

Evolutionary Game Theory Squared: Evolving Agents in Endogenously Evolving Zero-Sum Games

Stratis Skoulakis*

Tanner Fiez†

Ryann Sim*

Georgios Piliouras*‡

Lillian Ratliff†‡

Abstract

The predominant paradigm in evolutionary game theory and more generally online learning in games is based on a clear distinction between a population of *dynamic agents* that interact given a *fixed, static game*. In this paper, we move away from the artificial divide between dynamic agents and static games, to introduce and analyze a large class of competitive settings where both the agents and the games they play evolve strategically over time. We focus on arguably the most archetypal game-theoretic setting—zero-sum games (as well as network generalizations)—and the most studied evolutionary learning dynamic—replicator, the continuous-time analogue of multiplicative weights. Populations of agents compete against each other in a zero-sum competition that itself evolves adversarially to the current population mixture. Remarkably, despite the chaotic coevolution of agents and games, we prove that the system exhibits a number of regularities. First, the system has *conservation laws* of an information-theoretic flavor that couple the behavior of all agents and games. Secondly, the system is *Poincaré recurrent*, with effectively all possible initializations of agents and games lying on recurrent orbits that come arbitrarily close to their initial conditions infinitely often. Thirdly, the *time-average agent behavior and utility converge* to the Nash equilibrium values of the *time-average game*. Finally, we provide a polynomial time algorithm to efficiently predict this time-average behavior for any such coevolving network game.

1 Introduction

The problem of analyzing evolutionary learning dynamics in games is of fundamental importance in several fields such as evolutionary game theory (Sandholm, 2010), online learning in games (Cesa-Bianchi and Lugosi, 2006; Nisan et al., 2007), and multi-agent systems (Shoham and Leyton-Brown, 2008). The dominant paradigm in each area is that of evolutionary agents adapting to each others behavior. In other words, the dynamism of the environment of each agent is driven by the other agents, whereas the rules of interaction between the agents, that is, the game, is static. This separation between *evolving agents* and a *static game* is so standard that it typically goes unnoticed, however, this fundamental restriction does not allow us to capture many applications of interest. In artificial intelligence (Wang et al., 2019; Garciarena et al., 2018; Costa et al., 2019a; Miikkulainen et al., 2019; Wu et al., 2019; Stanley and Miikkulainen, 2002) as well as biology, sociology, and economics (Stewart and Plotkin, 2014; Tilman et al., 2020, 2017; Bowles et al., 2003; Weitz et al., 2016), the rules of interaction can themselves adapt to the collective history of the agent behavior. For example, in adversarial learning and curriculum learning (Huang et al., 2011; Bengio et al., 2009), the difficulty of the game can increase over time by exactly focusing on the settings where the agent has performed the weakest. Similarly, in biology or economics, if a particular advantageous strategy is used exhaustively by

*Singapore University of Technology and Design

†University of Washington

‡Joint last authors

§Mail: efstratios@sutd.edu.sg, fiezt@uw.edu, ryann_sim@mymail.sutd.edu.sg, georgios@sutd.edu.sg, ratliff1@uw.edu

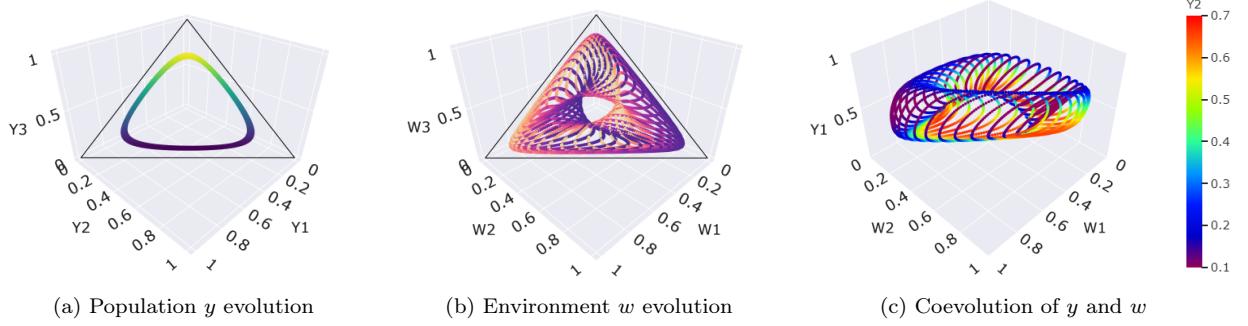


Figure 1: Poincaré recurrence in a time-evolving generalized Rock-Paper-Scissors model.

agents, then its relative advantages typically dissipate over time (negative frequency-dependent selection, see Heino et al. 1998), which once again drives the need for innovation and exploration.

In all these cases, the game itself stops being a passive object that the agents act upon, but instead is best thought of as an algorithm itself. Similar to online learning algorithms employed by agents, the game itself may have a memory/state that encodes history. However, unlike online learning algorithms that receive a history or sequence of payoff vectors and output the current behavior (e.g., a probability distribution over actions), an algorithmic game receives as input a history or sequence of agents’ behavior and outputs a new payoff matrix. Hence, learning and games are “dual” algorithmic objects which are coupled in their evolution (Figure 1).

How does one even hope to analyze evolutionary learning in time-evolving games? Once we move away from the safe haven of static games, we lose our prized standard methodology that roughly consists of two steps: i) compute/understand the equilibria of the given game (e.g., Nash, correlated, etc., see Nash 1951; Aumann 1974) and their properties; ii) connect the behavior of learning dynamics to a target class of equilibria (e.g., convergence). Indeed, the only prior work to ours, namely by Mai et al. (2018), which considers games larger than 2×2 , focused on a specific payoff matrix structure based on Rock-Paper-Scissors (RPS) and argued recurrent behavior via a tailored argument that was explicitly designed for the dynamical system in question with no clear connections to game theory. We revisit this problem and find a new systematic game-theoretic analysis that generalizes to arbitrary network zero-sum games.

Contributions

We provide a general framework for analyzing learning agents in time-evolving zero-sum games as well as rescaled network generalizations thereof. To begin, we develop a novel *reduction* that takes as input time-evolving games and reduces them to a game-theoretic graph that generalizes both graphical zero-sum games and evolutionary zero-sum games. In this generalized but static game, evolving agents and evolving games represent different types of nodes (nodes with and without self-loops) in a graph connected by edge 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 agents interacting in a static game, the mathematical tools of game theