so that their respective analysis is more tractable.

8.2 Future Work

Continuous- vs. Discrete-Time

Within this dissertation, continuous-time and discrete-time systems are often seen as separate entities which are studied with different tools. However, there is a strong connection between the two regimes, as can be seen in the analysis of Chapter 5. Indeed, there are several recent works which show that certain classes of discrete-time algorithms actually exhibit cycling behavior which is reminiscent of continuous-time dynamics. For instance, [11, 180] showed that alternating versions of discrete-time mirror descent approximates the behavior of continuous-time mirror descent, exhibiting near-optimal regret bounds. A key open question is whether or not one can show recurrence in the discrete-time case. Moreover, there is a deeper question of the discretization technique used – it is known that the symplectic Euler discretization of continuous-time mirror descent gives rise to alternating mirror descent. In light of this, can we utilize the large body of work studying discretization techniques of continuous dynamical systems to design discrete-time algorithms with better regret guarantees?

Learning in Quantum Games

As described earlier, several recent works have expanded upon our initial foray into learning in quantum games. Indeed, many existing results from the realm of classical normal-form games have been extended into the quantum setting. However, the interpretation of this class of games remains a hotly contested topic. The quantum games we study are *non-interactive*, which represents only the initial part of the comprehensive formulation of quantum games by [21]. Because of the non-interactive nature of these systems, the dynamics effectively always stay separable. Despite this, one can interpret the games we study from a semidefinite programming perspective [88]. A key question for future research is whether one can design online learning algorithms which are truly able to capture the complex, entangled nature of general quantum systems.

Clairvoyant Learning

The implementation of clairvoyance to obtain near-optimal learning has seen success in game theory as well as min-max optimization. However, one clear advantage that optimism has over clairvoyance is that last-iterate convergence of optimistic algorithms has been established in many classes of zero-sum games. From a theoretical perspective, it would then be fascinating to explore if it is possible to study the tradeoffs and interplay between optimism and clairvoyance from the perspective of the averaging scheme. In some cases such as convex-concave saddle point problems, time-average convergence of the extra-gradient method can actually be faster than last-iterate convergence [73]. Moreover, it is important to investigate other settings such as multi-agent reinforcement learning in which clairvoyant learning could potentially result in performance

improvements. Thanks to the increased rate of convergence to equilibria which is possible in games, we postulate that deriving clairvoyant modifications of reinforcement learning training processes that rely on self-play could be possible. This is in part due to the game-theoretic approaches to reinforcement learning in the literature [81, 102, 184].

Tractability of Equilibrium Concepts

The final direction for future work is perhaps a more philosophical one. While the Nash equilibrium has long been upheld as an influential solution concept in games, it has been shown to be intractable to compute [37, 48]. Moreover, it even fails as a natural model of the equilibration of decentralized learning [117, 122, 172]. Many different solution concepts have thus arisen in the literature in hopes that they can be reasonable alternatives to the Nash equilibrium in their respective game classes. Even so, the equilibrium selection problem [77] remains a key challenge – decentralized learning agents may not be able to coordinate the equilibrium which they converge to. From the perspective of a mechanism designer, it is thus important to attempt to design systems wherein the choice of equilibrium can be encoded within the learning dynamic itself, so that the players can be guided towards a socially optimal equilibrium. From a practical perspective, it is also paramount to study learning and equilibration in a broader sense inspired by other areas outside of machine learning, such as social choice theory and psychology [52, 183]. Much like how biology conceptually inspired the advent of neural networks in computer science, there is a wealth of knowledge which we can draw from beyond game theory, dynamical systems and optimization.

Bibliography

[1] Ioannis Anagnostides, Ioannis Panageas, Gabriele Farina, and Tuomas Sandholm. "On the convergence of no-regret learning dynamics in time-varying games". In: *Advances in Neural Information Processing Systems* 36 (2024).

[2] Ioannis Anagnostides et al. "Near-Optimal No-Regret Learning for Correlated Equilibria in Multi-Player General-Sum Games". In: *arXiv preprint arXiv:2111.06008* (2021).

[3] Ioannis Anagnostides et al. "Uncoupled Learning Dynamics with O(log T) Swap Regret in Multiplayer Games". In: *CoRR* abs/2204.11417 (2022).

[4] Itai Arieli and Yakov Babichenko. "Random Extensive Form Games". In: *J. Econ. Theory* 166 (2016), pp. 517–535.

[5] Vladimir Igorevich Arnol'd. *Mathematical methods of classical mechanics*. Vol. 60. Springer Science & Business Media, 2013.

[6] Sanjeev Arora, Elad Hazan, and Satyen Kale. "The Multiplicative Weights Update Method: a Meta-Algorithm and Applications". In: *Theory of Computing* 8.1 (2012), pp. 121–164. DOI: 10.4086/toc.2012.v008a006. URL: https://doi.org/10.4086/toc.2012.v008a006.

[7] Sanjeev Arora and Satyen Kale. "A Combinatorial, Primal-Dual Approach to Semidefinite Programs". In: *J. ACM* 63.2 (May 2016). ISSN: 0004-5411. DOI: 10.1145/2837020. URL: https://doi.org/10.1145/2837020.

[8] Robert J Aumann. "Subjectivity and correlation in randomized strategies". In: *Journal of mathematical Economics* 1.1 (1974), pp. 67–96.

[9] Waïss Azizian, Franck Iutzeler, Jérôme Malick, and Panayotis Mertikopoulos. "The last-iterate convergence rate of optimistic mirror descent in stochastic variational inequalities". In: *Conference on Learning Theory*. PMLR. 2021, pp. 326–358.

[10] Paul Bachmann. *Die analytische zahlentheorie*. Vol. 2. Teubner, 1923.

[11] James P Bailey, Gauthier Gidel, and Georgios Piliouras. "Finite regret and cycles with fixed step-size via alternating gradient descent-ascent". In: *Conference on Learning Theory*. 2020, pp. 391–407.

[12] James P Bailey, Sai Ganesh Nagarajan, and Georgios Piliouras. "Stochastic Multiplicative Weights Updates in Zero-Sum Games". In: *arXiv preprint arXiv:2110.02134* (2021).

[13] James P Bailey and Georgios Piliouras. "Multiplicative weights update in zero-sum games". In: *Proceedings of the 2018 ACM Conference on Economics and Computation*. 2018, pp. 321–338.

[14] D Balduzzi et al. "Open-ended learning in symmetric zero-sum games". In: *International Conference on Machine Learning*. Vol. 97. 2019, pp. 434–443.

[15] Stefan Banach. "Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales". In: *Fund. math* 3.1 (1922), pp. 133–181.

[16] Luis Barreira. "Poincaré recurrence: old and new". In: *XIVth International Congress on Mathematical Physics*. World Scientific. 2006, pp. 415–422.

[17] Tamer Başar and Geert Jan Olsder. *Dynamic noncooperative game theory*. SIAM, 1998.

[18] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. "Curriculum learning". In: *International Conference on Machine Learning*. 2009, pp. 41–48.

[19] David Blackwell. "An analog of the minimax theorem for vector payoffs". In: *Pacific Journal of Mathematics* 6 (1956), pp. 1–8.

[20] Victor Boone and Georgios Piliouras. "From Darwin to Poincaré and von Neumann: Recurrence and Cycles in Evolutionary and Algorithmic Game Theory". In: *International Conference on Web and Internet Economics*. Springer. 2019, pp. 85–99.

[21] John Bostanci and John Watrous. "Quantum game theory and the complexity of approximating quantum Nash equilibria". In: *arXiv preprint arXiv:2102.00512* (2021).

[22] Samuel Bowles, Jung-Kyoo Choi, and Astrid Hopfensitz. "The co-evolution of individual behaviors and social institutions". In: *Journal of theoretical biology* 223.2 (2003), pp. 135–147.

[23] Michael Bowling, Neil Burch, Michael Johanson, and Oskari Tammelin. "Heads-up Limit Hold'em Poker is Solved". In: *Science* 347.6218 (2015), pp. 145–149.

[24] LM Bregman and IN Fokin. "On separable non-cooperative zero-sum games". In: *Optimization* 44.1 (1998), pp. 69–84.

[25] Luitzen Egbertus Jan Brouwer. "Über abbildung von mannigfaltigkeiten". In: *Mathematische annalen* 71.1 (1911), pp. 97–115.

[26] G.W. Brown. "Iterative Solutions of Games by Fictitious Play". In: *Activity Analysis of Production and Allocation* (1951), 374–376.

[27] Noam Brown, Adam Lerer, Sam Gross, and Tuomas Sandholm. "Deep counterfactual regret minimization". In: *International conference on machine learning*. PMLR. 2019, pp. 793–802.

[28] Noam Brown and Tuomas Sandholm. "Superhuman AI for heads-up no-limit poker: Libratus beats top professionals". In: *Science* 359.6374 (2018), pp. 418–424.

[29] Noam Brown and Tuomas Sandholm. "Superhuman AI for multiplayer poker". In: *Science* 365.6456 (2019), pp. 885–890.

[30] Sébastien Bubeck. "Introduction to online optimization". In: *Lecture notes* 2 (2011), pp. 1–86.

[31] Yang Cai, Ozan Candogan, Constantinos Daskalakis, and Christos Papadimitriou. "Zero-sum Polymatrix Games: A Generalization of Minmax". In: *Mathematics of Operations Research* 41.2 (2016), pp. 648–655.

[32] Yang Cai and Constantinos Daskalakis. "On minmax theorems for multiplayer games". In: *Proceedings of the twenty-second annual ACM-SIAM symposium on Discrete algorithms*. SIAM. 2011, pp. 217–234.

[33] Adrian Rivera Cardoso, Jacob Abernethy, He Wang, and Huan Xu. "Competing against Nash equilibria in adversarially changing zero-sum games". In: *International Conference on Machine Learning*. 2019, pp. 921–930.

[34] Andrea Celli, Alberto Marchesi, Gabriele Farina, and Nicola Gatti. "Decentralized No-regret Learning Algorithms for Extensive-form Correlated Equilibria (Extended Abstract)". In: *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*. Ed. by Zhi-Hua Zhou. ijcai.org, 2021, pp. 4755–4759.

[35] Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

[36] Volkan Cevher, Georgios Piliouras, Ryann Sim, and Stratis Skoulakis. "Min-Max Optimization Made Simple: Approximating the Proximal Point Method via Contraction Maps". In: *SIAM Symposium on Simplicity in Algorithms* (2023).

[37] Xi Chen, Xiaotie Deng, and Shang-Hua Teng. "Settling the complexity of computing two-player Nash equilibria". In: *Journal of the ACM (JACM)* 56.3 (2009), pp. 1–57.

[38] Xi Chen and Binghui Peng. "Hedging in games: Faster convergence of external and swap regrets". In: *In Advances in Neural Information Processing Systems*. 2020.

[39] Yun Kuen Cheung. "Multiplicative Weights Updates with Constant Step-size in Graphical Constant-sum Games". In: *Advances in Neural Information Processing Systems*. 2018, pp. 3528–3538.

[40] Yun Kuen Cheung and Georgios Piliouras. "Vortices Instead of Equilibria in MinMax Optimization: Chaos and Butterfly Effects of Online Learning in Zero-Sum Games". In: *Conference on Learning Theory*. 2019, pp. 807–834.

[41] Earl A Coddington and Norman Levinson. *Theory of ordinary differential equations*. Tata McGraw-Hill Education, 1955.

[42] Victor Costa, Nuno Lourenço, João Correia, and Penousal Machado. "COEGAN: evaluating the coevolution effect in generative adversarial networks". In: *Proceedings of the Genetic and Evolutionary Computation Conference*. 2019, pp. 374–382.

[43] Victor Costa, Nuno Lourenço, João Correia, and Penousal Machado. "Using Skill Rating as Fitness on the Evolution of GANs". In: *International Conference on the Applications of Evolutionary Computation (Part of EvoStar)*. Springer. 2020, pp. 562–577.

[44] Wojciech M Czarnecki et al. "Real world games look like spinning tops". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 17443–17454.

[45] George Dantzig. "Linear programming and extensions". In: *Linear programming and extensions*. Princeton university press, 2016.

[46] Constantinos Daskalakis, Alan Deckelbaum, and Anthony Kim. "Near-optimal No-regret Algorithms for Zero-sum Games". In: *Proceedings of the Twenty-second Annual ACM-SIAM Symposium on Discrete Algorithms*. SODA '11. San Francisco, California: Society for Industrial and Applied Mathematics, 2011, pp. 235–254. URL: http://dl.acm.org/citation.cfm?id=2133036.2133057.

[47] Constantinos Daskalakis, Maxwell Fishelson, and Noah Golowich. "Near-optimal no-regret learning in general games". In: *Advances in Neural Information Processing Systems* 34 (2021).

[48] Constantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou. "The Complexity of Computing a Nash Equilibrium". In: *SIAM Journal on Computing* 39.1 (2009), pp. 195–259. DOI: 10.1137/070699652. eprint: https://doi.org/10.1137/070699652. URL: https://doi.org/10.1137/070699652.

[49] Constantinos Daskalakis, Andrew Ilyas, Vasilis Syrgkanis, and Haoyang Zeng. "Training GANs with Optimism." In: *International Conference on Learning Representations (ICLR 2018)*. 2018.

[50] Constantinos Daskalakis and Qinxuan Pan. "A counter-example to Karlin's strong conjecture for fictitious play". In: *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*. IEEE. 2014, pp. 11–20.

[51] Constantinos Daskalakis and Christos H Papadimitriou. "On a network generalization of the minmax theorem". In: *International Colloquium on Automata, Languages, and Programming*. Springer. 2009, pp. 423–434.

[52] Jan De Houwer, Dermot Barnes-Holmes, and Agnes Moors. "What is learning? On the nature and merits of a functional definition of learning". In: *Psychonomic bulletin & review* 20 (2013), pp. 631–642.

[53] Benoit Duvocelle, Panayotis Mertikopoulos, Mathias Staudigl, and Dries Vermeulen. "Learning in time-varying games". In: *arXiv preprint arXiv:1809.03066* (2018).

[54] Benoit Duvocelle, Panayotis Mertikopoulos, Mathias Staudigl, and Dries Vermeulen. "Multiagent online learning in time-varying games". In: *Mathematics of Operations Research* 48.2 (2023), pp. 914–941.

[55] Jens Eisert, Martin Wilkens, and Maciej Lewenstein. "Quantum games and quantum strategies". In: *Physical Review Letters* 83.15 (1999), p. 3077.

[56] Meta Fundamental AI Research Diplomacy Team (FAIR)† et al. "Human-level play in the game of Diplomacy by combining language models with strategic reasoning". In: *Science* 378.6624 (2022), pp. 1067–1074.

[57] Gabriele Farina, Tommaso Bianchi, and Tuomas Sandholm. "Coarse Correlation in Extensive-Form Games". In: *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*. AAAI Press, 2020, pp. 1934–1941.

[58] Gabriele Farina, Christian Kroer, Noam Brown, and Tuomas Sandholm. "Stable-predictive optimistic counterfactual regret minimization". In: *International conference on machine learning*. PMLR. 2019, pp. 1853–1862.

[59] Gabriele Farina, Christian Kroer, and Tuomas Sandholm. "Optimistic regret minimization for extensive-form games via dilated distance-generating functions". In: *arXiv preprint arXiv:1910.10906* (2019).

[60] Gabriele Farina, Alberto Marchesi, Christian Kroer, Nicola Gatti, and Tuomas Sandholm. "Trembling-Hand Perfection in Extensive-Form Games with Commitment". In: *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*. Ed. by Jérôme Lang. ijcai.org, 2018, pp. 233–239.

[61] Farzan Farnia and Asuman Ozdaglar. "Gans may have no nash equilibria". In: *arXiv preprint arXiv:2002.09124* (2020).

[62] Tanner Fiez, Ryann Sim, Stratis Skoulakis, Georgios Piliouras, and Lillian Ratliff. "Online Learning in Periodic Zero-Sum Games". In: *Advances in Neural Information Processing Systems* 34 (2021).

[63] Daniel Friedman. "Evolutionary Games in Economics". In: *Econometrica* 59.3 (1991), pp. 637–666. ISSN: 00129682, 14680262. URL: http://www.jstor.org/stable/2938222.

[64] Drew Fudenberg and David M Kreps. "Learning mixed equilibria". In: *Games and economic behavior* 5.3 (1993), pp. 320–367.

[65] Drew Fudenberg and David K Levine. *The theory of learning in games*. Vol. 2. MIT press, 1998.

[66] Unai Garciarena, Roberto Santana, and Alexander Mendiburu. "Evolved GANs for generating Pareto set approximations". In: *Proceedings of the Genetic and Evolutionary Computation Conference*. 2018, pp. 434–441.

[67] Ian Gemp, Brian McWilliams, Claire Vernade, and Thore Graepel. "Eigengame: PCA as a nash equilibrium". In: *arXiv preprint arXiv:2010.00554* (2020).

[68] Angeliki Giannou, Emmanouil-Vasileios Vlatakis-Gkaragkounis, and Panayotis Mertikopoulos. "On the rate of convergence of regularized learning in games: From bandits and uncertainty to optimism and beyond". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 22655–22666.

[69] Gauthier Gidel, Hugo Berard, Gaëtan Vignoud, Pascal Vincent, and Simon Lacoste-Julien. "A variational inequality perspective on generative adversarial networks". In: *arXiv preprint arXiv:1802.10551* (2018).

[70] Paul W Goldberg, Christos H Papadimitriou, and Rahul Savani. "The complexity of the homotopy method, equilibrium selection, and Lemke-Howson solutions". In: *ACM Transactions on Economics and Computation (TEAC)* 1.2 (2013), pp. 1–25.

[71] Sidney Golden. "Lower Bounds for the Helmholtz Function". In: *Phys. Rev.* 137 (4B 1965), B1127–B1128. DOI: 10.1103/PhysRev.137.B1127. URL: https://link.aps.org/doi/10.1103/PhysRev.137.B1127.

[72] Noah Golowich, Sarath Pattathil, and Constantinos Daskalakis. "Tight last-iterate convergence rates for no-regret learning in multi-player games". In: *Advances in neural information processing systems* 33 (2020), pp. 20766–20778.

[73] Noah Golowich, Sarath Pattathil, Constantinos Daskalakis, and Asuman Ozdaglar. "Last iterate is slower than averaged iterate in smooth convex-concave saddle point problems". In: *Conference on Learning Theory*. PMLR. 2020, pp. 1758–1784.

[74] Ian J. Goodfellow et al. "Generative Adversarial Nets". In: *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*. NIPS'14. Montreal, Canada: MIT Press, 2014, 2672–2680.

[75] Jack Kenneth Hale. *Ordinary differential equations*. Robert E. Krieger Publishing Company, 1980.

[76] James Hannan. "Approximation to Bayes risk in repeated play". In: *Contributions to the Theory of Games* 3.2 (1957), pp. 97–139.

[77] John C Harsanyi, Reinhard Selten, et al. "A general theory of equilibrium selection in games". In: *MIT Press Books* 1 (1988).

[78] Sergiu Hart and Andreu Mas-Colell. *Simple adaptive strategies: from regret-matching to uncoupled dynamics*. Vol. 4. World Scientific, 2013.

[79] Philip Hartman. *Ordinary differential equations*. SIAM, 2002.

[80] Elad Hazan et al. "Introduction to online convex optimization". In: *Foundations and Trends® in Optimization* 2.3-4 (2016), pp. 157–325.

[81] Johannes Heinrich and David Silver. "Deep reinforcement learning from self-play in imperfect-information games". In: *arXiv preprint arXiv:1603.01121* (2016).

[82] Samid Hoda, Andrew Gilpin, Javier Peña, and Tuomas Sandholm. "Smoothing Techniques for Computing Nash Equilibria of Sequential Games". In: *Math. Oper. Res.* 35.2 (2010), pp. 494–512.

[83] Wassily Hoeffding and J Wolfowitz. "Distinguishability of sets of distributions". In: *The Annals of Mathematical Statistics* 29.3 (1958), pp. 700–718.

[84] Josef Hofbauer, Karl Sigmund, et al. *Evolutionary games and population dynamics*. Cambridge university press, 1998.

[85] Josef Hofbauer, Sylvain Sorin, and Yannick Viossat. "Time average replicator and best-reply dynamics". In: *Mathematics of Operations Research* 34.2 (2009), pp. 263–269.

[86] Yu-Guan Hsieh, Kimon Antonakopoulos, and Panayotis Mertikopoulos. "Adaptive Learning in Continuous Games: Optimal Regret Bounds and Convergence to Nash Equilibrium". In: *In Conference on Learning Theory*. 2021.

[87] Ling Huang, Anthony D Joseph, Blaine Nelson, Benjamin IP Rubinstein, and J Doug Tygar. "Adversarial machine learning". In: *Proceedings of the 4th ACM workshop on Security and artificial intelligence*. 2011, pp. 43–58.

[88] Constantin Ickstadt, Thorsten Theobald, and Elias Tsigaridas. "Semidefinite games". In: *arXiv preprint arXiv:2202.12035* (2022).

[89] Georgios Jain Rahul nnd Piliouras and Ryann Sim. "Matrix Multiplicative Weights Updates in Quantum Zero-Sum Games: Conservation Laws & Recurrence". In: *Advances in Neural Information Processing Systems* 35 (2022).

[90] Rahul Jain and John Watrous. "Parallel approximation of non-interactive zero-sum quantum games". In: *2009 24th Annual IEEE Conference on Computational Complexity*. IEEE. 2009, pp. 243–253.

[91] Satyen Kale. *Efficient algorithms using the multiplicative weights update method*. Princeton University, 2007.

[92] Michael Kearns, Michael L Littman, and Satinder Singh. "Graphical Models for Game Theory". In: *arXiv preprint arXiv:1301.2281* (2013).

[93] David Kempe, Jon Kleinberg, and Éva Tardos. "Maximizing the Spread of Influence Through a Social Network". In: *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. 2003, pp. 137–146.

[94] Leonid Genrikhovich Khachiyan. "A polynomial algorithm in linear programming". In: *Doklady Akademii Nauk*. Vol. 244. 5. Russian Academy of Sciences. 1979, pp. 1093–1096.

[95] Naveen Kodali, Jacob Abernethy, James Hays, and Zsolt Kira. "On convergence and stability of gans". In: *arXiv preprint arXiv:1705.07215* (2017).

[96] Christian Kroer, Gabriele Farina, and Tuomas Sandholm. "Solving Large Sequential Games with the Excessive Gap Technique". In: *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*. Ed. by Samy Bengio et al. 2018, pp. 872–882.

[97] Harold William Kuhn and Albert William Tucker. *Contributions to the Theory of Games*. Vol. 2. Princeton University Press, 1953.

[98] H.W. Kuhn. "Extensive Form Games". In: *Proceedings of National Academy of Science* (1950), pp. 570–576.

[99] H.W. Kuhn. "Simplified Two-person Poker". In: *Contributions to the Theory of Games* I (1950), pp. 97–103.

[100] Steven J Lade, Alessandro Tavoni, Simon A Levin, and Maja Schlüter. "Regime shifts in a social-ecological system". In: *Theoretical ecology* 6.3 (2013), pp. 359–372.

[101] Marc Lanctot, Kevin Waugh, Martin Zinkevich, and Michael Bowling. "Monte Carlo Sampling for Regret Minimization in Extensive Games". In: *Advances in neural information processing systems* 22 (2009).

[102] Marc Lanctot et al. "A Unified Game-Theoretic Approach to Multiagent Reinforcement Learning". In: *Advances in Neural Information Processing Systems* 30 (2017), pp. 4190–4203.

[103] Edmund Landau. *Handbuch der Lehre von der Verteilung der Primzahlen*. Vol. 1. BG Teubner, 1909.

[104] Chung-Wei Lee, Christian Kroer, and Haipeng Luo. "Last-iterate convergence in extensive-form games". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 14293–14305.

[105] Carlton E Lemke and Joseph T Howson Jr. "Equilibrium points of bimatrix games". In: *Journal of the Society for industrial and Applied Mathematics* 12.2 (1964), pp. 413–423.

[106] Stefanos Leonardos and Georgios Piliouras. "Exploration-Exploitation in Multi-Agent Learning: Catastrophe Theory Meets Game Theory". In: *Artificial Intelligence* 304 (2022), p. 103653.

[107] Simon A Levin. "Public goods in relation to competition, cooperation, and spite". In: *Proceedings of the National Academy of Sciences* 111.Supplement 3 (2014), pp. 10838–10845.

[108] Wayne Lin, Georgios Piliouras, Ryann Sim, and Antonios Varvitsiotis. "No-Regret Learning and Equilibrium Computation in Quantum Games". In: *arXiv preprint arXiv:2310.08473* (2023).

[109] Wayne Lin, Georgios Piliouras, Ryann Sim, and Antonios Varvitsiotis. "Quantum Potential Games, Replicator Dynamics, and the Separability Problem". In: *arXiv preprint arXiv:2302.04789* (2023).

[110] Nick Littlestone and Manfred K Warmuth. "The weighted majority algorithm". In: *Information and computation* 108.2 (1994), pp. 212–261.

[111] Kyriakos Lotidis, Panayotis Mertikopoulos, and Nicholas Bambos. "Learning in quantum games". In: *arXiv preprint arXiv:2302.02333* (2023).

[112] Kyriakos Lotidis, Panayotis Mertikopoulos, Nicholas Bambos, and Jose Blanchet. "Payoff-based learning with matrix multiplicative weights in quantum games". In: *Advances in Neural Information Processing Systems* 36 (2024).

[113] Nikolai Lusin. "Sur les propriétés des fonctions mesurables". In: *CR Acad. Sci. Paris* 154.25 (1912), pp. 1688–1690.

[114] Tung Mai et al. "Cycles in Zero-Sum Differential Games and Biological Diversity". In: *Proceedings of the 2018 ACM Conference on Economics and Computation*. EC '18. Ithaca, NY, USA: Association for Computing Machinery, 2018, 339–350. ISBN: 9781450358293. DOI: 10.1145/3219166.3219227. URL: https://doi.org/10.1145/3219166.3219227.

[115] Bernard Martinet. "Régularisation d'inéquations variationnelles par approximations successives. Rev. Française Informat". In: *Recherche Opérationnelle* 4 (1970), pp. 154–158.

[116] John Maynard Smith. "Evolution and the Theory of Games". In: *American scientist* 64.1 (1976), pp. 41–45.

[117] Panayotis Mertikopoulos, Christos Papadimitriou, and Georgios Piliouras. "Cycles in adversarial regularized learning". In: *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM. 2018, pp. 2703–2717.

[118] Panayotis Mertikopoulos and Mathias Staudigl. "Equilibrium tracking and convergence in dynamic games". In: *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE. 2021, pp. 930–935.

[119] Panayotis Mertikopoulos et al. "Optimistic Mirror Descent in Saddle-Point Problems: Going the Extra (Gradient) Mile". In: *International Conference on Learning Representations (ICLR)*. 2019.

[120] Risto Miikkulainen et al. "Evolving deep neural networks". In: *Artificial intelligence in the age of neural networks and brain computing*. Elsevier, 2024, pp. 269–287.

[121] Paul Milgrom and Ilya Segal. "Envelope Theorems for Arbitrary Choice Sets". In: *Econometrica* 70.2 (2002), pp. 583–601. DOI: 10.1111/1468-0262.00296.

[122] Jason Milionis, Christos Papadimitriou, Georgios Piliouras, and Kelly Spendlove. "Nash, conley, and computation: Impossibility and incompleteness in game dynamics". In: *arXiv preprint arXiv:2203.14129* (2022).

[123] Aryan Mokhtari, Asuman Ozdaglar, and Sarath Pattathil. "A unified analysis of extra-gradient and optimistic gradient methods for saddle point problems: Proximal point approach". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2020, pp. 1497–1507.

[124] Dov Monderer and Lloyd S Shapley. "Potential games". In: *Games and economic behavior* 14.1 (1996), pp. 124–143.

[125] Jean-Jacques Moreau. "Proximité et dualité dans un espace hilbertien". In: *Bulletin de la Société mathématique de France* 93 (1965), pp. 273–299.

[126] Hervé Moulin and J P Vial. "Strategically zero-sum games: the class of games whose completely mixed equilibria cannot be improved upon". In: *International Journal of Game Theory* 7 (1978), pp. 201–221.

[127] Sai Ganesh Nagarajan, David Balduzzi, and Georgios Piliouras. "From chaos to order: Symmetry and conservation laws in game dynamics". In: *International Conference on Machine Learning*. 2020, pp. 7186–7196.

[128] John Nash. "Non-Cooperative Games". In: *Annals of Mathematics* 54.2 (1951), pp. 286–295. ISSN: 0003486X. URL: http://www.jstor.org/stable/1969529.

[129] Arkadi Nemirovski. "Prox-Method with Rate of Convergence O(1/t) for Variational Inequalities with Lipschitz Continuous Monotone Operators and Smooth Convex-Concave Saddle Point Problems". In: *SIAM J. on Optimization* 15.1 (2005), 229–251.

[130] Arkadij Semenovič Nemirovskij and David Borisovich Yudin. "Problem complexity and method efficiency in optimization". In: (1983).

[131] Yurii Nesterov. "A method for unconstrained convex minimization problem with the rate of convergence O (1/kˆ 2)". In: *Doklady an ussr*. Vol. 269. 1983, pp. 543–547.

[132] John von Neumann. "Zur Theorie der Gesellschaftsspiele". In: *Mathematische Annalen* 100 (1928), pp. 295–300.

[133] N Nisan, T Roughgarden, E Tardos, and VV Vazirani. *Algorithmic Game Theory*. Cambridge university press, 2007.

[134] Gerasimos Palaiopanos, Ioannis Panageas, and Georgios Piliouras. "Multiplicative weights update with constant step-size in congestion games: Convergence, limit cycles and chaos". In: *Advances in Neural Information Processing Systems* 30 (2017).

[135] Julien Perolat et al. "From Poincaré Recurrence to Convergence in Imperfect Information Games: Finding Equilibrium via Regularization". In: *International Conference on Machine Learning*. PMLR. 2021, pp. 8525–8535.

[136] Georgios Piliouras, Carlos Nieto-Granda, Henrik I Christensen, and Jeff S Shamma. "Persistent patterns: Multi-agent learning beyond equilibrium and utility". In: *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*. 2014, pp. 181–188.

[137] Georgios Piliouras and Leonard J Schulman. "Learning dynamics and the coevolution of competing sexual species". In: *arXiv preprint arXiv:1711.06879* (2017).

[138] Georgios Piliouras and Jeff S Shamma. "Optimization despite chaos: Convex relaxations to complex limit sets via Poincaré recurrence". In: *Symposium of Discrete Algorithms*. 2014, pp. 861–873.

[139] Georgios Piliouras, Ryann Sim, and Stratis Skoulakis. "Beyond Time-Average Convergence: Near-Optimal Uncoupled Online Learning via Clairvoyant Multiplicative Weights Update". In: *Advances in Neural Information Processing Systems* 35 (2022).

[140] Henri Poincaré. "Sur le problème des trois corps et les équations de la dynamique". In: *Acta mathematica* 13.1 (1890).

[141] Alexander Rakhlin and Karthik Sridharan. "Online Learning with Predictable Sequences". In: *Conference on Learning Theory*. PMLR. 2013, pp. 993–1019.

[142] Alexander Rakhlin and Karthik Sridharan. "Optimization, Learning, and Games with Predictable Sequences". In: *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013*. 2013, pp. 3066–3074.

[143] Julia Robinson. "An iterative method of solving a game". In: *Annals of mathematics* (1951), pp. 296–301.

[144] R Tyrrell Rockafellar. "Monotone operators and the proximal point algorithm". In: *SIAM journal on control and optimization* 14.5 (1976), pp. 877–898.

[145] Tim Roughgarden. "Algorithmic game theory". In: *Communications of the ACM* 53.7 (2010), pp. 78–86.

[146] Mark Rowland et al. "Multiagent Evaluation under Incomplete Information". In: *Advances in Neural Information Processing Systems* 32 (2019).

[147] Tim Salimans et al. "Improved techniques for training gans". In: *Advances in neural information processing systems* 29 (2016).

[148] William H. Sandholm. *Population Games and Evolutionary Dynamics*. The MIT Press, 2010. ISBN: 9780262195874. URL: http://www.jstor.org/stable/j.ctt5hhbq5.

[149] Yuzuru Sato, Eizo Akiyama, and J Doyne Farmer. "Chaos in learning a simple two-person game". In: *Proceedings of the National Academy of Sciences* 99.7 (2002), pp. 4748–4751.

[150] Thomas C Schelling. *The Strategy of Conflict: with a new Preface by the Author*. Harvard university press, 1980.

[151] Peter Schuster and Karl Sigmund. "Replicator dynamics". In: *Journal of theoretical biology* 100.3 (1983), pp. 533–538.

[152] Reinhard Selten. "Spieltheoretische Behandlung Eines Oligopolmodells Mit Nachfrageträgheit: Teil I: Bestimmung Des Dynamischen Preisgleichgewichts". In: *Zeitschrift für die gesamte Staatswissenschaft/Journal of Institutional and Theoretical Economics* (1965), pp. 301–324.

[153] Pier Giuseppe Sessa, Ilija Bogunovic, Andreas Krause, and Maryam Kamgarpour. "Contextual Games: Multi-Agent Learning with Side Information". In: *Advances in Neural Information Processing Systems*. 2020.

[154] Shai Shalev-Shwartz et al. "Online learning and online convex optimization". In: *Foundations and Trends® in Machine Learning* 4.2 (2012), pp. 107–194.

[155] Lloyd Shapley. "Some topics in two-person games". In: *Advances in game theory* 52 (1964), pp. 1–29.

[156] Yoav Shoham and Kevin Leyton-Brown. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press, 2008.

[157] David Silver et al. "Mastering chess and shogi by self-play with a general reinforcement learning algorithm". In: *arXiv preprint arXiv:1712.01815* (2017).

[158] David Silver et al. "Mastering the game of Go with deep neural networks and tree search". In: *nature* 529.7587 (2016), pp. 484–489.

[159] Ryann Sim, Stratis Skoulakis, Lillian J Ratliff, and Georgios Piliouras. "Fast Convergence of Optimistic Gradient Ascent in Network Zero-Sum Extensive Form Games". In: *International Symposium on Algorithmic Game Theory*. Springer. 2022.

[160] Stratis Skoulakis, Tanner Fiez, Ryann Sim, Georgios Piliouras, and Lillian Ratliff. "Evolutionary Game Theory Squared: Evolving Agents in Endogenously Evolving Zero Sum Games". In: *AAAI Conference on Artificial Intelligence*. 2021.

[161] Stephen Smale. "Dynamics in general equilibrium theory". In: *The American Economic Review* 66.2 (1976), pp. 288–294.

[162] Kenneth O Stanley and Risto Miikkulainen. "Evolving neural networks through augmenting topologies". In: *Evolutionary computation* 10.2 (2002), pp. 99–127.

[163] Alexander J Stewart and Joshua B Plotkin. "Collapse of cooperation in evolving games". In: *Proceedings of the National Academy of Sciences* 111.49 (2014), pp. 17558–17563.

[164] Vasilis Syrgkanis, Alekh Agarwal, Haipeng Luo, and Robert E. Schapire. "Fast Convergence of Regularized Learning in Games". In: *Proceedings of the 28th International Conference on Neural Information Processing Systems*. NIPS'15. Montreal, Canada: MIT Press, 2015, pp. 2989–2997. URL: http://dl.acm.org/citation.cfm?id=2969442.2969573.

[165] Oskari Tammelin, Neil Burch, Michael Johanson, and Michael Bowling. "Solving Heads-Up Limit Texas Hold'em". In: *Twenty-fourth International Joint Conference on Artificial Intelligence*. 2015.

[166] Peter D Taylor and Leo B Jonker. "Evolutionary stable strategies and game dynamics". In: *Mathematical biosciences* 40.1-2 (1978), pp. 145–156.

[167] Colin J. Thompson. "Inequality with Applications in Statistical Mechanics". In: *Journal of Mathematical Physics* 6.11 (1965), pp. 1812–1813. DOI: 10.1063/1.1704727. eprint: https://doi.org/10.1063/1.1704727. URL: https://doi.org/10.1063/1.1704727.

[168] Andrew R Tilman, Joshua B Plotkin, and Erol Akçay. "Evolutionary games with environmental feedbacks". In: *Nature communications* 11.1 (2020), pp. 1–11.

[169] Andrew R Tilman, James R Watson, and Simon Levin. "Maintaining cooperation in social-ecological systems". In: *Theoretical Ecology* 10.2 (2017), pp. 155–165.

[170] Koji Tsuda, Gunnar Rätsch, and Manfred K Warmuth. "Matrix exponentiated gradient updates for on-line learning and Bregman projection". In: *Journal of Machine Learning Research* 6.Jun (2005), pp. 995–1018.

[171] Francisca Vasconcelos, Emmanouil-Vasileios Vlatakis-Gkaragkounis, Panayotis Mertikopoulos, Georgios Piliouras, and Michael I Jordan. "A quadratic speedup in finding Nash equilibria of quantum zero-sum games". In: *arXiv preprint arXiv:2311.10859* (2023).

[172] Emmanouil-Vasileios Vlatakis-Gkaragkounis, Lampros Flokas, Panayotis Mertikopoulos, and Georgios Piliouras. "No-Regret Learning and Mixed Nash Equilibria: They Do Not Mix". In: *Annual Conference on Neural Information Processing Systems*. 2020.

[173] Emmanouil-Vasileios Vlatakis-Gkaragkounis, Lampros Flokas, and Georgios Piliouras. "Poincaré Recurrence, Cycles and Spurious Equilibria in Gradient-Descent-Ascent for Non-Convex Non-Concave Zero-Sum Games". In: *Advances in Neural Information Processing Systems*. 2019, pp. 10450–10461.

[174] John Von Neumann and Oskar Morgenstern. "Theory of games and economic behavior". In: Princeton university press, 2007.

[175] Bernhard Von Stengel. "Efficient computation of behavior strategies". In: *Games and Economic Behavior* 14.2 (1996), pp. 220–246.

[176] Chaoyue Wang, Chang Xu, Xin Yao, and Dacheng Tao. "Evolutionary generative adversarial networks". In: *IEEE Transactions on Evolutionary Computation* 23.6 (2019), pp. 921–934.

[177] Chen-Yu Wei, Chung-Wei Lee, Mengxiao Zhang, and Haipeng Luo. "Last-iterate convergence of decentralized optimistic gradient descent/ascent in infinite-horizon competitive Markov games". In: *Conference on Learning Theory*. PMLR. 2021, pp. 4259–4299.

[178] Chen-Yu Wei, Chung-Wei Lee, Mengxiao Zhang, and Haipeng Luo. "Linear Last-iterate Convergence in Constrained Saddle-point Optimization". In: *International Conference on Learning Representations*. 2020.

[179] Joshua S. Weitz, Ceyhun Eksin, Keith Paarporn, Sam P. Brown, and William C. Ratcliff. "An oscillating tragedy of the commons in replicator dynamics with game-environment feedback". In: *Proceedings of the National Academy of Sciences* 113.47 (2016), E7518–E7525. ISSN: 0027-8424. DOI: 10.1073/pnas.1604096113. eprint: https://www.pnas.org/content/113/47/E7518.full.pdf. URL: https://www.pnas.org/content/113/47/E7518.

[180] Andre Wibisono, Molei Tao, and Georgios Piliouras. "Alternating mirror descent for constrained min-max games". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 35201–35212.

[181] Andre Wibisono, Ashia C Wilson, and Michael I Jordan. "A variational perspective on accelerated methods in optimization". In: *proceedings of the National Academy of Sciences* 113.47 (2016), E7351–E7358.

[182] Yan Wu, Jeff Donahue, David Balduzzi, Karen Simonyan, and Timothy Lillicrap. "LOGAN: Latent Optimisation for Generative Adversarial Networks". In: *arXiv preprint arXiv:1912.00953* (2019).

[183] Siyang Xiong. "Rationalizable implementation of social choice functions: Complete characterization". In: *arXiv preprint arXiv:2202.04885* (2022).

[184] Yaodong Yang and Jun Wang. "An Overview of Multi-agent Reinforcement Learning From Game Theoretical Perspective". In: *arXiv preprint arXiv:2011.00583* (2020).

[185] Mengxiao Zhang, Peng Zhao, Haipeng Luo, and Zhi-Hua Zhou. "No-regret learning in time-varying zero-sum games". In: *International Conference on Machine Learning*. PMLR. 2022, pp. 26772–26808.

[186] Shengyu Zhang. "Quantum strategic game theory". In: *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*. 2012, pp. 39–59.

[187] Martin Zinkevich. "Online convex programming and generalized infinitesimal gradient ascent". In: *Proceedings of the 20th international conference on machine learning (icml-03)*. 2003, pp. 928–936.

[188] Martin Zinkevich, Michael Johanson, Michael H. Bowling, and Carmelo Piccione. "Regret Minimization in Games with Incomplete Information". In: *Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 3-6, 2007*. Ed. by John C. Platt, Daphne Koller, Yoram Singer, and Sam T. Roweis. Curran Associates, Inc., 2007, pp. 1729–1736.

Appendix A

Omitted Proofs from Chapter 3

A.1 Proof of Theorem 3.3.1

Formally, a time-evolving game is defined by a set of populations $(y_1, \dots, y_{n_y})$ and a set of environments $(w_1, \dots, w_{n_w})$, where $y_\ell(0) \in \Delta^{n-1}$ for each $\ell \in \{1, \dots, n_y\}$ and $w_k(0) \in \Delta^{n-1}$ for each $k \in \{1, \dots, n_w\}$. Let $\mathcal{N}_k^w$ be the set of populations which coevolve with the environment w_k and $\mathcal{N}_\ell^y$ be the set of environments which coevolve with the population y_ℓ via the building block structure from Figure 3.2. The time-evolving dynamics for each population ℓ are given componentwise by

$$\dot{y}_{\ell,i} = y_{\ell,i}\left((P_\ell(w)y_\ell)_i - y_\ell^\top P_\ell(w) y_\ell\right), \tag{A.1}$$

where

$$P_\ell(w) = P_\ell + \sum_{k \in \mathcal{N}_\ell^y} W^{\ell,k}$$

and $W^{\ell,k}$ is a matrix such that the (r,s) entry is given by

$$W^{\ell,k} = \begin{pmatrix} 0 & (A^{\ell,k}w_k)_1 - (A^{\ell,k}w_k)_2 & \cdots & (A^{\ell,k}w_k)_1 - (A^{\ell,k}w_k)_n \\ (A^{\ell,k}w_k)_2 - (A^{\ell,k}w_k)_1 & 0 & \cdots & (A^{\ell,k}w_k)_2 - (Aw_k)_n \\ \vdots & \vdots & \vdots & \vdots \\ (A^{\ell,k}w_k)_n - (A^{\ell,k}w_k)_1 & (A^{\ell,k}w_k)_n - (A^{\ell,k}w_k)_2 & \cdots & 0 \end{pmatrix}.$$

Furthermore, the time-evolving dynamics for each environment k are given componentwise by

$$\dot{w}_{k,i} = w_{k,i} \sum_{\ell \in \mathcal{N}_k^w} \sum_{j=1}^{n} w_{k,j}\left((A^{k,\ell}y_\ell)_i - (A^{k,\ell}y_\ell)_j\right). \tag{A.2}$$

We now prove that any time-evolving game defined by the dynamics in (A.1–A.2) is equivalent to replicator dynamics in a polymatrix game.

Environment Dynamics. We begin by showing that the dynamics for each environment reduces to replicator dynamics for a polymatrix game in which each environment plays edge games with each of the populations to which it is connected.

Following a similar argument as in the proof of Proposition 3.3.1, for each $k \in \{1, \dots, n_w\}$, given that $w_k(0) \in \Delta^{n-1}$, we have that $w_k(t) \in \Delta^{n-1}$ for all $t \geq 0$. Since $\sum_{j=1}^{n} w_{k,j}(t) = 1$ for any fixed t and for each environment k, an equivalent form of the dynamics given in (A.2) is

$$\begin{aligned}
\dot{w}_{k,i} &= w_{k,i} \sum_{\ell \in \mathcal{N}_k^w} \sum_{j=1}^{n} w_{k,j}\big((A^{k,\ell} y_\ell)_i - (A^{k,\ell} y_\ell)_j\big) \\
&= w_{k,i} \sum_{\ell \in \mathcal{N}_k^w} \Big(\sum_{j=1}^{n} w_{k,j} (A^{k,\ell} y_\ell)_i - \sum_{j=1}^{n} w_{k,j} (A^{k,\ell} y_\ell)_j \Big) \\
&= w_{k,i} \sum_{\ell \in \mathcal{N}_k^w} \Big((A^{k,\ell} y_\ell)_i - \sum_{j=1}^{n} w_{k,j} (A^{k,\ell} y_\ell)_j \Big) \\
&= w_{k,i} \sum_{\ell \in \mathcal{N}_k^w} \big((A^{k,\ell} y_\ell)_i - w_k^\top A^{k,\ell} y_\ell \big).
\end{aligned}$$

It is now clear that the dynamics from (A.2) equivalently correspond to replicator dynamics where each environment k plays against each connected population $\ell \in \mathcal{N}_k^w$ with payoff matrix $A^{k,\ell}$.

Population Dynamics. We now show that the dynamics for each of the populations reduce to replicator dynamics for a population playing against themselves in a self-loop game and against the environments to which the population is connected.

To begin, from an expansion of the payoff matrix $P_\ell(w)$, the dynamics from (A.1) are equivalent to

$$\begin{aligned}
\dot{y}_{\ell,i} &= y_{\ell,i} \Big((P_\ell y_\ell)_i - y_\ell^\top P_\ell y_\ell \Big) \\
&\quad + y_{\ell,i} \Big(\sum_{k \in \mathcal{N}_\ell^y} \sum_{j=1}^{n} ((A^{\ell,k} w_k)_i - (A^{\ell,k} w_k)_j) y_{\ell,j} - \sum_{k \in \mathcal{N}_\ell^y} \sum_{s=1}^{n} y_{\ell,s} \sum_{j=1}^{n} ((A^{\ell,k} w_k)_s - (A^{\ell,k} w_k)_j) y_{\ell,j} \Big).
\end{aligned}$$

Now, observe that for each $k \in \mathcal{N}_\ell^y$,

$$\begin{aligned}
&\sum_{s=1}^{n} y_{\ell,s} \sum_{j=1}^{n} ((A^{\ell,k} w_k)_s - (A^{\ell,k} w_k)_j) y_{\ell,j} \\
&= \sum_{s=1}^{n} y_{\ell,s} \sum_{j=1}^{n} (A^{\ell,k} w_k)_s y_{\ell,j} - \sum_{s=1}^{n} y_{\ell,s} \sum_{j=1}^{n} (A^{\ell,k} w_k)_j y_{\ell,j} \\
&= \sum_{j=1}^{n} y_{\ell,j} \sum_{s=1}^{n} (A^{\ell,k} w_k)_s y_{\ell,s} - \sum_{s=1}^{n} y_{\ell,s} \sum_{j=1}^{n} (A^{\ell,k} w_k)_j y_{\ell,j} \\
&= 0.
\end{aligned}$$

Hence, along with the fact that $\sum_{j=1}^{n} y_{\ell,j} = 1$, we obtain

$$\begin{aligned}
\dot{y}_{\ell,i} &= y_{\ell,i}\Big((P_\ell y_\ell)_i - y_\ell^\top P_\ell y_\ell\Big) + y_{\ell,i}\sum_{k\in\mathcal{N}_\ell^y}\sum_{j=1}^{n}((A^{\ell,k}w_k)_i - (A^{\ell,k}w_k)_j)y_{\ell,j} \\
&= y_{\ell,i}\Big((P_\ell y_\ell)_i - y_\ell^\top P_\ell y_\ell\Big) + y_{\ell,i}\sum_{k\in\mathcal{N}_\ell^y}\Big(\sum_{j=1}^{n}(A^{\ell,k}w_k)_i y_{\ell,j} - \sum_{j=1}^{n}(A^{\ell,k}w_k)_j y_{\ell,j}\Big) \\
&= y_{\ell,i}\Big((P_\ell y_\ell)_i - y_\ell^\top P_\ell y_\ell\Big) + y_{\ell,i}\sum_{k\in\mathcal{N}_\ell^y}\Big((A^{\ell,k}w_k)_i - \sum_{j=1}^{n}(A^{\ell,k}w_k)_j y_{\ell,j}\Big) \\
&= y_{\ell,i}\Big((P_\ell y_\ell)_i - y_\ell^\top P_\ell y_\ell\Big) + y_{\ell,i}\sum_{k\in\mathcal{N}_\ell^y}\big((A^{\ell,k}w_k)_i - y_\ell^\top A^{\ell,k}w_k\big).
\end{aligned}$$

The final equation shows that the dynamics from (A.1) equivalently correspond to replicator where each population ℓ plays against itself with the payoff matrix $A^{\ell,\ell} = P_\ell$ and against each environment $k \in \mathcal{N}_\ell^y$ to which it is connected with payoff matrix $A^{\ell,k}$.

Static Polymatrix Game. It is now clear that the dynamics from (A.1–A.2) correspond to replicator dynamics for a polymatrix game with V the combined index set of environments and populations such that $|V| = n_y + n_w$. The edge games are defined such that each population player ℓ plays against themselves with $A^{\ell,\ell} = P_\ell$ and against each environment $k \in \mathcal{N}_\ell^w$ to which they are connected with game $A^{\ell,k}$, and such that each environment k plays against each population $\ell \in \mathcal{N}_k^w$ to which it is connected with $A^{k,\ell}$. If each $P_\ell = -P_\ell^\top$ and $\sum_{i\in V}\eta_i u_i(x) = 0$ for all $x \in \mathcal{X}$ and some positive coefficients $\{\eta_i\}_{i\in V}$, then the polymatrix game is rescaled zero-sum.

Finally, we remark that while it may appear complex to verify if the polymatrix game resulting from the reduction of the time-evolving dynamics is rescaled zero-sum, [31] have shown that whether a polymatrix game is constant-sum can be determined in polynomial time and this result can apply to rescaled zero-sum games.

Theorem A.1.1 (Theorem 8 [31]). *Let $G = (V, E)$ be a polymatrix game. For any player $i \in V$, pure strategy $\alpha \in \mathcal{S}_i$, and joint strategy x_{-i} of the rest of the players, denote by $W(\alpha, x_{-i}) = \sum_{j\in V} u_j(\alpha, x_{i-1})$ the sum of all players' payoffs when agent i plays strategy α and the rest of the agents play x_{-i}. The polymatrix game G is a constant-sum game if and only if the optimal objective value of the problem*

$$\max_{x_{-i}} \quad W(\beta, x_{-i}) - W(\alpha, x_{-i})$$

equals zero for all $i \in V$ and $\alpha, \beta \in \mathcal{S}_i$. Moreover, this condition can be checked in polynomial time in the number of players and strategies.

A.2 Proof of Lemma 3.4.1

We need to show

$$\sum_{i\in V}\sum_{\alpha\in\mathcal{S}_i}\frac{d}{dz_{i\alpha}}F_{i\alpha}(z)=0.$$

Recall from (3.17) that for each $\alpha\in\mathcal{S}_i$ and $i\in V$,

$$F_{i\alpha}(z)=\sum_{j\in V}\sum_{\beta\in\mathcal{S}_j}A^{ij}_{\alpha\beta}\frac{e^{z_{j\beta}}}{\sum_{\ell\in\mathcal{S}_j}e^{z_{j\ell}}}-\sum_{j\in V}\sum_{\beta\in\mathcal{S}_j}A^{ij}_{1\beta}\frac{e^{z_{j\beta}}}{\sum_{\ell\in\mathcal{S}_j}e^{z_{j\ell}}}.$$

It follows that for any agent $i\in V$,

$$\sum_{\alpha\in\mathcal{S}_i}\frac{d}{dz_{i\alpha}}F_{i\alpha}(z)=\sum_{\alpha\in\mathcal{S}_i}\sum_{j\in V}\sum_{\beta\in\mathcal{S}_j}A^{ij}_{\alpha\beta}\frac{d}{dz_{i\alpha}}\frac{e^{z_{j\beta}}}{\sum_{\ell\in\mathcal{S}_j}e^{z_{j\ell}}}-\sum_{\alpha\in\mathcal{S}_i}\sum_{j\in V}\sum_{\beta\in\mathcal{S}_j}A^{ij}_{1\beta}\frac{d}{dz_{i\alpha}}\frac{e^{z_{j\beta}}}{\sum_{\ell\in\mathcal{S}_j}e^{z_{j\ell}}}.$$

Moreover, observe that for $i\neq j$,

$$\frac{d}{dz_{i\alpha}}\frac{e^{z_{j\beta}}}{\sum_{\ell\in\mathcal{S}_j}e^{z_{j\ell}}}=0.$$

Consequently, we get that

$$\sum_{\alpha\in\mathcal{S}_i}\frac{d}{dz_{i\alpha}}F_{i\alpha}(z)=\sum_{\alpha\in\mathcal{S}_i}\sum_{\beta\in\mathcal{S}_i}A^{ii}_{\alpha\beta}\frac{d}{dz_{i\alpha}}\frac{e^{z_{i\beta}}}{\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}}}-\sum_{\alpha\in\mathcal{S}_i}\sum_{\beta\in\mathcal{S}_i}A^{ii}_{1\beta}\frac{d}{dz_{i\alpha}}\frac{e^{z_{i\beta}}}{\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}}}.$$

We now separate each sum over $\beta\in\mathcal{S}_i$ into a pair of sums over $\beta\neq\alpha$ and $\beta=\alpha$ for $\alpha\in\mathcal{S}_i$ and any $i\in V$ to get that

$$\begin{aligned}\sum_{\alpha\in\mathcal{S}_i}\frac{d}{dz_{i\alpha}}F_{i\alpha}(z)=&\sum_{\alpha\in\mathcal{S}_i}\sum_{\beta\neq\alpha}A^{ii}_{\alpha\beta}\frac{d}{dz_{i\alpha}}\frac{e^{z_{i\beta}}}{\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}}}-\sum_{\alpha\in\mathcal{S}_i}A^{ii}_{\alpha\alpha}\frac{d}{dz_{i\alpha}}\frac{e^{z_{i\alpha}}}{\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}}}\\&-\sum_{\alpha\in\mathcal{S}_i}\sum_{\beta\neq\alpha}A^{ii}_{1\beta}\frac{d}{dz_{i\alpha}}\frac{e^{z_{i\beta}}}{\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}}}-\sum_{\alpha\in\mathcal{S}_i}A^{ii}_{1\alpha}\frac{d}{dz_{i\alpha}}\frac{e^{z_{i\alpha}}}{\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}}}.\end{aligned}\tag{A.3}$$

Recall that the self-loops are antisymmetric, which means that $A^{ii}_{\alpha\alpha}=0$ for any $\alpha\in\mathcal{S}_i$ and $i\in V$. From this property of the game class,

$$\sum_{\alpha\in\mathcal{S}_i}A^{ii}_{\alpha\alpha}\frac{d}{dz_{i\alpha}}\frac{e^{z_{i\alpha}}}{\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}}}=0.$$

Accordingly, an equivalent form of (A.3) is the expression

$$\begin{aligned}\sum_{\alpha\in\mathcal{S}_i}\frac{d}{dz_{i\alpha}}F_{i\alpha}(z) = &\sum_{\alpha\in\mathcal{S}_i}\sum_{\beta\neq\alpha}A^{ii}_{\alpha\beta}\frac{d}{dz_{i\alpha}}\frac{e^{z_{i\beta}}}{\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}}}\\ &-\sum_{\alpha\in\mathcal{S}_i}\sum_{\beta\neq\alpha}A^{ii}_{1\beta}\frac{d}{dz_{i\alpha}}\frac{e^{z_{i\beta}}}{\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}}}\\ &-\sum_{\alpha\in\mathcal{S}_i}A^{ii}_{1\alpha}\frac{d}{dz_{i\alpha}}\frac{e^{z_{i\alpha}}}{\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}}}.\end{aligned} \tag{A.4}$$

The derivatives in (A.4) are given by

$$\frac{d}{dz_{i\alpha}}\frac{e^{z_{i\beta}}}{\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}}} = \begin{cases}\frac{\sum_{\ell\in\mathcal{S}_i}e^{z_{i\alpha}+z_{i\ell}}-e^{z_{i\alpha}+z_{i\alpha}}}{(\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}})^2}, & \alpha=\beta\\ -e^{z_{i\beta}+z_{i\alpha}}/(\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}})^2, & \alpha\neq\beta.\end{cases}$$

Evaluating the derivatives in (A.4), we get that

$$\begin{aligned}\sum_{\alpha\in\mathcal{S}_i}\frac{d}{dz_{i\alpha}}F_{i\alpha}(z) = &-\sum_{\alpha\in\mathcal{S}_i}\sum_{\beta\neq\alpha}A^{ii}_{\alpha\beta}\frac{e^{z_{i\beta}+z_{i\alpha}}}{(\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}})^2}\\ &+\sum_{\alpha\in\mathcal{S}_i}\sum_{\beta\neq\alpha}A^{ii}_{1\beta}\frac{e^{z_{i\beta}+z_{i\alpha}}}{(\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}})^2}-\sum_{\alpha\in\mathcal{S}_i}A^{ii}_{1\alpha}\frac{\sum_{\beta\in\mathcal{S}_i}e^{z_{i\beta}+z_{i\alpha}}-e^{z_{i\alpha}+z_{i\alpha}}}{(\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}})^2}.\end{aligned} \tag{A.5}$$

Moreover, from a series of algebraic manipulations, we find that

$$\begin{aligned}&\sum_{\alpha\in\mathcal{S}_i}\sum_{\beta\neq\alpha}A^{ii}_{1\beta}\frac{e^{z_{i\beta}+z_{i\alpha}}}{(\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}})^2}-\sum_{\alpha\in\mathcal{S}_i}\mathcal{S}^{ii}_{1\alpha}\frac{\sum_{\beta\in\mathcal{S}_i}e^{z_{i\beta}+z_{i\alpha}}-e^{z_{i\alpha}+z_{i\alpha}}}{(\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}})^2}\\ &=\sum_{\alpha\in\mathcal{S}_i}\sum_{\beta\neq\alpha}A^{ii}_{1\beta}\frac{e^{z_{i\beta}+z_{i\alpha}}}{(\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}})^2}+\sum_{\alpha\in\mathcal{S}_i}A^{ii}_{1\alpha}\frac{e^{z_{i\alpha}+z_{i\alpha}}}{(\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}})^2}-\sum_{\alpha\in\mathcal{S}_i}\sum_{\beta\in\mathcal{S}_i}A^{ii}_{1\alpha}\frac{e^{z_{i\beta}+z_{i\alpha}}}{(\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}})^2}\\ &=\sum_{\alpha\in\mathcal{S}_i}\sum_{\beta\in\mathcal{S}_i}A^{ii}_{1\beta}\frac{e^{z_{i\beta}+z_{i\alpha}}}{(\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}})^2}-\sum_{\alpha\in\mathcal{S}_i}\sum_{\beta\in\mathcal{S}_i}A^{ii}_{1\alpha}\frac{e^{z_{i\beta}+z_{i\alpha}}}{(\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}})^2}\\ &=0.\end{aligned}$$

It follows that (A.5) is equivalent to

$$\sum_{\alpha\in\mathcal{S}_i}\frac{d}{dz_{i\alpha}}F_{i\alpha}(z) = -\sum_{\alpha\in\mathcal{S}_i}\sum_{\beta\neq\alpha}A^{ii}_{\alpha\beta}\frac{e^{z_{i\beta}+z_{i\alpha}}}{(\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}})^2}. \tag{A.6}$$

Finally, we reorganize the sums in (A.6) over pairs (α,β) such that $\beta\neq\alpha$ and invoke the fact that each matrix A^{ii} is antisymmetric (meaning that $(A^{ii})^\top=-A^{ii}$) to

conclude

$$\sum_{\alpha\in\mathcal{S}_i}\frac{d}{dz_{i\alpha}}F_{i\alpha}(z)=-\sum_{\alpha\in\mathcal{S}_i}\sum_{\beta\neq\alpha}A^{ii}_{\alpha\beta}\frac{e^{z_{i\beta}+z_{i\alpha}}}{(\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}})^2}=\sum_{(\alpha,\beta):\ \beta\neq\alpha}(-A^{ii}_{\alpha\beta}-A^{ii}_{\beta\alpha})\frac{e^{z_{i\alpha}}+e^{z_{i\beta}}}{(\sum_{\ell\in\mathcal{S}_i}e^{z_{i\ell}})^2}=0. \tag{A.7}$$

So, by summing Equation A.7 over $i\in V$, we obtain

$$\sum_{i\in V}\sum_{\alpha\in\mathcal{S}_i}\frac{d}{dz_{i\alpha}}F_{i\alpha}(z)=0.$$

This completes the proof of the lemma.

A.3 Proof of Lemma 3.4.2

Consider an N-player rescaled zero-sum polymatrix game with an interior Nash equilibrium such that for positive coefficients $\{\eta\}_{i\in V}$, $\sum_{i\in V}\eta_i u_i(x)=0$ for any $x\in\mathcal{X}$. We need to show the function

$$\Phi(t)=\sum_{i\in V}\sum_{\alpha\in\mathcal{S}_i}\eta_i x^*_{i\alpha}\ln x_{i\alpha} \tag{A.8}$$

is time invariant for any trajectory generated by the replicator dynamics, meaning that $\Phi(t)=\Phi(0)$ for all $t\geq 0$.

In order to prove $\Phi(t)$ as given in (A.8) is time-invariant, we show that the time derivative of the function is equal to zero. To begin, recall the form of the replicator dynamics from (3.3) given by

$$\dot{x}_{i\alpha}=x_{i\alpha}(u_{i\alpha}(x)-u_i(x)),\ \ \forall\alpha\in\mathcal{S}_i. \tag{A.9}$$

We simplify the time derivative of $\Phi(t)$ using the structure of the dynamics given in (A.9) as follows:

$$\begin{aligned}\frac{d\Phi(t)}{dt}&=\sum_{i\in V}\eta_i\sum_{\alpha\in\mathcal{S}_i}x^*_{i\alpha}\frac{d\ln x_{i\alpha}}{dt}\\&=\sum_{i\in V}\eta_i\sum_{\alpha\in\mathcal{S}_i}x^*_{i\alpha}\frac{\dot{x}_{i\alpha}}{x_{i\alpha}}\\&=\sum_{i\in V}\eta_i\sum_{\alpha\in\mathcal{S}_i}x^*_{i\alpha}(u_{i\alpha}(x)-u_i(x))\\&=\sum_{i\in V}\eta_i\sum_{\alpha\in\mathcal{S}_i}x^*_{i\alpha}u_{i\alpha}(x)-\sum_{i\in V}\eta_i\sum_{\alpha\in\mathcal{S}_i}x^*_{i\alpha}u_i(x).\end{aligned} \tag{A.10}$$

Let $E'=\{(i,j):i\neq j,(i,j)\in E\}$ denote the edge set of the polymatrix game excluding self-loops. In the remainder of the proof, denote by e_α a one-hot vector of appropriate dimension containing all zeros, except for a one in the α–th entry. Furthermore, recall $u_{i\alpha}(x)$ denotes the utility of player $i\in V$ for playing the pure strategy $\alpha\in\mathcal{S}_i$, which can be represented by $x_i=e_\alpha$, when the rest of the agents play x_{-i}. Then, from the fact that

$A^{ii}_{\alpha\alpha} = 0$ for all $\alpha \in \mathcal{S}_i$ and $i \in V$, we obtain

$$\begin{aligned}\sum_{i\in V}\eta_i \sum_{\alpha\in\mathcal{S}_i} x^*_{i\alpha}u_{i\alpha}(x) &= \sum_{i\in V}\eta_i \sum_{\alpha\in\mathcal{S}_i} x^*_{i\alpha} \sum_{j:(i,j)\in E}(A^{ij}x_j)_\alpha \\ &= \sum_{i\in V}\eta_i \sum_{\alpha\in\mathcal{S}_i} x^*_{i\alpha}(A^{ii}_\alpha e_\alpha)_\alpha + \sum_{i\in V}\eta_i \sum_{\alpha\in\mathcal{S}_i} x^*_{i\alpha} \sum_{j:(i,j)\in E'}(A^{ij}x_j)_\alpha \\ &= \sum_{i\in V}\eta_i \sum_{\alpha\in\mathcal{S}_i} x^*_{i\alpha}A^{ii}_{\alpha\alpha} + \sum_{i\in V}\eta_i \sum_{\alpha\in\mathcal{S}_i} x^*_{i\alpha} \sum_{j:(i,j)\in E'}(A^{ij}x_j)_\alpha \\ &= \sum_{i\in V}\eta_i (x^*_i)^\top \sum_{j:(i,j)\in E'} A^{ij}x_j. \end{aligned} \tag{A.11}$$

Moreover, since $\sum_{\alpha\in\mathcal{S}_i} x^*_{i\alpha} = 1$ for each $i \in V$ and $\sum_{i\in V}\eta_i u_i(x) = 0$ for any strategy profile $x \in \mathcal{X}$ from the game being rescaled zero-sum, we get that

$$\sum_{i\in V}\eta_i \sum_{\alpha\in\mathcal{S}_i} x^*_{i\alpha}u_i(x) = \sum_{i\in V}\eta_i u_i(x) \sum_{\alpha\in\mathcal{S}_i} x^*_{i\alpha} = \sum_{i\in V}\eta_i u_i(x) = 0. \tag{A.12}$$

Combining (A.10), (A.11), and (A.12), we have

$$\frac{d\Phi(t)}{dt} = \sum_{i\in V}\eta_i (x^*_i)^\top \sum_{j:(i,j)\in E'} A^{ij}x_j. \tag{A.13}$$

For the interior Nash equilibrium x^* under consideration,

$$\begin{aligned}\sum_{i\in V}\eta_i u_i(x^*) &= \sum_{i\in V}\eta_i (x^*_i)^\top \sum_{j:(i,j)\in E} A^{ij}x^*_j \\ &= \sum_{i\in V}\eta_i (x^*_i)^\top \sum_{j:(i,j)\in E'} A^{ij}x^*_j \qquad \text{(A.14)} \\ &= 0. \qquad \text{(A.15)}\end{aligned}$$

Note that (A.14) holds from the fact that $(x^*_i)^\top A^{ii}x^*_i = 0$ for all $x^*_i \in \mathcal{X}_i$ and $i \in V$ since the self-loops are antisymmetric and (A.15) as a result of the polymatrix game being rescaled zero-sum. We continue by subtracting (A.14) from (A.13) since it is equal to zero and obtain

$$\begin{aligned}\frac{d\Phi(t)}{dt} &= \sum_{i\in V}\eta_i (x^*_i)^\top \sum_{j:(i,j)\in E'} A^{ij}x_j - \sum_{i\in V}\eta_i (x^*_i)^\top \sum_{j:(i,j)\in E'} A^{ij}x^*_j \qquad \text{(A.16)} \\ &= \sum_{i\in V}\eta_i \sum_{j:(i,j)\in E'} (x^*_i)^\top A^{ij}(x_j - x^*_j). \qquad \text{(A.17)}\end{aligned}$$

We now prove that (A.17) is equal to zero. To do so, we rely on the results of Cai and Daskalakis [32, Section 4], who show that any zero-sum polymatrix game without self-loops can be transformed to a payoff equivalent, pairwise constant-sum game. Indeed, the results of Cai and Daskalakis [32] apply to rescaled zero-sum polymatrix games

since for any strategy profile $x \in \mathcal{X}$,

$$\sum_{i \in V} \eta_i u_i(x) = \sum_{i \in V} x_i^\top \sum_{j:(i,j) \in E} \eta_i A^{ij} x_j = \sum_{i \in V} x_i^\top \sum_{j:(i,j) \in E'} \eta_i A^{ij} x_j = 0.$$

This means that for each edge $(i,j) \in E'$ there exists a matrix B^{ij} such that the following properties hold (see Lemma 3.1, 3.2, and 3.4, respectively in [32]):

Property 1. $\eta_i A^{ij}_{\alpha\beta} - \eta_i A^{ij}_{\alpha\gamma} = B^{ij}_{\alpha\beta} - B^{ij}_{\alpha\gamma}$ for any $\alpha \in \mathcal{S}_i$ and $\beta, \gamma \in \mathcal{S}_j$.

Property 2. $B^{ij} + (B^{ji})^\top = c_{ij} \cdot \mathbf{1}_{n_i \times n_j}$, where $\mathbf{1}_{n_i \times n_j}$ is an $n_i \times n_j$ matrix of all ones.

Property 3. In every joint pure strategy profile, every player $i \in V$ has the same utility in the game defined by the payoff matrices $\{\eta_i A^{ij}\}_{(i,j) \in E'}$ as in the game defined by the payoff matrices $\{B^{ij}\}_{(i,j) \in E'}$.

Fixing a strategy $\gamma \in \mathcal{S}_j$, we can express the summand of (A.17) using Property 1 as follows:

$$\begin{aligned}
(x_i^*)^\top \eta_i A^{ij}(x_j - x_j^*) &= \sum_{\alpha \in \mathcal{S}_i} \sum_{\beta \in \mathcal{S}_j} x^*_{i\alpha} \eta_i A^{ij}_{\alpha\beta}(x_{j\beta} - x^*_{j\beta}) \\
&= \sum_{\alpha \in \mathcal{S}_i} \sum_{\beta \in \mathcal{S}_j} x^*_{i\alpha} \big(B^{ij}_{\alpha\beta} - B^{ij}_{\alpha\gamma} + \eta_i A^{ij}_{\alpha\gamma}\big)(x_{j\beta} - x^*_{j\beta}) \\
&= (x_i^*)^\top B^{ij}(x_j - x_j^*) + \sum_{\alpha \in \mathcal{S}_i} x_{i\alpha} \left(\eta_i A^{ij}_{\alpha\gamma} - B^{ij}_{\alpha\gamma}\right) \sum_{\beta \in \mathcal{S}_j} (x_{j\beta} - x^*_{j\beta}).
\end{aligned} \tag{A.18}$$

Moreover, observe that since both x_j and x_j^* are on the simplex, $\sum_{\beta \in \mathcal{S}_j}(x_{j\beta} - x^*_{j\beta}) = 0$, and consequently

$$\sum_{\alpha \in \mathcal{S}_i} x_{i\alpha} \left(-B^{ij}_{\alpha\gamma} + \eta_i A^{ij}_{\alpha\gamma}\right) \sum_{\beta \in \mathcal{S}_j} (x_{j\beta} - x^*_{j\beta}) = 0. \tag{A.19}$$

Then, relating (A.18) and (A.19), we obtain

$$(x_i^*)^\top \eta_i A^{ij}(x_j - x_j^*) = (x_i^*)^\top B^{ij}(x_j - x_j^*).$$

As a result, (A.17) is equivalently

$$\frac{d\Phi(t)}{dt} = \sum_{i \in V} \sum_{j:(i,j) \in E'} (x_i^*)^\top \eta_i A^{ij}(x_j - x_j^*).$$

Then, swapping the sum indexing and taking the transpose of the quadratic form $(x_i^*)^\top B^{ij}(x_j - x_j^*)$,

$$\begin{aligned}\frac{d\Phi(t)}{dt} &= \sum_{j\in V}\sum_{i:(j,i)\in E'} (x_i^*)^\top B^{ij}(x_j - x_j^*) \\ &= \sum_{i\in V}\sum_{i:(j,i)\in E'} (x_j - x_j^*)^\top (B^{ij})^\top x_i^*.\end{aligned}$$

We now invoke Property 2 to replace $(B^{ij})^\top$ with $c^{ji}\mathbf{1}_{n_j\times n_i} - B^{ji}$ in the previous equation, which results in

$$\frac{d\Phi(t)}{dt} = \sum_{j\in V}\sum_{i:(j,i)\in E'} (x_j - x_j^*)^\top \left(c^{ji}\mathbf{1}_{n_j\times n_i} - B^{ji}\right) x_i^* \tag{A.20}$$

For any $x_j \in \mathcal{X}_j$, $x_j^* \in \mathcal{X}_j$ and $x_i^* \in \mathcal{X}_i$, we have

$$c^{ji}(x_j - x_j^*)^\top \mathbf{1}_{n_j\times n_i} x_i^* = c^{ji}(x_j - x_j^*)^\top \mathbf{1}_{n_j} = c^{ji} - c^{ji} = 0,$$

since $\mathcal{X}_j = \Delta^{n_j}$ and $\mathcal{X}_i = \Delta^{n_i}$ so that $\sum_{\alpha\in\mathcal{S}_j} x_{j\alpha} = \sum_{\alpha\in\mathcal{S}_j} x_{j\alpha}^* = \sum_{\alpha\in\mathcal{S}_i} x_{i\alpha}^* = 1$. Accordingly, we simplify (A.20) and get

$$\frac{d\Phi(t)}{dt} = -\sum_{j\in V}\sum_{i:(j,i)\in E'} (x_j - x_j^*)^\top B^{ji} x_i^*. \tag{A.21}$$

Following a similar argument as above, we analyze the summand in (A.21) for some $j \in V$. Using Property 1 and fixing any strategy $\gamma_i \in \mathcal{S}_i$ for each $i \in V \setminus \{j\}$, we have that

$$\begin{aligned}\sum_{i:(j,i)\in E'} (x_j - x_j^*)^\top B^{ji} x_i^* &= \sum_{i:(j,i)\in E'}\sum_{\alpha\in\mathcal{S}_j}\sum_{\beta\in\mathcal{S}_i} z_{j\alpha} B^{ji}_{\alpha\beta} x^*_{i\beta} \\ &= \sum_{i:(j,i)\in E'}\sum_{\alpha\in\mathcal{S}_j}\sum_{\beta\in\mathcal{S}_i} z_{j\alpha}\big(\eta_j A^{ji}_{\alpha\beta} - \eta_j A^{ji}_{\alpha\gamma_i} + B^{ji}_{\alpha\gamma_i}\big) x^*_{i\beta} \\ &= \sum_{i:(j,i)\in E'} \eta_j (x_j - x_j^*)^\top A^{ji} x_i^* \\ &\qquad + \sum_{i:(j,i)\in E'}\sum_{\alpha\in\mathcal{S}_j}\sum_{\beta\in\mathcal{S}_i} z_{j\alpha}(-\eta_j A^{ji}_{\alpha\gamma_i} + B^{ji}_{\alpha\gamma_i}) x^*_{i\beta}.\end{aligned}$$

where $z_{j\alpha} := x_{j\alpha} - x^*_{j\alpha}$. We now examine the last term in the equation overhead, and use the fact $\sum_{\beta \in \mathcal{S}_i} x^*_{i\beta} = 1$ since $x^*_i \in \mathcal{X}^*_i = \Delta^{n_i - 1}$ to get that

$$\sum_{i:(j,i)\in E'} \sum_{\alpha \in \mathcal{S}_j} \sum_{\beta \in \mathcal{S}_i} (x_{j\alpha} - x^*_{j\alpha})(-\eta_j A^{ji}_{\alpha\gamma_i} + B^{ji}_{\alpha\gamma_i}) x^*_{i\beta} \tag{A.22}$$

$$= \sum_{i:(j,i)\in E'} \sum_{\alpha \in \mathcal{S}_j} (x_{j\alpha} - x^*_{j\alpha})(-\eta_j A^{ji}_{\alpha\gamma_i} + B^{ji}_{\alpha\gamma_i}) \sum_{\beta \in \mathcal{S}_i} x^*_{i\beta}$$

$$= \sum_{\alpha \in \mathcal{S}_j} (x_{j\alpha} - x^*_{j\alpha}) \sum_{i:(j,i)\in E'} (-\eta_j A^{ji}_{\alpha\gamma_i} + B^{ji}_{\alpha\gamma_i}) \tag{A.23}$$

For each $\alpha \in \mathcal{S}_j$, the terms $\sum_{i:(j,i)\in E'} \eta_j A^{ji}_{\alpha\gamma_i}$ and $\sum_{i:(j,i)\in E'} B^{ji}_{\alpha\gamma_i}$ give the utility of player $j \in V$ in the games defined by $\{\eta_j A^{ji}\}_{(j,i)\in E'}$ and $\{B^{ji}\}_{(j,i)\in E'}$ under a pure strategy profile such that agent j plays α and each other agent $i \in V \setminus \{j\}$ plays some $\gamma_i \in \mathcal{S}_i$. From Property 3, we conclude for each $\alpha \in \mathcal{S}_j$ that

$$\sum_{i:(j,i)\in E'} (-\eta_j A^{ji}_{\alpha\gamma_i} + B^{ji}_{\alpha\gamma_i}) = 0.$$

Relating (A.3) back to (A.23) and then (A.22), for each $j \in V$, we obtain

$$\sum_{i:(j,i)\in E'} (x_j - x^*_j)^\top B^{ji} x^*_i = \sum_{i:(j,i)\in E'} \eta_j (x_j - x^*_j)^\top A^{ji} x^*_i. \tag{A.24}$$

Finally, combining (A.24) and (A.21), we have

$$\frac{d\Phi(t)}{dt} = -\sum_{j\in V} \sum_{i:(j,i)\in E} \eta_j (x_j - x^*_j)^\top A^{ji} x^*_i = 0$$

where the final equality holds since x^* is an interior Nash equilibrium, which means $u_{j\alpha}(x^*) = u_j(x^*)$ for all strategies $\alpha \in \mathcal{S}_j$ and any linear combination thereof. Consequently, we conclude that $\Phi(t) = \Phi(0)$ for all $t \geq 0$.

A.4 Proof of Theorem 3.5.1

Let x^* denote the unique Nash equilibrium of the game. Recall that the trajectory $x(t)$ remains on the interior of the simplex for all $t \geq 0$ as a result of Lemma 3.4.2. Integrating the replicator dynamics from (3.3) given by

$$\dot{x}_{i\alpha}(t) = x_{i\alpha}(t)(u_{i\alpha}(x(t)) - u_i(x(t))$$

for each player $i \in V$ and each strategy $\alpha \in \mathcal{S}_i$, we obtain

$$\frac{1}{T} \int_{x(0)}^{x(T)} \frac{1}{x_{i\alpha}(\tau)} dx(\tau) = \frac{1}{T} \int_0^T (u_{i\alpha}(x(\tau)) - u_i(x(\tau)))\, d\tau.$$

Furthermore,

$$\frac{1}{T}\int_{x(0)}^{x(T)} \frac{1}{x_{i\alpha}(\tau)} dx(\tau) = \frac{1}{T}\left(\log x_{i\alpha}(T) - \log x_{i\alpha}(0)\right)$$

so that

$$\frac{1}{T}\int_0^T \left(u_{i\alpha}(x(\tau)) - u_i(x(\tau))\right) d\tau = \frac{1}{T}\left(\log x_{i\alpha}(T) - \log x_{i\alpha}(0)\right). \tag{A.25}$$

Define

$$z_{i\alpha}(T) = \frac{1}{T}\int_0^T x_{i\alpha}(\tau)\, d\tau.$$

Clearly $z_{i\alpha}(T)$ is bounded for all T since $x_{i\alpha}(T)$ remains bounded. Moreover, the bounds on $z_{i\alpha}(T)$ are the same as those on $x_{i\alpha}(T)$. Consider any sequence T_k converging to infinity. The Bolzano–Weierstrass theorem guarantees that the bounded sequence $z_{i\alpha}(T_k)$ admits a convergent subsequence $z_{i\alpha}(T_{k_\ell})$ such that $z_{i\alpha}(T_{k_\ell})$ converges towards some limit which we denote by $\bar{x}_{i\alpha}$. Since we can repeat this argument for all $i \in V$ and all $\alpha \in \mathcal{S}_i$, let $\bar{x}_i = (\bar{x}_1, \ldots, \bar{x}_{n_i})$ for each $i \in V$.

The sequences $\log(x_{i\alpha}(T_k)) - \log(x_{i\alpha}(0))$ are also bounded. Passing to the limit in (A.25) and using the fact that $x_{i\alpha}(t)$ remains bounded away from zero for all $t \geq 0$, for each $i \in V$, we have that

$$\lim_{T\to\infty} \frac{1}{T}\int_0^T \left(u_{i\alpha}(x(\tau)) - u_i(x(\tau))\right) d\tau = 0, \;\; \forall \alpha \in \mathcal{S}_i. \tag{A.26}$$

Rearranging (A.26), for each $i \in V$, we have that

$$\begin{aligned}
\lim_{T\to\infty} \frac{1}{T}\int_0^T u_i(x(\tau))d\tau &= \lim_{T\to\infty} \frac{1}{T}\int_0^T u_{i\alpha}(x(\tau))d\tau, \;\; \forall \alpha \in \mathcal{S}_i \\
&= \lim_{T\to\infty} \frac{1}{T}\int_0^T \sum_{j:(i,j)\in E} (A^{ij}x_j(\tau))_\alpha d\tau, \;\; \forall \alpha \in \mathcal{S}_i \\
&= \sum_{j:(i,j)\in E} (A^{ij}\bar{x}_j)_\alpha, \;\; \forall \alpha \in \mathcal{S}_i,
\end{aligned} \tag{A.27}$$

where the last equality follows from linearity of the integral, finiteness of the sum, and the well-defined limit. Hence, weighting by $\bar{x}_{i\alpha}$ and summing across $\alpha \in \mathcal{S}_i$, we have that

$$\begin{aligned}
u_i(\bar{x}) &= \sum_{\alpha\in\mathcal{S}_i} \bar{x}_{i\alpha} \sum_{j:(i,j)\in E} (A^{ij}\bar{x}_j)_\alpha \\
&= \sum_{\alpha\in\mathcal{S}_i} \bar{x}_{i\alpha} \left(\lim_{T\to\infty} \frac{1}{T}\int_0^T u_i(x(\tau))d\tau\right) \\
&= \lim_{T\to\infty} \frac{1}{T}\int_0^T u_i(x(\tau))d\tau
\end{aligned}$$

where the last equality holds since $\sum_{\alpha\in\mathcal{S}_i}\bar{x}_{i\alpha}=1$. In turn, the above implies that

$$u_i(\bar{x})=\sum_{j:(i,j)\in E}(A^{ij}\bar{x}_j)_\alpha=u_{i\alpha}(\bar{x}),\ \forall\alpha\in\mathcal{S}_i$$

so that $\bar{x}=(\bar{x}_1,\ldots,\bar{x}_N)$ is a Nash Equilibrium. Since there exists a unique Nash equilibrium by assumption, we have that $\bar{x}=x^*$ which proves (i). Combining this fact with (A.27), we have that

$$\lim_{T\to\infty}\frac{1}{T}\int_0^T u_i(x(\tau))d\tau=u_{i\alpha}(x^*)=u_i(x^*)$$

which proves (ii).

A.5 Proof of Theorem 3.5.2

Let OPT denote the optimal value of the linear program

$$\begin{aligned}\min_{x\in\mathcal{X}}\quad&\sum_{i=1}^n\eta_i v_i\\ \text{s.t}\quad&v_i\geq u_{i\alpha}(x),\ \forall\, i\in V,\ \forall\,\alpha\in\mathcal{S}_i.\end{aligned}\tag{A.28}$$

We begin by proving that OPT ≤ 0. Since a Nash equilibrium always exists [128], there exists a strategy profile x such that $\max_{\alpha\in\mathcal{S}_i}u_{i\alpha}(x)=u_i(x)$. That is,

$$\max_{\alpha\in\mathcal{S}_i}\sum_{j:(i,j)\in E}(A^{ij}x_j)_\alpha=\sum_{\alpha\in\mathcal{S}_i}x_{i\alpha}\sum_{j:(i,j)\in E}(A^{ij}x_j)_\alpha,\ \forall\, i\in V.\tag{A.29}$$

Let $v_i=\max_{\alpha\in\mathcal{S}_i}\sum_{j:(i,j)\in E}(A^{ij}x_j)_\alpha$ for all $i\in V$. Then, the pair of vectors (v,x) forms a feasible solution for the linear program in (A.28). As a result, using (A.29), we have that

$$\text{OPT}\leq\sum_{i\in V}\eta_i v_i=\sum_{i\in V}\eta_i\sum_{\alpha\in\mathcal{S}_i}x_{i\alpha}\sum_{j:(i,j)\in E}(A^{ij}x_j)_\alpha=0$$

where the last equality follows by the fact that $\sum_{i=1}^n\eta_i u_i(x)=0$ since the game is rescaled zero-sum.

Let (v^*,x^*) denote the optimal solution of the linear program in (A.28). We now prove that x^* is a Nash equilibrium using the fact that OPT ≤ 0 . For the sake of contradiction, assume x^* is not a Nash equilibrium, which would mean there exists an agent $i\in V$ and a strategy $\alpha\in\mathcal{S}_i$ satisfying

$$\max_{\alpha\in\mathcal{S}_i}u_{i\alpha}(x^*)=\max_{\alpha\in\mathcal{S}_i}\sum_{j:(i,j)\in E}(A^{ij}x_j^*)_\alpha>\sum_{\alpha\in\mathcal{S}_i}x_{i\alpha}^*\sum_{j:(i,j)\in E}(A^{ij}x_j^*)_\alpha=u_i(x^*).\tag{A.30}$$

Moreover, since (v^*,x^*) is the optimal solution of the linear program in (A.28), we know that $v_i^*\geq u_{i\alpha}(x^*)$ for all $i\in V$ and $\alpha\in\mathcal{S}_i$, which then implies $v_i^*\geq\max_{\alpha\in\mathcal{S}_i}u_{i\alpha}(x^*)$ for

all $i \in V$. As a direct result, we obtain the inequality

$$\mathrm{OPT} = \sum_{i \in V} \eta_i v_i^* \geq \sum_{i \in V} \eta_i \max_{\alpha \in \mathcal{S}_i} \sum_{j:(i,j)\in E} (A^{ij} x_j^*)_\alpha \tag{A.31}$$

Finally, combining (A.30) and (A.31), we get that

$$\mathrm{OPT} > \sum_{i \in V} \eta_i \sum_{\alpha \in \mathcal{S}_i} x_{i\alpha}^* \sum_{j:(i,j)\in E} (A^{ij} x_j^*)_\alpha = 0,$$

where the last equality follows by the fact that $\sum_{i=1}^n \eta_i u_i(x) = 0$ since the game is rescaled zero-sum. Yet, this leads to contradiction since $\mathrm{OPT} \leq 0$, which means x^* must be a Nash equilibrium.

A.6 Proof of Proposition 3.5.1

We begin by presenting preliminaries and notation needed for an intermediate technical result. Denote by $v_i(x) = (u_{i\alpha}(x))_{\alpha \in \mathcal{S}_i}$ the payoff vector for any agent $i \in V$ that includes the utility of each pure strategy $\alpha \in \mathcal{S}_i$ under the joint profile $x = (\alpha, x_{-i}) \in \mathcal{X}$. The utility of the player $i \in V$ under the joint strategy profile $x = (x_i, x_{-i}) \in \mathcal{X}$ is then given by $u_i(x) = \langle v_i(x), x_i \rangle$. The learning dynamics given by

$$\begin{aligned} y_i(t) &= \int_0^t v_i(x(\tau)) d\tau \\ x_i(t) &= Q_i(y_i(t)) \end{aligned}$$

characterize the `FTRL` updates for player $i \in V$ at time $t \geq 0$. The so-called choice map $Q_i : \mathbb{R}^{n_i} \to \mathcal{X}_i$ is defined by

$$Q_i(y_i) = \arg\max_{x_i \in \mathcal{X}_i} \{\langle y_i, x_i \rangle - h_i(x_i)\}$$

for a strongly convex and continuously differentiable regularizer function $h_i : \mathcal{X}_i \to \mathbb{R}$. The strong convexity of h_i along with the convexity and compactness of $\mathcal{X}_i$ ensure a unique solution exists for the update $x_i(t)$ so that it is well-defined. The negative entropy regularizer function

$$h_i(x_i) = \sum_{\alpha \in \mathcal{S}_i} x_{i\alpha} \log(x_{i\alpha})$$

gives rise to the replicator dynamics we study in this work. Furthermore, the convex conjugate of the regularizer function h_i is given by

$$h_i^*(y_i) = \max_{x_i \in \mathcal{X}_i} \{\langle y_i, x_i \rangle - h_i(x_i)\}.$$

A simple corollary of this definition is Fenchel's inquality, which says for every $x_i \in \mathcal{X}_i$ and $y_i \in \mathbb{R}^{n_i}$,

$$\langle y_i, x_i \rangle \leq h_i(x_i) + h_i^*(y_i).$$

Moreover, by the maximizing argument (see e.g., [154, Ch. 2]), $x_i(t) = Q_i(y_i(t)) = \nabla h_i^*(y_i(t))$.

We now state and prove an intermediate result, which we then invoke to conclude Proposition 3.5.1.

Lemma A.6.1. *Let $h_{\max,i} = \max_{x_i \in \mathcal{X}_i} h_i(x_i)$ and $h_{\min,i} = \min_{x_i \in \mathcal{X}_i} h_i(x_i)$. If player $i \in V$ follows the replicator dynamics from* (3.3)*, then independent of the rest of the players in the game,*

$$\max_{x_i \in \mathcal{X}_i} \int_0^t \langle v_i(x(\tau)), x_i \rangle ds - \int_0^t \langle v_i(x(\tau)), x_i(\tau) \rangle d\tau \leq h_{\max,i} - h_{\min,i}.$$

Proof of Lemma A.6.1. We begin by deriving a bound for every fixed $x_i \in \mathcal{X}_i$ on the expression

$$\int_0^t \langle v_i(x(\tau)), x_i \rangle d\tau. \tag{A.32}$$

From the definition of the utility dynamics given by

$$y_i(t) = \int_0^t v_i(x(\tau)) d\tau,$$

an equivalent representation of (A.32) is

$$\int_0^t \langle v_i(x(\tau)), x_i \rangle d\tau = \langle y_i(t), x_i \rangle. \tag{A.33}$$

From Fenchel's inequality $\langle y_i(t), x_i \rangle \leq h_i(x_i) + h_i^*(y_i(t))$ and by definition $h_i(x_i) \leq h_{\max,i}$. Combining each inequality with (A.33), we get

$$\int_0^t \langle v_i(x(\tau)), x_i \rangle d\tau \leq h_i^*(y_i(t)) + h_{\max,i}. \tag{A.34}$$

We now work on obtaining a bound for $h_i^*(y_i(t))$. Observe that by definition

$$\begin{aligned} h_i^*(y_i(t)) &= \langle y_i(t), Q_i(y_i(t)) \rangle - h_i(Q_i(y_i(t))) \\ &= \int_0^t \langle v_i(x(\tau)), Q_i(y_i(\tau)) \rangle d\tau - h_i(Q_i(y_i(t))). \end{aligned}$$

Now define the function

$$\phi : (z, t) \mapsto \int_0^t \langle v_i(z(\tau)), z(\tau) \rangle d\tau - h(z(t)).$$

For any fixed $t \geq 0$, we can verify by the maximizing argument (see, e.g., [154]) that $Q_i(y_i(t))$ maximizes $\phi(\cdot, t)$, so we can apply the envelope theorem [121] to take the partial derivative of $\phi(Q_i(y_i(t)), t)$ with respect to the argument t. In doing so, we get

$$\frac{d}{dt} h_i^*(y_i(t)) = \frac{\partial}{\partial t} \phi(Q_i(y_i(t)), t)) = \langle v_i(x_i(t)), Q_i(y_i(t)) \rangle.$$

Then, integrating the equation overhead, we obtain

$$h_i^*(y_i(t)) - h_i^*(y_i(0)) = \int_0^t \langle v_i(x(\tau)), Q_i(y_i(\tau))\rangle d\tau.$$

Since $h_i^*(y_i(0)) = -h_{\min,i}$, we get the bound

$$h_i^*(y_i(t)) \leq \int_0^t \langle v_i(x(\tau)), Q_i(y_i(\tau))\rangle d\tau - h_{\min,i}.$$

Finally, combining the previous equation with (A.34), we conclude the stated result of

$$\max_{x_i \in \mathcal{X}_i} \int_0^t \langle v_i(x(\tau)), x_i\rangle d\tau - \int_0^t \langle v_i(x(\tau)), x_i(\tau)\rangle d\tau \leq h_{\max,i} - h_{\min,i}.$$

□

We now return to proving Proposition 3.5.1. By definition $u_i(x) = \langle v_i(x), x_i\rangle$, which means we can directly apply Lemma A.6.1 to the regret definition. We now do so and obtain the stated result:

$$\begin{aligned}
\mathrm{Reg}_i(t) &= \max_{x_i \in \mathcal{X}_i} \frac{1}{t} \int_0^t (u_i(x_i, x_{-i}(\tau)) - u_i(x(\tau)))d\tau \\
&= \max_{x_i \in \mathcal{X}_i} \frac{1}{t} \int_0^t \langle v_i(x(\tau)), x_i - x_i(\tau)\rangle d\tau \\
&\leq \frac{h_{\max,i} - h_{\min,i}}{t} = \frac{\Omega_i}{t}.
\end{aligned}$$

Appendix B

Omitted Proofs from Chapter 4

B.1 Proof of Proposition 4.3.1

To prove this result, we construct time-evolving zero-sum games without both periodic payoffs and a time-invariant equilibrium in which the GDA dynamics are not Poincaré recurrent.

Example 1. Consider a time-evolving zero-sum game on scalar action spaces so that $x_1, x_2 \in \mathbb{R}$ with the time-evolving payoff matrix $A(t) = t^{-2}$. Without loss of generality, we can consider $t > 0$ so that the GDA dynamics are well-defined. This time-evolving zero-sum does not have periodic payoffs, but $(x_1^*, x_2^*) = (0, 0)$ is a time-invariant Nash equilibrium. We now show that the GDA dynamics are not Poincaré recurrent in this game. Before formally proving this, we remark that the intuition for why this statement holds is that since the payoff matrix goes to zero, the distance the dynamics can travel is bounded so it is impossible that the trajectory could return arbitrarily close to an initial condition infinitely often.

The GDA dynamics in this time-evolving zero-sum game are described by the system

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & \frac{1}{t^2} \\ -\frac{1}{t^2} & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}.$$

The solution of a time-varying linear system of this form is given by

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \exp\left(\int_{t_0}^{t} \begin{bmatrix} 0 & \frac{1}{\tau^2} \\ -\frac{1}{\tau^2} & 0 \end{bmatrix} d\tau \right) \begin{bmatrix} x_1(t_0) \\ x_2(t_0) \end{bmatrix}.$$

We consider $t_0 > 1$ without loss of generality. To derive the explicit solution, we begin by computing the integral of the evolving payoff matrix and get that

$$\int_{t_0}^{t} \begin{bmatrix} 0 & \frac{1}{\tau^2} \\ -\frac{1}{\tau^2} & 0 \end{bmatrix} d\tau = \begin{bmatrix} 0 & \frac{1}{t_0} - \frac{1}{t} \\ \frac{1}{t} - \frac{1}{t_0} & 0 \end{bmatrix}.$$

Recalling the following identity

$$\exp\left(\theta \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \right) = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix},$$

we can determine that the matrix exponential is then given by

$$\exp\left(\begin{bmatrix} 0 & \frac{1}{t_0}-\frac{1}{t} \\ \frac{1}{t}-\frac{1}{t_0} & 0 \end{bmatrix}\right) = \begin{bmatrix} \cos\left(\frac{1}{t}-\frac{1}{t_0}\right) & -\sin\left(\frac{1}{t}-\frac{1}{t_0}\right) \\ \sin\left(\frac{1}{t}-\frac{1}{t_0}\right) & \cos\left(\frac{1}{t}-\frac{1}{t_0}\right) \end{bmatrix}.$$

Therefore, the solution of the system simplifies to be given by

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} \cos\left(\frac{1}{t}-\frac{1}{t_0}\right) & -\sin\left(\frac{1}{t}-\frac{1}{t_0}\right) \\ \sin\left(\frac{1}{t}-\frac{1}{t_0}\right) & \cos\left(\frac{1}{t}-\frac{1}{t_0}\right) \end{bmatrix} \begin{bmatrix} x_1(t_0) \\ x_2(t_0) \end{bmatrix},$$

and equivalently,

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} \cos\left(\frac{1}{t}-\frac{1}{t_0}\right)x_1(t_0) - \sin\left(\frac{1}{t}-\frac{1}{t_0}\right)x_2(t_0) \\ \sin\left(\frac{1}{t}-\frac{1}{t_0}\right)x_1(t_0) + \cos\left(\frac{1}{t}-\frac{1}{t_0}\right)x_2(t_0) \end{bmatrix}.$$

Given the explicit form of the solution, we now show that we can construct an initial condition that the strategies do not return back to infinitely often. This will immediately allow us to conclude the system is not Poincaré recurrent by definition. We remark that the following choice of initial condition is only for the simplicity of the proof and identical conclusions would hold for almost all initial conditions.

Let $x_1(t_0) = 1$ and $x_2(t_0) = 0$. Given this initial condition, the solution of the system simplifies to be given by

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} \cos\left(\frac{1}{t}-\frac{1}{t_0}\right) \\ \sin\left(\frac{1}{t}-\frac{1}{t_0}\right) \end{bmatrix}.$$

Taking the limit of the solution as $t \to \infty$, we have that

$$\lim_{t\to\infty} x_1(t) = \lim_{t\to\infty} \cos\left(\frac{1}{t}-\frac{1}{t_0}\right) = \cos\left(\frac{1}{t_0}\right)$$

and

$$\lim_{t\to\infty} x_2(t) = \lim_{t\to\infty} \sin\left(\frac{1}{t}-\frac{1}{t_0}\right) = -\sin\left(\frac{1}{t_0}\right).$$

This shows that the dynamics converge to a fixed point $(\bar{x}_1, \bar{x}_2) = (\cos\left(\frac{1}{t_0}\right), -\sin\left(\frac{1}{t_0}\right))$. Since $(\bar{x}_1, \bar{x}_2) = (\cos\left(\frac{1}{t_0}\right), \sin\left(\frac{1}{t_0}\right)) \neq (1, 0) = (x_1(t_0), x_2(t_0))$ for $t_0 > 1$ unless $t_0 \to \infty$, this immediately implies that the GDA dynamics are not Poincaré recurrent in this time-evolving zero-sum game since they do not return infinitely often back to an arbitrarily small neighborhood around the initial condition.

Example 2. In the previous example, we showed that the GDA dynamics were not Poincaré recurrent in a time-evolving zero-sum game that had a time-invariant Nash equilibrium but not periodic payoffs. In this example, we provide theoretical evidence that the GDA dynamics are not Poincaré recurrent in a time-evolving zero-sum game with periodic payoffs but without a time-invariant Nash equilibrium.

Consider a time-evolving zero-sum game on scalar action spaces so that $x_1, x_2 \in \mathbb{R}$ with a periodic payoff matrix $A(t) = A(t+T)$ for any $t \geq 0$ and $T = 3$ that evolves over a period such that $A(t) = 1$ for $0 \leq t \leq 1$ and $A(t) = -1$ for $1 \leq t \leq 3$. We treat player 2 as a dummy player that just plays the fixed strategy of $x_2 = 1$ for all t. Thus, this time-evolving zero-sum game can be viewed as a trivial game that is equivalent to an optimization problem for player 1. Since player 1 is a utility maximizer, the Nash equilibrium of the game at each time simply corresponds to the strategy of player 1 that maximizes its utility. Thus, the Nash equilibrium when $A(t) = 1$ is $x_1^* = \infty$ and given $A(t) = -1$ it is $x_2 1^* = -\infty$. Therefore, this corresponds to a time-evolving zero-sum game that is periodic but there is not a time-invariant Nash equilibrium.

We now show that the GDA dynamics are not Poincaré recurrent in this game. The dynamics can be described by the system $\dot{x}_1 = A(t)$. Consequently, the solution is $x_1(t) = x_1(t_0) + t$ on the interval with $A(t) = 1$ and $x_1(t) = x_1(t_0) - 1$ on the interval with $A(t) = -1$. This means that after 1 period of the game, $x_1(t) = x_1(t_0) - 1$, which implies that $x_1(t) \to -\infty$ as $t \to \infty$. Thus the dynamics are not Poincaré recurrent in this time-evolving zero-sum game since they do not return infinitely often back to an arbitrarily small neighborhood around the initial condition.

Example 3. Additionally, we provide an experimental example to show the non-existence of Poincaré recurrence in the setting of FTRL dynamics. In particular, we simulate a time-evolving zero-sum game which has periodic payoffs but does not have a time-invariant Nash equilibrium. Consider a time-evolving zero-sum game where the payoff matrix for the first quarter of a period is standard Matching Pennies $A = \left(\begin{smallmatrix} 1 & -1 \\ -1 & 1 \end{smallmatrix}\right)$. For the remaining three quarters of the period it is instead $A = \left(\begin{smallmatrix} 0.05 & -0.5 \\ -0.5 & 5 \end{smallmatrix}\right)$. Note that here the payoff matrix for the second player is just $-A$. The former game has a mixed Nash equilibrium where both players play $[0.5, 0.5]$ and the latter game has a mixed Nash equilibrium where both players play $[0.9091, 0.0909]$. We simulate this example with replicator dynamics, which is an instantiation of FTRL dynamics. First, we plot the trajectories of the player's strategies against each other. Moreover, we plot the L_1-norm between the joint trajectory of the players and the initial condition. The simulation results show that the trajectory does not return back arbitrarily close to the initial condition, thus the dynamics are not Poincaré recurrent in this periodic evolving game without a time-invariant equilibrium (Figure B.1).

B.2 FTRL Poincaré Recurrence: Proof of Lemma 4.4.2

Recall that Lemma 4.4.2 states that the orbits of the $\dot{z}$ dynamics are bounded. To prove this statement, we show that the function

$$\Phi(x^*, y(t)) = \sum_{i \in V} \left(h_i^*(y_i(t)) - \langle x_i^*, y_i(t) \rangle + h_i(x_i^*) \right)$$

is time-invariant where x^* denotes the time-invariant fully mixed Nash equilibrium and then argue that this is sufficient to ensure that orbits of the $\dot{z}$ dynamics are bounded.

To prove that the function $\Phi(x^*, y(t))$ is time-invariant, we show that the time-derivative of the function is equal to zero. The time-derivative of $\Phi(x^*, y(t))$ simplifies using the

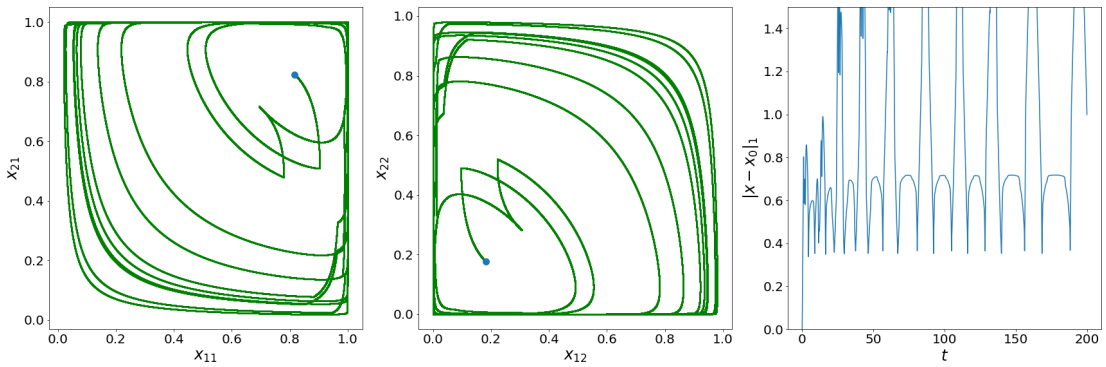


FIGURE B.1: (a, b) Replicator trajectories for periodically evolving game without time-invariant equilibrium. (c) L_1-norm plot showing that recurrence does not hold in this example.

fact that $h_i(x_i^*)$ is time-independent to the following:

$$\frac{d\Phi(x^*, y(t))}{dt} = \frac{d}{dt}\sum_{i\in V} h_i^*(y_i(t)) + \frac{d}{dt}\sum_{i\in V}\langle x_i^*, y_i(t)\rangle.$$

We begin by showing that the time-derivative of $\sum_{i\in V} h_i^*(y_i(t)) = 0$. This holds by the following computation that is explained below:

$$\begin{aligned}
\frac{d}{dt}\sum_{i\in V} h_i^*(y_i(t)) &= \sum_{i\in V}\langle \nabla h_i^*(y_i(t)), \dot{y}_i(t)\rangle && \text{(B.1)}\\
&= \sum_{i\in V}\langle x_i(t), \dot{y}_i(t)\rangle && \text{(B.2)}\\
&= \sum_{i\in V}\langle x_i(t), v_i(x(t), t)\rangle && \text{(B.3)}\\
&= \sum_{i\in V} u_i(x(t), t) && \text{(B.4)}\\
&= 0. && \text{(B.5)}
\end{aligned}$$

We obtain (B.1) by the chain rule, (B.2) by the maximizing argument of convex conjugates (see e.g., [154, Chapter 2]) that implies $x_i(t) = Q_i(y_i(t)) = \nabla h_i^*(y_i(t))$, (B.3) by the definition of $y_i(t)$ and the fundamental theorem of calculus, (B.4) by definition of the pure strategy utilities $v_i(x(t), t)$ and the utility $u_i(x(t), t)$, and (B.5) by the fact that the polymatrix game is zero-sum.

We now finish by showing that the time-derivative of $\sum_{i\in V}\langle x_i^*, y_i(t)\rangle = 0$. To begin, observe that the time-derivative can be described by

$$\begin{aligned}\frac{d}{dt}\sum_{i\in V}\langle x_i^*, y_i(t)\rangle &= \sum_{i\in V}\langle x_i^*, \dot{y}_i(t)\rangle \\ &= \sum_{i\in V}\sum_{j:(i,j)\in E}(x_i^*)^\top A^{ij}(t)x_j(t) \qquad \text{(B.6)}\\ &= \sum_{i\in V}\sum_{j:(i,j)\in E}(x_i^*)^\top A^{ij}(t)(x_j(t)-x_j^*). \qquad \text{(B.7)}\end{aligned}$$

Observe that (B.6) follows from the definition of $y_i(t)$ and the fundamental theorem of calculus and (B.7) comes about from subtracting $\sum_{i\in V} u_i(x^*, t)$ which is zero by the fact that the polymatrix game is zero-sum for any $t \geq 0$.

To continue, we remark that any zero-sum polymatrix game can be transformed to a payoff equivalent, pairwise constant-sum game [32]. This means that for each edge $(i,j) \in E$ there exists a matrix $B^{ij}(t)$ such that the following properties hold (see Lemma 3.1, 3.2, and 3.4, respectively [32]):

Property 1. $A^{ij}_{\alpha\beta}(t) - A^{ij}_{\alpha\gamma}(t) = B^{ij}_{\alpha\beta}(t) - B^{ij}_{\alpha\gamma}(t)$ for any pure strategies $\alpha \in \mathcal{S}_i$ and $\beta, \gamma \in \mathcal{S}_j$.

Property 2. $B^{ij}(t) + (B^{ji}(t))^\top = c_{ij}(t)\cdot \mathbf{1}_{n_i\times n_j}$, where $c_{ij}(t)$ is a constant and $\mathbf{1}_{n_i\times n_j}$ is an $n_i \times n_j$ matrix of ones.

Property 3. In every joint pure strategy profile, every player $i \in V$ has the same utility in the game defined by the individual payoff matrices $\{A^{ij}(t)\}_{(i,j)\in E}$ as in the game defined by the individual payoff matrices $\{B^{ij}(t)\}_{(i,j)\in E}$.

Fixing a strategy $\gamma \in \mathcal{S}_j$, we can equivalently express any summand of (B.7) in the following manner that is justified below:

$$\begin{aligned}(x_i^*)^\top A^{ij}(x_j(t)-x_j^*) &= \sum_{\alpha\in\mathcal{S}_i}\sum_{\beta\in\mathcal{S}_j} x^*_{i\alpha}A^{ij}_{\alpha\beta}(x_{j\beta}(t)-x^*_{j\beta}) \\ &= \sum_{\alpha\in\mathcal{S}_i}\sum_{\beta\in\mathcal{S}_j} x^*_{i\alpha}\big(B^{ij}_{\alpha\beta}(t) - B^{ij}_{\alpha\gamma}(t) + A^{ij}_{\alpha\gamma}(t)\big)(x_{j_\beta}(t)-x^*_{j_\beta}) \qquad \text{(B.8)}\\ &= (x_i^*)^\top B^{ij}(t)(x_j(t)-x_j^*) + \sum_{\alpha\in\mathcal{S}_i} x^*_{i\alpha}\left(A^{ij}_{\alpha\gamma}(t) - B^{ij}_{\alpha\gamma}(t)\right)\sum_{\beta\in\mathcal{S}_j}(x_{j\beta}(t)-x^*_{j\beta}). \\ &= (x_i^*)^\top B^{ij}(t)(x_j(t)-x_j^*). \qquad \text{(B.9)}\end{aligned}$$

Observe that (B.8) results from applying Property 1 and (B.9) holds since both $x_j(t)$ and x_j^* are on the simplex so that $\sum_{\beta\in\mathcal{S}_j} x_{j\beta} = 1$ and $\sum_{\beta\in\mathcal{S}_j} x^*_{j\beta} = 1$ which implies $\sum_{\beta\in\mathcal{S}_j}(x_{j\beta} - x^*_{j\beta}) = 0$.

Thus, continuing from (B.7) and using that $(x_i^*)^\top A^{ij}(x_j(t)-x_j^*) = (x_i^*)^\top B^{ij}(t)(x_j(t)-x_j^*)$ from above and swapping the sum indexing and taking the transpose of the quadratic

form $(x_i^*)^\top B^{ij}(t)(x_j(t) - x_j^*)$, we get that

$$\begin{aligned}\frac{d}{dt}\sum_{i\in V}\langle x_i^*, y_i(t)\rangle &= \sum_{i\in V}\sum_{j:(i,j)\in E}(x_i^*)^\top A^{ij}(t)(x_j(t)-x_j^*)\\ &= \sum_{i\in V}\sum_{j:(i,j)\in E}(x_i^*)^\top B^{ij}(t)(x_j(t)-x_j^*) \qquad \text{(B.10)}\\ &= \sum_{j\in V}\sum_{i:(j,i)\in E}(x_j(t)-x_j^*)^\top (B^{ij}(t))^\top x_i^*.\end{aligned}$$

Moreover, we obtain the following expression that is justified below:

$$\begin{aligned}\frac{d}{dt}\sum_{i\in V}\langle x_i^*, y_i(t)\rangle &= \sum_{j\in V}\sum_{i:(j,i)\in E}(x_j(t)-x_j^*)^\top (B^{ij}(t))^\top x_i^*\\ &= \sum_{j\in V}\sum_{i:(j,i)\in E}(x_j(t)-x_j^*)^\top (c^{ji}(t)\mathbf{1}_{n_j\times n_i} - B^{ji}(t))x_i^* \qquad \text{(B.11)}\\ &= \sum_{j\in V}\sum_{i:(j,i)\in E} c^{ji}(t)(x_j(t)-x_j^*)^\top \mathbf{1}_{n_j\times n_i}x_i^* - \sum_{j\in V}\sum_{i:(j,i)\in E}(x_j(t)-x_j^*)^\top B^{ji}(t)x_i^*\\ &= -\sum_{j\in V}\sum_{i:(j,i)\in E}(x_j(t)-x_j^*)^\top B^{ji}(t)x_i^*. \qquad \text{(B.12)}\end{aligned}$$

Note that (B.11) results from applying Property 2 and we obtain (B.12) using that $\sum_{j\in V}\sum_{i:(j,i)\in E} c^{ji}(t)(x_j(t)-x_j^*)^\top = 0$ since each summand is zero as can be seen by noting that $\sum_{\alpha\in\mathcal{S}_j} x_{j\alpha} = \sum_{\alpha\in\mathcal{S}_j} x_{j\alpha}^* = \sum_{\alpha\in\mathcal{S}_i} x_{i\alpha}^* = 1$ which gives

$$c^{ji}(t)(x_j(t)-x_j^*)^\top \mathbf{1}_{n_j\times n_i}x_i^* = c^{ji}(x_j(t)-x_j^*)^\top \mathbf{1}_{n_j} = c^{ji}(t) - c^{ji}(t) = 0.$$

We now analyze the summand in (B.12) for some $j \in V$. Fixing any pure strategy $\gamma_i \in \mathcal{S}_i$ for each $i \in V\setminus\{j\}$, obtain the following simplification that is explained below:

$$\begin{aligned}&\sum_{i:(j,i)\in E}(x_j(t)-x_j^*)^\top B^{ji}(t)x_i^* = \sum_{i:(j,i)\in E}\sum_{\alpha\in\mathcal{S}_j}\sum_{\beta\in\mathcal{S}_i}(x_{j\alpha}(t)-x_{j\alpha}^*)B_{\alpha\beta}^{ji}(t)x_{i\beta}^*\\ &= \sum_{i:(j,i)\in E}\sum_{\alpha\in\mathcal{S}_j}\sum_{\beta\in\mathcal{S}_i}(x_{j\alpha}(t)-x_{j\alpha}^*)\big(A_{\alpha\beta}^{ji}(t) - A_{\alpha\gamma_i}^{ji}(t) + B_{\alpha\gamma_i}^{ji}(t)\big)x_{i\beta}^* \qquad \text{(B.13)}\\ &= \sum_{i:(j,i)\in E}(x_j(t)-x_j^*)^\top A^{ji}(t)x_i^* + \sum_{\alpha\in\mathcal{S}_j}(x_{j\alpha}(t)-x_{j\alpha}^*)\sum_{i:(j,i)\in E}(B_{\alpha\gamma_i}^{ji}(t) - A_{\alpha\gamma_i}^{ji}(t))\sum_{\beta\in\mathcal{S}_i}x_{i\beta}^*\\ &= \sum_{i:(j,i)\in E}(x_j(t)-x_j^*)^\top A^{ji}(t)x_i^* + \sum_{\alpha\in\mathcal{S}_j}(x_{j\alpha}(t)-x_{j\alpha}^*)\sum_{i:(j,i)\in E}(B_{\alpha\gamma_i}^{ji}(t) - A_{\alpha\gamma_i}^{ji}(t)) \qquad \text{(B.14)}\\ &= \sum_{i:(j,i)\in E}(x_j(t)-x_j^*)^\top A^{ji}(t)x_i^*. \qquad \text{(B.15)}\end{aligned}$$

The equation in (B.13) follows from applying Property 1 and the equation in (B.14) holds since $\sum_{\beta\in\mathcal{S}_i} x_{i\beta}^* = 1$ as a result of the strategy spaces being on the simplex. Finally, to see how (B.15) is obtained, observe that for each $\alpha \in \mathcal{S}_j$ the terms $\sum_{i:(j,i)\in E} A_{\alpha\gamma_i}^{ji}(t)$ and

$\sum_{i:(j,i)\in E} B^{ji}_{\alpha\gamma_i}(t)$ give the utility of player $j \in V$ in the games with payoffs $\{A^{ji}(t)\}_{(j,i)\in E}$ and $\{B^{ji}(t)\}_{(j,i)\in E}$ respectively under a joint pure strategy. Hence, by Property 3, the respective utilities are equal so that the difference is zero.

Finally, relating (B.15) back to (B.12), we conclude that the time-derivative is zero:

$$\begin{aligned}\frac{d}{dt}\sum_{i\in V}\langle x_i^*, y_i(t)\rangle &= -\sum_{j\in V}\sum_{i:(j,i)\in E}(x_j(t)-x_j^*)^\top B^{ji}(t)x_i^* \\ &= -\sum_{j\in V}\sum_{i:(j,i)\in E}(x_j(t)-x_j^*)^\top A^{ji}(t)x_i^* = 0.\end{aligned}$$

The final equality holds since x^* is an interior Nash equilibrium, which implies $u_{j\alpha}(x^*,t) = u_j(x^*,t)$ for all strategies $\alpha \in \mathcal{S}_j$ and any linear combination thereof.

Hence,

$$\frac{d\Phi(x^*,y(t))}{dt} = \frac{d}{dt}\sum_{i\in V} h_i^*(y_i(t)) + \frac{d}{dt}\sum_{i\in V}\langle x_i^*, y_i(t)\rangle = 0,$$

which implies that $\Phi(x^*, y(t))$ is time-invariant.

Finally, by Lemma D.2 of Mertikopoulos, Papadimitriou, and Piliouras [117], the time-invariance of Φ is sufficient to ensure that the flow ϕ^t of the differential equation $\dot{z}$ has bounded orbits. This finishes the proof.

B.3 `FTRL` Time-Average Result: Proof of Theorem 4.4.2

The outline of this proof is as follows. We begin by restating the relevant notation specialized to periodic zero-sum bimatrix games and then provide a more formal mathematical statement of the claim being proven. Following that we introduce a technical result regarding the bounded regret property of `FTRL` dynamics and state the implications that can be drawn from it. Finally, using the implications of bounded regret and properties of zero-sum bimatrix games we reach the conclusion.

Notation. Recall that we consider a periodic zero-sum bimatrix game for this result, which is a game that consists of a pair of players i and j and the bimatrix game between them at time $t \geq 0$ is described by the pair of payoffs $\{A^{ij}(t), A^{ji}(t)\} = \{A(t), -A^\top(t)\}$ and the sequence is periodic so that $\{A^{ij}(t+T), A^{ji}(t+T)\} = \{A^{ij}(t), A^{ji}(t)\}$ or equivalently $\{A(t+T), -A^\top(t+T)\} = \{A(t), -A^\top(t)\}$ for some finite period T and all time $t \geq 0$. Moreover, the bimatrix game at each time $t \geq 0$ is zero-sum which means that $u_i(x_i, x_j, t) + u_j(x_i, x_j, t) = 0$ for any strategy pair $x_i \in \mathcal{X}_i$ and $x_j \in \mathcal{X}_j$. Observe that we include the time-dependence t in the notation of player's utility to make it explicit the utility is time-dependent as a result of the time-varying payoff matrix. Finally, let the strategy pair $(x_i^*, x_j^*) \in \mathcal{X}_i \times \mathcal{X}_j$ denote the time-invariant Nash equilibrium.

Formal Statement of Result. Our goal is to prove that the time-average utility of each player converges to the time-average of the values of the games over a period. That is,

we seek to show

$$\lim_{t\to\infty}\frac{1}{t}\int_0^t u_i(x(\tau),\tau)d\tau=\frac{1}{T}\int_0^T u_i(x_i^*,x_j^*,\tau)d\tau=\bar{V}$$

and

$$\lim_{t\to\infty}\frac{1}{t}\int_0^t u_j(x(\tau),\tau)d\tau=\frac{1}{T}\int_0^T u_j(x_j^*,x_i^*,\tau)d\tau=-\bar{V},$$

where $\bar{V}:=\frac{1}{T}\int_0^T V(\tau)d\tau$ denotes the time-average of the values of the games over a period and $V(\tau)$ denotes the value of the game at time τ for any $\tau\geq 0$.

Bounded Regret Property. The proof of the above statement requires an intermediate technical result. The following result of Mertikopoulos, Papadimitriou, and Piliouras [117] states that regardless of what other players do in a polymatrix game (not necessarily zero-sum), if a player follows FTRL learning dynamics then the regret of the player is bounded. It is important to remark that this result directly applies to periodic zero-sum polymatrix games. This follows from the fact that there is no assumptions on the behavior of other players, so the dynamics from the game can be viewed as arising from the behavior of the other players.

Proposition B.3.1 (Theorem 3.1, [117]). *Let $h_{\max,i}=\max_{x_i\in\mathcal{X}_i}h_i(x_i)$ and $h_{\min,i}=\min_{x_i\in\mathcal{X}_i}h_i(x_i)$. If player $i\in V$ in a polymatrix game follows* FTRL *dynamics, then for every continuous trajectory of play $x_{-i}(t)$ of the opponents of player i the following regret bound holds:*

$$\max_{x_i\in\mathcal{X}_i}\frac{1}{t}\int_0^t\big[u_i(x_i,x_{-i}(\tau),\tau)-u_i(x(\tau),\tau)\big]d\tau\leq\frac{h_{\max,i}-h_{\min,i}}{t}.$$

Implications of Bounded Regret. Proposition B.3.1 ensures that the following bounds hold for the regret of player i:

$$\begin{aligned}\frac{1}{t}\int_0^t\big[u_i(x_i^*,x_j(\tau),\tau)-u_i(x(\tau),\tau)\big]d\tau&\leq\max_{x_i\in\mathcal{X}_i}\frac{1}{t}\int_0^t\big[u_i(x_i,x_j(\tau),\tau)-u_i(x(\tau),\tau)\big]d\tau\\&\leq\frac{h_{\max,i}-h_{\min,i}}{t}.\end{aligned}\tag{B.16}$$

Similarly, Proposition B.3.1 guarantees the following bounds hold for the regret of player j:

$$\begin{aligned}\frac{1}{t}\int_0^t\big[u_j(x_j^*,x_i(\tau),\tau)-u_j(x(\tau),\tau)\big]d\tau&\leq\max_{x_j\in\mathcal{X}_j}\frac{1}{t}\int_0^t\big[u_j(x_j,x_i(\tau),\tau)-u_j(x(\tau),\tau)\big]d\tau\\&\leq\frac{h_{\max,j}-h_{\min,j}}{t}.\end{aligned}\tag{B.17}$$

Observe that the lower bounds in (B.16) and (B.17) hold by replacing the maximizing argument over the strategy space of a player with a fixed strategy. In particular, the fixed strategy is taken to be the invariant Nash equilibrium strategy for player i or j.

Now, taking the limit as $t \to \infty$ of each side of (B.16) and using the zero-sum property of the bimatrix game at each time $\tau \geq 0$, we obtain the following:

$$\lim_{t\to\infty} \frac{1}{t} \int_0^t \big[u_i(x_i^*, x_j(\tau), \tau) - u_i(x(\tau), \tau)\big] d\tau = \lim_{t\to\infty} \frac{1}{t} \int_0^t \big[u_i(x_i^*, x_j(\tau), \tau) + u_j(x(\tau), \tau)\big] d\tau \leq 0. \tag{B.18}$$

Similarly, taking the limit as $t \to \infty$ of each side of (B.17) and using the zero-sum property of the bimatrix game at each time $\tau \geq 0$, we get that:

$$\lim_{t\to\infty} \frac{1}{t} \int_0^t \big[u_j(x_j^*, x_i(\tau), \tau) - u_j(x(\tau), \tau)\big] d\tau = \lim_{t\to\infty} \frac{1}{t} \int_0^t \big[u_j(x_j^*, x_i(\tau), \tau) + u_i(x(\tau), \tau)\big] d\tau \leq 0. \tag{B.19}$$

Time-Average Utility Convergence. We now proceed to show that

$$\lim_{t\to\infty} \frac{1}{t} \int_0^t u_i(x_i^*, x_j^*, \tau) d\tau \leq \lim_{t\to\infty} \frac{1}{t} \int_0^t u_i(x(\tau), \tau) d\tau \leq \lim_{t\to\infty} \frac{1}{t} \int_0^t u_i(x_i^*, x_j^*, \tau) d\tau. \tag{B.20}$$

The lower bound on the time-average utility of player i holds by the following analysis that is explained below:

$$\begin{aligned}
\lim_{t\to\infty} \frac{1}{t} \int_0^t u_i(x(\tau), \tau) d\tau &\geq \lim_{t\to\infty} \frac{1}{t} \int_0^t u_i(x_i^*, x_j(\tau), \tau) d\tau && \text{(B.21)} \\
&\geq \lim_{t\to\infty} \frac{1}{t} \int_0^t \min_{x_j \in \mathcal{X}_j} u_i(x_i^*, x_j, \tau) d\tau && \text{(B.22)} \\
&= \lim_{t\to\infty} \frac{1}{t} \int_0^t u_i(x_i^*, x_j^*, \tau) d\tau. && \text{(B.23)}
\end{aligned}$$

The inequality in (B.21) is a direct implication of (B.18). Moreover, the inequality in (B.22) is immediate by the fact that any fixed strategy of player j must give at least as much utility to player i as the strategy which minimizes the utility of player i. Finally, the last conclusion in (B.23) holds by the definition of a Nash equilibrium in a zero-sum bimatrix game.

The upper bound on the time-average utility of player i holds by the following similar analysis that is detailed below:

$$\begin{aligned}
\lim_{t\to\infty} \frac{1}{t} \int_0^t u_i(x(\tau), \tau) d\tau &\leq - \lim_{t\to\infty} \frac{1}{t} \int_0^t u_j(x_j^*, x_i(\tau), \tau) d\tau && \text{(B.24)} \\
&= \lim_{t\to\infty} \frac{1}{t} \int_0^t u_i(x_i(\tau), x_j^*, \tau) d\tau && \text{(B.25)} \\
&\leq \lim_{t\to\infty} \frac{1}{t} \int_0^t \max_{x_i \in \mathcal{X}_i} u_i(x_i, x_j^*, \tau) d\tau && \text{(B.26)} \\
&= \lim_{t\to\infty} \frac{1}{t} \int_0^t u_i(x_i^*, x_j^*, \tau) d\tau. && \text{(B.27)}
\end{aligned}$$

The inequality in (B.24) follows directly from (B.19) and the equality in (B.25) is a result of the zero-sum property of the game at each time $\tau \geq 0$. Furthermore, the inequality in (B.26) holds by the fact that the the strategy of player i that maximizes the utility must give at least as much utility as any fixed strategy. The last conclusion in (B.27) again holds by the definition of a Nash equilibrium in a zero-sum bimatrix game.

The preceding arguments prove that the claimed inequalities in (B.20) hold. Observe that the time-average of the utility values of player i at the invariant Nash equilibrium converge to the time-average of the values of the games over a period as a result of the periodic nature of the game and the fact that the utility value at any Nash equilibrium in a zero-sum bimatrix game is unique. That is,

$$\lim_{t\to\infty} \frac{1}{t}\int_0^t u_i(x_i^*, x_j^*, \tau)d\tau = \frac{1}{T}\int_0^T u_i(x_i^*, x_j^*, \tau)d\tau = \bar{V}.$$

Thus, the squeeze theorem applied to (B.20) allows us to conclude the statement given in (B.3):

$$\lim_{t\to\infty} \frac{1}{t}\int_0^t u_i(x(\tau), \tau)d\tau = \frac{1}{T}\int_0^T u_i(x_i^*, x_j^*, \tau)d\tau = \bar{V}$$

Finally, by the zero-sum property of the bimatrix game at each time $\tau \geq 0$, the statement given in (B.3) immediately follows from the equation above. That is,

$$\lim_{t\to\infty} \frac{1}{t}\int_0^t u_j(x(\tau), \tau)d\tau = \frac{1}{T}\int_0^T u_i(x_j^*, x_i^*, \tau)d\tau = -\bar{V}.$$

This finishes the proof.

B.4 `FTRL` Time-Average Result: Proof of Proposition 4.4.1

We now provide the proof of Proposition 4.4.1 stating that there exists periodic zero-sum bimatrix games satisfying Definition 4.2.2 in which the time-average strategies of `FTRL` dynamics fail to converge to the time-invariant Nash equilibrium.

To prove this result we construct a specific periodic zero-sum bimatrix game that is the basis of the counterexample. In order to demonstrate that the `FTRL` strategies may not converge to the time-invariant Nash equilibrium, we consider the regularization function that leads to the replicator dynamics. Then, for replicator dynamics in the constructed game, we prove that the strategies are symmetric about the half period of the game and consequently return to the initial condition in a period of the game so that the time-average of the strategies in the limit corresponds to the time-average of the strategies over a half-period of the game. Finally, we use this property to show that the choice of the period of the game can ensure that the time-average strategies cannot converge to the time-invariant Nash equilibrium.

Counterexample Construction. A periodic zero-sum bimatrix game between players i and j is described by a periodic sequence of payoffs where the game at time $t \geq 0$ is described by the pair of payoffs $\{A^{ij}(t), A^{ji}(t)\} = \{A(t), -A^\top(t)\}$. To obtain a

counterexample, we consider the periodic zero-sum bimatrix game defined by

$$A(t) = \gamma(t)A \quad \text{where} \quad A = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \quad \text{and} \quad \gamma(t) = \sin\Big(\frac{2\pi t}{T}\Big).$$

This game corresponds to a periodic version of matching pennies and the period of the game is T. The joint strategy (x_i^*, x_j^*) where $x_i^* = (1/2, 1/2)$ and $x_j^* = (1/2, 1/2)$ is the unique time-invariant Nash equilibrium of the game.

Recall that in periodic zero-sum bimatrix games between players i and j, we denote the utility of each player at time $t \geq 0$ under the joint strategy $x(t)$ by $u_i(x(t), t)$ and $u_j(x(t), t)$ to emphasize the dependence on the time-dependent payoffs which are given by $\{\gamma(t)A, -\gamma(t)A^\top\}$ in this construction. Furthermore, in this problem construction we denote the utility of each player with payoffs $\{A, -A^\top\}$ under the joint strategy $x(t)$ by $u_i(x(t))$ and $u_j(x(t))$ where A is defined at the matching pennies payoff matrix defined above.

The regularization function $h_i(x_i) = \sum_{\alpha \in \mathcal{S}_i} x_{i\alpha} \log x_{i\alpha}$ in `FTRL` dynamics gives rise to the replicator dynamics commonly studied in evolutionary game theory. For the periodic zero-sum bimatrix game under consideration, the replicator dynamics for any strategy $\alpha \in \mathcal{S}_i$ of player i are given by

$$\dot{x}_{i\alpha}(t) = x_{i\alpha}(t)\big[u_{i\alpha}(x(t), t) - u_i(x(t), t)\big] = \gamma(t)x_{i\alpha}(t)\big[u_{i\alpha}(x(t)) - u_i(x(t))\big] := \gamma(t)\dot{x}'_{i\alpha}(t).$$

Similarly, the replicator dynamics for any strategy $\alpha \in \mathcal{S}_j$ of player j are given by

$$\dot{x}_{j\alpha}(t) = x_{j\alpha}(t)\big[u_{j\alpha}(x(t), t) - u_j(x(t), t)\big] = \gamma(t)x_{j\alpha}(t)\big[u_{j\alpha}(x(t)) - u_j(x(t))\big] := \gamma(t)\dot{x}'_{j\alpha}(t).$$

We now analyze the time-average of the dynamics of any strategy for each player in this game and show that they do not correspond to the time-invariant Nash equilibrium.

Time-Average Strategies. We begin by showing that for each player $k \in \{i, j\}$ and strategy of the player $\alpha \in \mathcal{S}_k$,

$$x_{k\alpha}\big(\tfrac{T}{2} + t\big) = x_{k\alpha}\big(\tfrac{T}{2} - t\big). \tag{B.28}$$

To see this, observe that for each player $k \in \{i, j\}$ and strategy of the player $\alpha \in \mathcal{S}_k$ and some initial condition t_0,

$$x_{k\alpha}(t) = x_{k\alpha}(t_0) + \int_{t_0}^{t} \dot{x}_{j\alpha}(\tau)d\tau = x_{k\alpha}(t) + \int_{t_0}^{t} \sin\big(\tfrac{2\pi}{T}\tau\big)\dot{x}'_{k\alpha}(\tau)d\tau. \tag{B.29}$$

To prove the claim in (B.28), we show that both $x_{k\alpha}\big(\frac{T}{2} + t\big)$ and $x_{k\alpha}\big(\frac{T}{2} - t\big)$ satisfy the same ordinary differential equation and initial condition. That is, we invoke the fundamental theorem of ordinary differential equations which says the solutions exist and are unique so that the claim holds.

Indeed, from (B.29) and the fundamental theorem of calculus

$$\begin{aligned}\frac{d}{dt}x_{k\alpha}\big(\tfrac{T}{2}+t\big) &= \dot{x}'_{k\alpha}\big(\tfrac{T}{2}+t\big)\sin\big(\tfrac{2\pi}{T}\big(\tfrac{T}{2}+t\big)\big)\\ &= \dot{x}'_{k\alpha}\big(\tfrac{T}{2}+t\big)\sin\big(\pi+\tfrac{2\pi}{T}t\big)\\ &= -\dot{x}'_{k\alpha}\big(\tfrac{T}{2}+t\big)\sin\big(\tfrac{2\pi}{T}t\big).\end{aligned}$$

Similarly,

$$\begin{aligned}\frac{d}{dt}x_{k\alpha}\big(\tfrac{T}{2}-t\big) &= -\dot{x}'_{k\alpha}\big(\tfrac{T}{2}-t\big)\sin\big(\tfrac{2\pi}{T}\big(\tfrac{T}{2}-t\big)\big)\\ &= -\dot{x}'_{k\alpha}\big(\tfrac{T}{2}-t\big)\sin\big(\pi-\tfrac{2\pi}{T}t\big)\\ &= -\dot{x}'_{k\alpha}\big(\tfrac{T}{2}-t\big)\sin\big(\tfrac{2\pi}{T}t\big).\end{aligned}$$

We conclude that the functions $x_{k\alpha}\big(\frac{T}{2}+t\big)$ and $x_{k\alpha}\big(\frac{T}{2}-t\big)$ satisfy the same ordinary differential equation. That is, the functional form of the ordinary differential equation is the same in both expressions. Furthermore, $x_{k\alpha}\big(\frac{T}{2}+0\big) = x_{k\alpha}\big(\frac{T}{2}-0\big) = x_{k\alpha}\big(\frac{T}{2}\big)$ so they satisfy the same initial condition. Hence, invoking the uniqueness property of the fundamental theorem of ordinary differential equations, the claim given in (B.28) holds.

The property in (B.28) implies for each player $k \in \{i,j\}$ and strategy of the player $\alpha \in \mathcal{S}_k$ that

$$\lim_{t\to\infty}\frac{1}{t}\int_0^t x_{k\alpha}(\tau)d\tau = \frac{1}{T}\int_0^T x_{k\alpha}(\tau)d\tau = \frac{2}{T}\int_0^{T/2} x_{k\alpha}(\tau)d\tau.$$

That is, the limiting time-average strategy is equal to the time-average strategy over half a period of the periodic game.

Now recall that the time-invariant Nash equilibrium strategy is given by the joint strategy (x_i^*, x_j^*) where $x_i^* = (1/2, 1/2)$ and $x_j^* = (1/2, 1/2)$. Thus, to finish the proof we need to show for some player $k \in \{i,j\}$ and strategy of $\alpha \in \mathcal{S}_k$ that

$$\frac{2}{T}\int_0^{T/2} x_{k\alpha}(\tau)d\tau \neq \frac{1}{2}.$$

To see that the claim above holds, recall from (B.29) that

$$x_{k\alpha}(t) = x_{k\alpha}(0) + \int_0^t \sin\big(\tfrac{2\pi}{T}\tau\big)\dot{x}'_{k\alpha}(\tau)d\tau.$$

Observe that for any $\tau \geq 0$,

$$-1 \leq \sin\Big(\frac{2\pi}{T}\tau\Big) \leq 1 \quad \text{and} \quad -2 \leq \dot{x}'_{k\alpha}(\tau) \leq 2.$$

The previous expressions combine to imply

$$x_{k\alpha}(0) - 2t \leq x_{k\alpha}(t) \leq x_{k\alpha}(0) + 2t$$

and consequently

$$x_{k\alpha}(0) - \frac{T}{2} \leq \frac{2}{T}\int_0^{T/2} x_{k\alpha}(\tau)d\tau \leq x_{k\alpha}(0) + \frac{T}{2}.$$

Finally, suppose that $x_{k\alpha}(0) > 1/2$. Then, when $T < 2(x_{k\alpha}(0) - 1/2)$, the time-average of the strategy

$$\frac{2}{T}\int_0^{T/2} x_{k\alpha}(\tau)d\tau > \frac{1}{2}.$$

Analogously, suppose that $x_{k\alpha}(0) < 1/2$. Then, when $T < 2(1/2 - x_{k\alpha}(0))$, the time-average of the strategy

$$\frac{2}{T}\int_0^{T/2} x_{k\alpha}(\tau)d\tau < \frac{1}{2}.$$

Thus, unless the players initialize at the time-invariant Nash equilibrium strategy, there is a choice of the period T of the periodic zero-sum matrix such that the time-average of the strategies do not converge to the time-invariant Nash equilibrium. This completes the proof.

Appendix C

Omitted Proofs from Chapter 6

C.1 Proof of Lemma 6.2.1

We first describe how a behavioral plan σ_i can be transformed to a vector $x_i(h) \in \mathcal{X}_i$. For any $h \in \mathcal{X}_i$ we let $x_i(h) := \Pi_{(h,h') \in \mathcal{P}(h) \cap \mathcal{X}_i} \sigma_i(h, \alpha_{h'})$ where $\alpha_{h'}$ is the action $\alpha \in \mathcal{A}(h)$ such that $h' = \text{Next}(h, \alpha)$. We set $x_i(h) := 1$ for all $h \in \mathcal{H}_i$ with $\text{Prev}(h, i) = \varnothing$. Notice that by definition $U_1(\sigma) = \sum_{z \in \mathcal{Z}} x_1(z) \cdot p_1(z) \cdot x_2(z) = x_1^\top \cdot A_1^\Gamma \cdot x_2$ and respectively $U_2(\sigma) = \sum_{z \in \mathcal{Z}} x_2(z) \cdot p_2(z) \cdot x_1(z) = x_2^\top \cdot A_2^\Gamma \cdot x_1$.

Up next we show that all the constraints are satisfied. Consider the a state $h \in \mathcal{H}_i$ and the states $h' \in \text{Next}(h, \alpha, i)$ for some $\alpha \in \mathcal{A}(h)$. Notice that for each $h' \in \text{Next}(h, \alpha, i)$, $x_i(h') = x_i(h)\sigma_i(h, \alpha)$. This implies that $\sum_{\alpha \in \mathcal{A}(h)} x_i(\text{Next}(h, \alpha, i)) = x_i(h)$ since $\sum_{\alpha \in \mathcal{A}(h)} \sigma_i(h, \alpha) = 1$.

Now let $h_1, h_2 \in \mathcal{H}_i$ where $h_1 \in \text{Next}(h_1', \alpha, i)$, $h_2 \in \text{Next}(h_2', \alpha, i)$ and $I(h_1) = I(h_2)$. Consider the set $\mathcal{P}(h_1) \cap \mathcal{X}_i := \{p_1, \dots, p_k, h_1\}$ and $\mathcal{P}(h_2) \cap \mathcal{X}_i := \{q_1, \dots, q_k, h_2\}$. Due to the perfect recall property, $m = k$ and $I(p_\ell) = I(q_\ell)$. Thus, $x_i(h_1) = x_i(h_2)$.

Up next we show how a vector $x_i \in \mathcal{X}_i$ can be converted to a behavioral plan $\sigma_i \in \Sigma_i$. Let $\sigma_i(h, \alpha) := \frac{x_i(h')}{x(h)}$ for some $h' \in \text{Next}(h, \alpha, i)$. Notice that due the third constraint, $x_i(h') = x_i(h'')$ for all $h', h'' \in \text{Next}(h, \alpha, i)$ and thus $\sigma(h, \alpha)$ is well-defined. For $h \in \mathcal{H}_i$ let $h_\alpha \in \text{Next}(h, \alpha, i)$. By the third constraint we get that $\sum_{\alpha \in \mathcal{A}(h)} \sigma(h, \alpha) = 1$. Finally let $h_1, h_2 \in H_i$ with $I(h_1) = I(h_2)$ then $\sigma(h_1, \alpha) = \frac{x_i(h_1')}{x_i(h_1)}$ for some $h_1 \in \text{Next}(h_1, \alpha, i)$ and $\sigma(h_2, \alpha) = \frac{x_i(h_2')}{x_i(h_2)}$ for some $h_2 \in \text{Next}(h_2, \alpha, i)$. As a result, by the second constraint we get that $\sigma(h_1, \alpha) = \sigma(h_2, \alpha)$ for all $\alpha \in \mathcal{A}(h)$.

C.2 Proof of Lemma 6.3.1

We first describe how a behavioral plan $\sigma_u \in \Sigma_u$ can be transformed to a vector $x_u(h) \in \mathcal{X}_u$. If there exists a game Γ^{uv} with $(u, v) \in E$ such that $\text{Prev}^{\Gamma^{uv}}(h, u) = \varnothing$ we set $x_u(h) := 1$. Let us first verify that the above assignment is valid i.e. if $\text{Prev}^{\Gamma^{uv}}(h, u) = \varnothing$ for some $(u, v) \in E$ then $\text{Prev}^{\Gamma^{uv'}}(h, u) = \varnothing$ for all $(u, v') \in E$. Notice that $\mathcal{P}^{uv}(h) \cap \mathcal{X}_u = \{h\}$ and thus by the second constraint of Definition 6.3.5, $\mathcal{P}^{uv'}(h) \cap \mathcal{X}_u = \{h\}$ for all $(u, v') \in E$. Now for the remaining nodes $h \in \mathcal{H}_u$ we select an arbitrary two-player EFG Γ^{uv} ($(u, v) \in E$) containing the state h and set $x_u(h) := \Pi_{(h,h') \in \mathcal{P}^{uv}(h) \cap \mathcal{X}_u} \sigma_u(h, \alpha_{h'})$

where $\alpha_{h'}$ is the action $\alpha \in \mathcal{A}(h)$ such that $h' = \text{Next}^{\Gamma^{uv}}(h, \alpha, u)$. We again need to argue that $x_u(h)$ is independent of the arbitrary choice of the game Γ^{uv}. Let assume that state h also belongs in the two-player EFG $\Gamma^{uv'}$ for some $(u, v') \in E$. Again by the second constraint of Definition 6.3.3 we know that for the sets $\mathcal{P}^{uv}(h) \cap \mathcal{X}_u = \{p_1, \ldots, p_k, h\}$ and $\mathcal{P}^{uv'}(h) \cap \mathcal{X}_u = \{q_1, \ldots, q_m, h\}$ the following holds:

(1) $k = m$.

(2) $I(p_\ell) = I(q_\ell)$ for all $\ell \in \{1, \ldots, k\}$.

(3) $p_{\ell+1} \in \text{Next}^{\Gamma_{uv}}(p_\ell, \alpha, u)$ and $q_{\ell+1} \in \text{Next}^{\Gamma_{uv'}}(q_\ell, \alpha, u)$ for some action $\alpha \in \mathcal{A}(p_\ell)$.

Since $I(p_\ell) = I(q_\ell)$ means that $\sigma_u(p_\ell, \alpha) = \sigma_u(q_\ell, \alpha)$ for all $\alpha \in \mathcal{A}(p_\ell) = \mathcal{A}(q_\ell)$, we get that

$$\Pi_{(h,h') \in \mathcal{P}^{uv}(h) \cap \mathcal{X}_u} \sigma_u(h, \alpha_{h'}) = \Pi_{(h,h') \in \mathcal{P}^{uv'}(h) \cap \mathcal{X}_u} \sigma_u(h, \alpha_{h'})$$

Conversely, we show how a strategy in sequence form $x_u \in \mathcal{X}_u$ can be converted to behavioral plan $\sigma_u \in \Sigma_u$. Given a state $h \in \mathcal{H}_u$ we consider an edge $(u, v) \in E$ such that Γ^{uv} containing $h \in \mathcal{H}_u$ and set

$$\sigma(h, \alpha) := \frac{x_u(h')}{x_u(h)} \quad \text{for some } h' \in \text{Next}^{\Gamma^{uv}}(h, \alpha, u)$$

We first need to show that this is a valid probability distribution, $\sum_{h \in \mathcal{A}(h)} \sigma_u(h, \alpha) = 1$. Since $x_u \in \mathcal{X}_u^{\Gamma^{uv}}$, the second constraint of Definition 6.2.7 ensures that

$$\sum_{\alpha \in \mathcal{A}(h)} x_u(\text{Next}(h, \alpha, u)) = x_u(h)$$

The latter implies that $\sum_{h \in \mathcal{A}(h)} \sigma_u(h, \alpha) = 1$.

We now need to establish that $\sigma(h, \cdot)$ is independent of the selection of the edge $(u, v) \in E$. Let h be a state of the game $\Gamma^{uv'}$ for some $(u, v') \in E$. By constraint 2 of Definition 6.3.5, for any $h' \in \text{Next}^{\Gamma^{uv}}(h, \alpha, u)$ and $h'' \in \text{Next}^{\Gamma^{uv'}}(h, \alpha, u)$ we have that $x_u(h') = x_u(h'')$ and thus $\sigma(h, \alpha) = \frac{x_u(h'')}{x_u(h)}$.

Finally we need to argue that if $h_1, h_2 \in \mathcal{H}_u$ with $I(h_1) = I(h_2)$, then $\sigma(h_1, \alpha) = \sigma(h_2, \alpha)$ for all $\alpha \in \mathcal{A}(h_1) = \mathcal{A}(h_2)$. Let $\sigma(h_1, \alpha) = \frac{x_u(h_1')}{x_u(h)}$ for some $h_1' \in \text{Next}(h_1, \alpha, u)$ and $\sigma(h_1, \alpha) = \frac{x_u(h_1')}{x_u(h)}$ for some $h_2' \in \text{Next}(h_2, \alpha, u)$. Then by Constraint 3 of Definition 6.3.3 we get that $x(h_1') = x(h_2')$ and thus $\sigma(h_1, \alpha) = \sigma(h_2, \alpha)$.

C.3 Proof of Lemma 6.4.2

Let $\hat{x} := (\hat{x}_1, \ldots, \hat{x}_n)$ be an ϵ-symmetric Nash Equilibrium. Now consider the vector $x' \in \mathcal{X}$ defined as follows: $x_{u'} = \hat{x}_{u'}$ for all $u' \neq u$ and x'_u is an arbitrary vector in $\mathcal{X}_u$. By the definition of the ϵ-symmetric Nash Equilibrium we get that

$$\hat{x}^\top \cdot R \cdot \hat{x} - (x')^\top \cdot R \cdot \hat{x} \leq \epsilon$$

Notice that $(x')^\top \cdot R \cdot \hat{x} = -\sum_{v:(u,v)\in E}(x'_u)^\top \cdot A^{uv} \cdot \hat{x}_v - \sum_{u'\neq u}\sum_{v:(u',v)\in E} \hat{x}_{u'}^\top \cdot A^{u'v} \cdot \hat{x}_v$. Thus we get

$$-\sum_{v:(u,v)\in E}(x'_u)^\top \cdot A^{uv} \cdot \hat{x}_v + \sum_{v:(u,v)\in E}(\hat{x}_u)^\top \cdot A^{uv} \cdot \hat{x}_v \geq -\epsilon \quad \text{for all } x_u \in \mathcal{X}_u$$

Theorem 6.4.1 follows by repeating the same argument for all players $u \in V$.

C.4 Proof of Lemma 6.4.3

We first prove a simpler version of Lemma 6.4.3 where $x = y \in \mathcal{X}$.

Lemma C.4.1. *$x^\top \cdot R \cdot x = 0$ for all $x \in \mathcal{X}$.*

Proof. Consider a vector $x \in \mathcal{X}$. To simplify notation let $x \coloneqq (x_1, \ldots, x_n)$ where each vector $x_u \in \mathcal{X}_u$. Let $\sigma_u^x \in \Sigma$ denote the behavioral plan for player u constructed from the vector $x_u \in \mathcal{X}_u$ as described in Lemma 6.3.1. By the zero-sum property of Definition 6.3.4, we get that

$$\sum_{u\in V}\sum_{v:(u,v)\in E} U_u^{uv}(\sigma_u^x, \sigma_v^x) = 0$$

By Lemma 6.3.1 we get that $U^u(\sigma^x) = \sum_{v:(u,v)\in E} U_u^{uv}(\sigma_u^x, \sigma_v^x) = \sum_{v:(u,v)\in E} x_u^\top \cdot A^{uv} \cdot x_v$ meaning that

$$\sum_{u\in V}\sum_{v:(u,v)\in E} x_u^\top \cdot A^{uv} \cdot x_v = 0$$

As a result, we get that $x^\top \cdot R \cdot x = 0$. □

We will also utilize the following result:

Lemma C.4.2. *Consider a node $u \in V$ and its neighbors $\mathcal{N}_u = \{v_1, v_2, \ldots, v_k\}$. Let $x_u \in \mathcal{X}_u$ represent a mixed strategy for u and x_v a mixed strategy of the neighbor $v \in \mathcal{N}_u$. For any fixed collection $\{x_v\}_{v\in\mathcal{N}_u}$ the quantity*

$$\sum_{v\in\mathcal{N}_u} x_v^\top \cdot A^{vu} \cdot x_u + \sum_{v\in\mathcal{N}_u} x_u^\top \cdot A^{uv} \cdot x_v$$

remains constant over the range of x_u.

Proof. For any vector $x \coloneqq (x_1, \ldots, x_n) \in \mathcal{X}$, consider the vector $x' \in \mathcal{X}$ such that $x'_v = x_v$ for all $v \neq u$. By Lemma C.4.1 we get that

$$x^\top \cdot R \cdot x - (x')^\top \cdot R \cdot x' = 0$$

The latter directly implies that

$$\sum_{v\in\mathcal{N}_u} x_v^\top \cdot A^{vu} \cdot x_u + \sum_{v\in\mathcal{N}_u} x_u^\top \cdot A^{uv} \cdot x_v = \sum_{v\in\mathcal{N}_u} x_v^\top \cdot A^{vu} \cdot x'_u + \sum_{v\in\mathcal{N}_u} (x'_u)^\top \cdot A^{uv} \cdot x_v$$

for all $x_u, x'_u \in \mathcal{X}_u$. □

Proof of Lemma 6.4.3. Consider vectors $x, y \in \mathcal{X}$. Consider the vector $y' \in \mathcal{X}$ such that $y_v = y'_v$ for all $v \neq u$. We first show that

$$x^\top \cdot R \cdot y + y^\top \cdot R \cdot x = x^\top \cdot R \cdot y' + (y')^\top \cdot R \cdot x$$

Let $\mathcal{N}_u$ denote the neighbors of player $u \in V$,

$$\begin{aligned}
& x^\top \cdot R \cdot y + y^\top \cdot R \cdot x - x^\top \cdot R \cdot y' - (y')^\top \cdot R \cdot x \\
= \quad & \sum_{v \in \mathcal{N}_u} x_v^\top \cdot A^{vu} \cdot y_u + \sum_{v \in \mathcal{N}_u} x_u^\top \cdot A^{uv} \cdot y_v + \sum_{v \in \mathcal{N}_u} y_v^\top \cdot A^{vu} \cdot x_u + \sum_{v \in \mathcal{N}_u} y_u^\top \cdot A^{uv} \cdot x_v \\
& - \sum_{v \in \mathcal{N}_u} x_v^\top \cdot A^{vu} \cdot y'_u - \sum_{v \in \mathcal{N}_u} x_u^\top \cdot A^{uv} \cdot y'_v - \sum_{v \in \mathcal{N}_u} (y'_v)^\top \cdot A^{vu} \cdot x_u - \sum_{v \in \mathcal{N}_u} (y'_u)^\top \cdot A^{uv} \cdot x_v \\
= \quad & \sum_{v \in \mathcal{N}_u} x_v^\top \cdot A^{vu} \cdot y_u - \sum_{v \in \mathcal{N}_u} x_u^\top \cdot A^{uv} \cdot y_v - \sum_{v \in \mathcal{N}_u} x_v^\top \cdot A^{vu} \cdot y'_u - \sum_{v \in \mathcal{N}_u} (y'_u)^\top \cdot A^{uv} \cdot x_v \\
= \quad & 0
\end{aligned}$$

where the last equality follows by Lemma C.4.2. By gradually transforming vector y to vector x we get that $x^\top \cdot R \cdot y + y^\top \cdot R \cdot x = 2 \cdot x^\top \cdot R \cdot x = 0$. □

C.5 Proof of Lemma 6.4.4

Applying Lemma 1 of [141] to our setting, we obtain:

Lemma C.5.1 ([141]). *Let $\{x^t, \hat{x}^t\}$ be the sequences produced by* (s-OGD). *Then,*

$$\begin{aligned}
\sum_{t=1}^{T} (x^t)^\top \cdot R \cdot x^t - \min_{x \in \mathcal{X}} \sum_{t=1}^{T} x^\top \cdot R \cdot x^t \quad \leq \quad & \frac{\mathcal{D}^2}{\eta} + \frac{1}{2} \sum_{t=1}^{T} \|R \cdot x^t - R \cdot x^{t-1}\|^2 \\
& + \frac{1}{2} \sum_{t=1}^{T} \|x^t - \hat{x}^t\|^2 - \frac{1}{2\eta} \sum_{t=1}^{T} \left[\|\hat{x}^t - x^t\|^2 + \|\hat{x}^t - x^{t+1}\|^2 \right]
\end{aligned}$$

where $\mathcal{D}$ is the diameter of the treeplex polytope $\mathcal{X}$.

Setting $\eta = \min\{1/(8 \cdot \|R\|^2), 1\}$ in Lemma C.5.1 we get that

$$\begin{aligned}
& \sum_{t=1}^{T} (x^t)^\top \cdot R \cdot x^t - \min_{x \in \mathcal{X}} \sum_{t=1}^{T} x^\top \cdot R \cdot x^t \\
\leq \; & \frac{\mathcal{D}^2}{\eta} + \frac{1}{2} \sum_{t=1}^{T} \|R \cdot x^t - R \cdot x^{t-1}\|^2 - \frac{1}{4\eta} \cdot \sum_{t=1}^{T} \left[\|\hat{x}^t - x^t\|^2 + \|\hat{x}^t - x^{t+1}\|^2\right] \\
\leq \; & \frac{\mathcal{D}^2}{\eta} + \frac{1}{2} \sum_{t=1}^{T} \|R \cdot x^t - R \cdot x^{t-1}\|^2 - 2\|R\|^2 \cdot \sum_{t=1}^{T} \left[\|\hat{x}^t - x^t\|^2 + \|\hat{x}^t - x^{t+1}\|^2\right] \\
\leq \; & \frac{\mathcal{D}^2}{\eta} + \frac{\|R\|^2}{2} \sum_{t=1}^{T} \|x^t - x^{t-1}\|^2 - \|R\|^2 \cdot \sum_{t=1}^{T} \|x^t - x^{t-1}\|^2 \\
\leq \; & \frac{\mathcal{D}^2}{\eta}
\end{aligned}$$

Setting $\hat{x} = \sum_{s=1}^{T} x^s/T$ and using the fact that $(x^t)^\top \cdot R \cdot x^t = 0$ we get $\min_{x \in \mathcal{X}} x^\top \cdot R \cdot \hat{x} \geq -\frac{\mathcal{D}^2 \|R\|^2}{T}$.

C.6 Proof of Theorem 6.4.1

Let $\hat{x}$ be the time-average vector produced by (s−OGD). By Lemma 6.4.4, we have

$$\min_{x \in \mathcal{X}} x^\top \cdot R \cdot \hat{x} \geq -\Theta\left(\frac{\mathcal{D}^2 \|R\|^2}{T}\right)$$

Using the fact that $\hat{x}^\top \cdot R \cdot \hat{x} = 0$ we get that

$$\hat{x}^\top \cdot R \cdot \hat{x} \leq \min_{x \in \mathcal{X}} x^\top \cdot R \cdot \hat{x} + \Theta\left(\frac{\mathcal{D}^2 \|R\|^2}{T}\right)$$

meaning that $(\hat{x}, \hat{x})$ is a $\Theta\left(\frac{\mathcal{D}^2 \|R\|^2}{T}\right)$-approximate symmetric Nash Equilibrium of the symmetric game (R, R). By Lemma 6.4.2 we get that $\hat{x}$ is a $\Theta\left(\frac{\mathcal{D}^2 \|R\|^2}{T}\right)$-approximate NE for the original network zero-sum EFG.

C.7 Proof of Theorem 6.4.3

First of all, in the proof of this theorem and in the lemmas presented within the proof, let $\mathcal{X}^* := \{x^* \in \mathcal{X} : \min_{x \in \mathcal{X}} x^\top \cdot R \cdot x^* = 0\}$, which describes the set of symmetric Nash Equilibria.

In order to establish Theorem 6.4.3, we follow the approach and notation of [178], with minor modifications along the way to apply the steps to our setting. Applying Lemma 1 of [178]) to (s−OGD), we get the following lemma:

Lemma C.7.1 ([178]). *Let $\{x^t, \hat{x}^t\}_{t\geq 1}$ be the sequence of strategy vectors produced by* (s-OGD) *for $\eta \leq 1/8\|R\|^2$. Then,*

$$\eta(R\cdot x^t)^\top(x^t - x) \leq D_\psi(x,\hat{x}^t) - D_\psi(x,\hat{x}^{t+1}) - D_\psi(\hat{x}^{t+1},x^t) - \frac{15}{16}D_\psi(x^t,\hat{x}^t) + \frac{1}{16}D_\psi(\hat{x}^t,x^{t-1})$$

Since for OGD we have that $D_\psi(x) = \frac{1}{2}\|x\|^2$, we can write the above inequality as:

$$2\eta(R\cdot x^t)^\top(x^t - x) \leq \|\hat{x}^t - x\|^2 - \|\hat{x}^{t+1} - x\|^2 - \|\hat{x}^{t+1} - x^t\|^2 - \frac{15}{16}\|x^t - \hat{x}^t\|^2 + \frac{1}{16}\|\hat{x}^t - x^{t-1}\|^2 \tag{C.1}$$

To simplify notation let $x^* := \Pi_{\mathcal{X}^*}(\hat{x}^t) \in \mathcal{X}^*$ meaning that x^* is a symmetric Nash Equilibrium for the symmetric game (R, R) and let us apply Equation C.1 with $x = x^*$. Now the LHS of Equation C.1 takes the following form

$$\begin{aligned} 2\eta(x^t)^\top \cdot R^T \cdot (x^t - x^*) &= -2\eta(x^t)^\top \cdot R^T \cdot x^* && ((x^t)^\top \cdot R^T \cdot x^t = 0) \\ &= -2\eta(x^*)^\top \cdot R \cdot x^t \\ &= -2\eta(x^t)^\top \cdot R \cdot x^* && \text{(by Lemma 6.4.3)} \\ &\geq 0 \end{aligned}$$

where the last inequality follows by the fact that (x^*, x^*) is a symmetric Nash Equilibrium of the game (R, R). Since the LHS of Equation C.1 is greater or equal to 0 we get that,

$$\|\hat{x}^{t+1} - \Pi_{\mathcal{X}^*}(\hat{x}^t)\|^2 \leq \|\hat{x}^t - \Pi_{\mathcal{X}^*}(\hat{x}^t)\|^2 - \|\hat{x}^{t+1} - x^t\|^2 - \frac{15}{16}\|x^t - \hat{x}^t\|^2 + \frac{1}{16}\|\hat{x}^t - x^{t-1}\|^2$$

By definition, the left hand side of the above is bounded below by $\mathrm{dist}^2(\hat{x}^{t+1}, \mathcal{X}^*)$. Thus, we have the following inequality,

$$\mathrm{dist}^2(\hat{x}^{t+1}, \mathcal{X}^*) \leq \mathrm{dist}^2(\hat{x}^t, \mathcal{X}^*) - \|\hat{x}^{t+1} - x^t\|^2 - \frac{15}{16}\|x^t - \hat{x}^t\|^2 + \frac{1}{16}\|\hat{x}^t - x^{t-1}\|^2 \tag{C.2}$$

Now, we define $\Theta^t := \|\hat{x}^t - \Pi_{\mathcal{X}^*}(\hat{x}^t)\|^2 + \frac{1}{16}\|\hat{x}^t - x^{t-1}\|^2$ and $\xi^t := \|\hat{x}^{t+1} - x^t\|^2 + \|x^t - \hat{x}^t\|^2$ and rewrite Equation C.2 as follows,

$$\Theta^{t+1} \leq \Theta^t - \frac{15}{16}\xi^t \tag{C.3}$$

As in [178], we now lower bound ξ^t by a quantity related to $\mathrm{dist}^2(\hat{x}^{t+1}, \mathcal{X}^*)$ which will then give us a convergence rate for Θ^t. To do so we need to establish a property that is known as saddle-point metric subregularity ([178]).

Lemma C.7.2. *(Saddle-Point Metric Subregularity (SP-MS)) For any $x, x' \in \mathcal{X} \setminus \mathcal{X}^*$,*

$$\sup_{x'\in\mathcal{X}} \frac{(R\cdot x)^\top(x - x')}{\|x - x'\|} \geq c \cdot \|x - \Pi_{\mathcal{X}^*}(x)\|$$

for some game-dependent constant $c > 0$.

We present the proof of Lemma C.7.2 in Section C.8. To this end, we remark that once the proof of Lemma C.7.2 is established, the proof of Theorem 6.4.3 follows by the analysis of [178]. For the sake of completeness, we conclude the section with this analysis.

Lemma C.7.3 ([178]). *If the parameter η in* (s-OGD) *is selected less than $1/8\|R\|^2$, then for any $t \geq 0$ and $x' \in \mathcal{X}$ with $x' \neq \hat{x}^{t+1}$,*

$$\|\hat{x}^{t+1} - x^t\|^2 + \|x^t - \hat{x}^t\|^2 \geq \frac{32}{81}\eta^2 \frac{\left[(R \cdot \hat{x}^{t+1})^\top (\hat{x}^{t+1} - x')\right]_+^2}{\|\hat{x}^{t+1} - x'\|^2}$$

where $[a]_+ \coloneqq \max\{a, 0\}$, and similarly, for $x' \neq x^{t+1}$,

$$\|\hat{x}^{t+1} - x^{t+1}\|^2 + \|x^t - \hat{x}^{t+1}\|^2 \geq \frac{32}{81}\eta^2 \frac{\left[(R \cdot x^{t+1})^\top (x^{t+1} - x')\right]_+^2}{\|x^{t+1} - x'\|^2}$$

Now taking the telescoping sum of Equation C.3 over t, we get:

$$\Theta^1 \geq \Theta^1 - \Theta^T \geq \frac{15}{16}\sum_{t=1}^{T-1} \xi^t \geq \frac{15}{16}\sum_{t=1}^{T-1} \left(\|\hat{x}^{t+1} - x^t\|^2 + \|x^t - \hat{x}^t\|^2\right) \geq \frac{15}{32}\sum_{t=2}^{T-1} \|x^t - x^{t-1}\|^2$$

where the final inequality follows due to strong convexity of $\frac{1}{2}\|x\|^2$. Now, since the rightmost term is a summation of nonnegative terms and is upper bounded by a finite constant, we have that $\|x^{t-1} - x^t\| \to 0$ as $T \to \infty$. Thus, x^t converges to a point as $T \to \infty$. In addition, due to Theorem 6.4.1, we know that the time-average value of the iterates converge to a Nash Equilibrium. Combining these two observations, we can thus conclude that x^t indeed converges to a Nash Equilibrium in the last-iterate sense.

To show the explicit rate of convergence, we will require a few additional observations. First, note that the following inequality holds for Equation C.3:

$$\|\hat{x}^{t+1} - x^t\|^2 \leq \xi^t \leq \frac{16}{15}\Theta^t \leq \cdots \leq \frac{16}{15}\Theta^1 \tag{C.4}$$

Then we have:

$$\begin{aligned}
\xi^t &\geq \frac{1}{2}\|\hat{x}^{t+1} - x^t\|^2 + \frac{1}{2}(\|\hat{x}^{t+1} - x^t\|^2 + \|x^t - \hat{x}^t\|^2) \\
&\geq \frac{1}{2}\|\hat{x}^{t+1} - x^t\|^2 + \frac{16\eta^2}{81}\sup_{x' \in \mathcal{X}} \frac{\left[(R \cdot \hat{x}^{t+1})^\top (\hat{x}^{t+1} - x')\right]_+^2}{\|\hat{x}^{t+1} - x'\|^2} && \text{(Lemma C.7.3)} \\
&\geq \frac{1}{2}\|\hat{x}^{t+1} - x^t\|^2 + \frac{16\eta^2 C^2}{81}\|\hat{x}^{t+1} - \Pi_{\mathcal{X}^*}(\hat{x}^{t+1})\|^2 && \text{(SP-MS condition)} \\
&\geq \min\left\{\frac{16\eta^2 C^2}{81}, \frac{1}{2}\right\}\left(\|\hat{x}^{t+1} - x^t\|^2 + \|\hat{x}^{t+1} - \Pi_{\mathcal{X}^*}(\hat{x}^{t+1})\|^2\right) && \text{(Equation C.4)} \\
&= C_2\Theta^{t+1}
\end{aligned}$$

Now, we can show the explicit convergence rate as follows. Combining the above inequality with Equation C.3, we obtain

$$\Theta^{t+1} \leq \Theta^t - C_2\Theta^{t+1} \tag{C.5}$$

This immediately implies that $\Theta^{t+1} \leq (1 + C_2)^{-1}\Theta^t$. By iteratively expanding the right hand side of the inequality, we can equivalently write:

$$\Theta^t \leq (1 + C_2)^{-t+1}\Theta^1 \leq 2\Theta^1(1 + C_2)^{-t} \tag{C.6}$$

Next, notice that Θ^1 is precisely $\mathrm{dist}^2(\hat{x}^1, \mathcal{X}^*)$. Moreover, by using the triangle inequality, we can write:

$$\begin{aligned}\mathrm{dist}^2(x^t, \mathcal{X}^*) &\leq \|x^t - \Pi_{\mathcal{X}^*}(\hat{x}^{t+1})\|^2 \\ &\leq 2\|\hat{x}^{t+1} - \Pi_{\mathcal{X}^*}(\hat{x}^{t+1})\|^2 + 2\|\hat{x}^{t+1} - x^t\|^2 \\ &\leq 32\Theta^{t+1} \leq 32\Theta^t\end{aligned}$$

Combining this observation with Equation C.6 we get that

$$\mathrm{dist}^2(x^t, \mathcal{X}^*) \leq 64\mathrm{dist}^2(x^1, \mathcal{X}^*)(1 + C_2)^{-t}$$

where $C_2 = \min\left\{\frac{16\eta^2C^2}{81}, \frac{1}{2}\right\}$, which completes the proof of Theorem 6.4.3.

C.8 Proof of Lemma C.7.2

Lemma C.7.2 follows easily from Lemma C.8.1, the proof of which is presented in Section C.9.

Lemma C.8.1. *For any $x \in \mathcal{X}$ the following holds:*

$$-\min_{x' \in \mathcal{X}} x'^{\top} R \cdot x \geq c \cdot \|x - \Pi_{\mathcal{X}^*}(x)\|. \tag{C.7}$$

for some game-dependent constant $c \in (0, 1)$.

Proof of Lemma C.7.2. Consider the LHS of the inequality in Lemma C.8.1 and note that $-\min_{x' \in \mathcal{X}} x'^{\top} R\bar{x} = 0$ if and only if $\bar{x} \in \mathcal{X}^*$.

Let $\mathcal{D}$ denote the diameter of $\mathcal{X}$ which is assumed to be finite. Then,

$$\begin{aligned}
\max_{x' \in \mathcal{X}} \frac{(R \cdot x)^\top (x - x')}{\|x - x'\|} &\geq \max_{x' \in \mathcal{X}} \frac{1}{\mathcal{D}} (R \cdot x)^\top (x - x') \\
&= \frac{1}{\mathcal{D}} \max_{x' \in \mathcal{X}} x^\top R^\top (x - x') \\
&= \frac{1}{\mathcal{D}} \max_{x' \in \mathcal{X}} [x^\top R^\top x - x^\top R^\top x'] \\
&= \frac{1}{\mathcal{D}} \max_{x' \in \mathcal{X}} [-x^\top R^\top x'] && (x^\top R^\top x = 0) \\
&= -\frac{1}{\mathcal{D}} \min_{x' \in \mathcal{X}} x^\top R^\top x' \\
&= -\frac{1}{\mathcal{D}} \min_{x' \in \mathcal{X}} x'^\top R x \\
&\geq \frac{c}{\mathcal{D}} \|x - \Pi_{\mathcal{X}^*}(x)\| && \text{(Lemma C.8.1)}
\end{aligned}$$

□

C.9 Proof of Lemma C.8.1

The proof of this lemma follows the basic steps in the proof of Theorem 5 in [178], with some necessary modifications. We remind the reader that for the purposes of the proof, we defined the set of symmetric Nash Equilibria as $\mathcal{X}^* = \{x^* : \min_{x \in \mathcal{X}} x^\top \cdot R \cdot x^* = 0\}$. The proof is split into several auxiliary lemmas/claims, which can then be combined to show the required result.

Lemma C.9.1. *The set $\mathcal{X}^*$ is a polytope.*

Proof. Let $x^* \in \mathcal{X}^*$ then $\min_{x \in \mathcal{X}} x^\top \cdot R \cdot x^* = 0$. Since $\mathcal{X}$ is a polytope the minimum value is attained in one of the vertices of polytope $\mathcal{X}$, the set of which is denoted by $\mathcal{V}(\mathcal{X})$. Thus

$$\min_{x \in \mathcal{X}} x^\top \cdot R \cdot x^* = \min_{v \in \mathcal{V}(\mathcal{X})} v^\top \cdot R \cdot x^* = 0$$

As a result, the set $\mathcal{X}^*$ can be equivalently described as the set of vector $x^* \in \mathcal{X}$ that additionally satisfy $v_i^\top \cdot R \cdot x^* \geq 0$ for all vertices $v_i \in \mathcal{V}(\mathcal{X})$. □

Let us describe $\mathcal{X}$ in the following polytopal form:

$$\mathcal{X} := \{x : \ \alpha_i^\top \cdot x \leq \beta_i \text{ for } i = 1, \dots L\}$$

where L is a positive integer. Consider also the following polytopal form of the set $\mathcal{X}^*$ as

$$\mathcal{X}^* := \{x^* \in \mathcal{X} : \ c_i^\top \cdot x^* \geq 0 \text{ for } i = 1, \dots, K\}$$

where $c_i := v_i^\top \cdot R$ with v_i denoting the i-th vertex of polytope $\mathcal{X}$ and K denotes the number of different vertices.

Now fix a specific $x \in \mathcal{X} \setminus \mathcal{X}^*$ and let $x^* := \Pi_{\mathcal{X}^*}(x)$. The vector x^* satisfies some of the polytopal constraints with equality. These constraints are called *tight*, and without loss of generality we can assume that

- $\alpha_i^\top \cdot x^* = \beta_i$ for $i = 1, \ldots, \ell$
- $c_i^\top \cdot x^* = 0$ for $i = 1, \ldots, k$

Lemma C.9.2. *The vector $x \in \mathcal{X}$ violates at least one tight constraint of the form $\{c_i^\top \cdot x = 0$ for $i = 1, \ldots, k\}$.*

Proof. Let assume that $\{c_i^\top \cdot x = 0$ for $i = 1, \ldots, k\}$. Since $x \notin \mathcal{X}^*$ there exists at least one vertex $v \in \mathcal{V}(\mathcal{X})$ such that $v^\top \cdot R \cdot x < 0$. The latter implies that there exits $x' \in \mathcal{X}$ lying in line segment between x and x^* such that $v^\top \cdot R \cdot x \geq 0$ for all vertices $v \in \mathcal{V}(\mathcal{X})$. The latter implies that $x' \in \mathcal{X}^*$ which contradicts with the fact that $x^* = \Pi_{\mathcal{X}^*}(x)$. □

Now, note that the normal cone of $\mathcal{X}^*$ at x^* is

$$\mathcal{N}_{x^*} = \left\{x' - x^* : \quad x^* = \Pi_{\mathcal{X}^*}(x')\right\}$$

From a standard result in linear programming literature [178], we know that the normal cone can be written in the following form:

$$\mathcal{N}_{x^*} = \left\{\sum_{i=1}^{\ell} p_i \cdot \alpha_i + \sum_{i=1}^{k} q_i \cdot c_i \text{ for some } p_i, q_i \geq 0\right\}$$

Again, following the steps of [178], we have the following claim:

Claim C.9.1. *For any $x \in \mathcal{X}$ such that $x^* = \Pi_{x \in \mathcal{X}^*}(x)$ the vector $x - x^*$ belongs in the set*

$$\mathcal{M}_{x^*} = \left\{\sum_{i=1}^{\ell} p_i \cdot \alpha_i + \sum_{i=1}^{k} q_i \cdot c_i : \quad p_i, q_i \geq 0, \alpha_j^\top \cdot \left(\sum_{i=1}^{\ell} p_i \cdot \alpha_i + \sum_{i=1}^{k} q_i \cdot c_i\right) \leq 0\right\}$$

Proof. As mentioned previously, we know that $x - x^*$ belongs in the normal cone of x^*, $\mathcal{N}_{x^*}$. Thus it can be expressed as $\sum_{i=1}^{\ell} p_i \cdot \alpha_i + \sum_{i=1}^{k} q_i \cdot c_i$ with $p_i, q_i \geq 0$. As such, we need only additionally show that $x - x^*$ satisfies the following:

$$\alpha_i^\top (x - x^*) \leq 0, \quad \forall i \in 1, \ldots, \ell$$

Notice that for all $i = 1, \ldots, \ell$, we have:

$$\begin{aligned} \alpha_i^\top (x - x^*) &= (\alpha_i^\top x^* - b_i) + \alpha_i^\top (x - x^*) && (i\text{-th constraint is tight at } x^*) \\ &= \alpha_i^\top (x^* + x - x^*) - b_i \\ &= \alpha_i^\top x - b_i \leq 0 && (x \in \mathcal{X}) \end{aligned}$$

□

Claim C.9.2. *$x - x^*$ can be written as $\sum_{i=1}^{\ell} p_i \cdot \alpha_i + \sum_{i=1}^{k} q_i \cdot c_i$ with $0 \leq p_i, q_i \leq C' \|x - x^*\|$ for all i and some problem-dependent constant $C' < \infty$.*

Proof. Note that $\frac{x-x^*}{\|x-x^*\|} \in \mathcal{M}_{x^*}$ because $0 \neq x - x^* \in \mathcal{M}_{x^*}$ and $\mathcal{M}_{x^*}$ is a cone. Furthermore, $\frac{x-x^*}{\|x-x^*\|} \in \{v \in \mathbb{R}^M : \|v\|_\infty \leq 1\}$. Thus, $\frac{x-x^*}{\|x-x^*\|} \in \mathcal{M}_{x^*} \cap \{v \in \mathbb{R}^M : \|v\|_\infty \leq 1\}$, which is a bounded subset of the cone $\mathcal{M}_{x^*}$.

We will argue that there exists large enough $C' > 0$ such that:

$$\left\{ \sum_{i=1}^{\ell} p_i \cdot \alpha_i + \sum_{i=1}^{k} q_i \cdot c_i : 0 \leq p_i, q_i \leq C', \forall i \right\} \supseteq \mathcal{M}_{x^*} \cap \{v \in \mathbb{R}^M : \|v\|_\infty \leq 1\} \coloneqq \mathcal{P}.$$

First note that $\mathcal{P}$ is a polytope. For every vertex $\hat{v}$ of $\mathcal{P}$, the smallest C' such that $\hat{v}$ belongs to the left-hand side set above is the solution to the following linear program:

$$\begin{aligned} \min_{p_i, q_i, C'_{\hat{v}}} \quad & C'_{\hat{v}} \\ \text{s.t.} \quad & \hat{v} = \sum_{i=1}^{\ell} p_i \cdot \alpha_i + \sum_{i=1}^{k} q_i \cdot c_i, \quad 0 \leq p_i, q_i \leq C'_{\hat{v}}. \end{aligned}$$

Since $\hat{v} \in \mathcal{M}_{x^*}$, this LP is always feasible and admits a finite solution $C'_{\hat{v}} < \infty$. Now, let $C' = \max_{\hat{v} \in \mathcal{V}(\mathcal{P})}$ where $\mathcal{V}(\mathcal{P})$ is the set of all vertices of $\mathcal{P}$. Then, since any $v \in \mathcal{P}$ can be expressed as a convex combination of points in $\mathcal{V}(\mathcal{P})$, v can thus be expressed as $\sum_{i=1}^{\ell} p_i \cdot \alpha_i + \sum_{i=1}^{k} q_i \cdot c_i$ where $0 \leq p_i, q_i \leq C'$. As a result, $\frac{x-x^*}{\|x-x^*\|}$ can be written as $\sum_{i=1}^{\ell} p_i \cdot \alpha_i + \sum_{i=1}^{k} q_i \cdot c_i$ where $0 \leq p_i, q_i \leq C'$, so it follows that $x - x^*$ can be written as: $\sum_{i=1}^{\ell} p_i \cdot \alpha_i + \sum_{i=1}^{k} q_i \cdot c_i$ where $0 \leq p_i, q_i \leq C'\|x - x^*\|$.

□

Now, again following [178], we can piece together all of the auxiliary results to show Lemma C.8.1. Define $A_i \coloneqq \alpha_i^\top (x - x^*)$ and $C_i \coloneqq c_i^\top (x - x^*)$. By Claim C.9.2, we can write $x - x^*$ as $\sum_{i=1}^{\ell} p_i \cdot \alpha_i + \sum_{i=1}^{k} q_i \cdot c_i$ where $0 \leq p_i, q_i \leq C'\|x - x^*\|$. Thus:

$$\sum_{i=1}^{\ell} p_i \cdot A_i + \sum_{i=1}^{k} q_i \cdot C_i = \left(\sum_{i=1}^{\ell} p_i \cdot \alpha_i + \sum_{i=1}^{k} q_i \cdot c_i \right)^\top (x - x^*) = \|x - x^*\|^2$$

Moreover, since $x - x^* \in \mathcal{M}_{x^*}$ by Claim C.9.1, we have

$$\sum_{i=1}^{\ell} p_i \cdot A_i = \sum_{i=1}^{\ell} p_i \cdot \alpha_i \leq 0$$

and

$$\sum_{i=1}^{k} q_i \cdot C_i \leq \left(\max_{i \in \{1,\ldots,k\}} C_i \right) \sum_{i=1}^{k} q_i \leq \left(\max_{i \in \{1,\ldots,k\}} C_i \right) kC'\|x - x^*\|$$

The first inequality follows because $p_i \geq 0$. The second inequality follows because $\max_{i \in \{1,\ldots,k\}} C_i > 0$ (by Lemma C.9.2) and $0 \leq q_i \leq C'\|x - x^*\|$.

Combining the above, we obtain:

$$\max_{i\in\{1,\dots,k\}} C_i \geq \frac{1}{kC'}\|x - x^*\|$$

Now, note that:

$$\max_{i\in\{1,\dots,k\}} C_i = \max_{i\in\{1,\dots,k\}} (c_i^\top x - d_i) \leq \max_{i\in\{1,\dots,|\mathcal{V}(\mathcal{X})|\}} (c_i^\top x - d_i) = \max_{x'\in\mathcal{X}} (x'^\top R x)$$

where the last equality follows from the formulation of problem constraints in the proof of Lemma C.9.1. Finally, by combining the last two statements, we can conclude that

$$-\min_{x'\in\mathcal{X}} (x'^\top R x) \geq \frac{1}{kC'}\|x - x^*\|.$$

Here k and C' only depend on the set of tight constraints at x^*. There are only finitely many sets of tight constraints, so there exists a constant $C > 0$ such that $-\min_{x'\in\mathcal{X}}(x'^\top R x) \geq \frac{1}{kC'}\|x - x^*\|$ holds for all x and x^*, completing the proof.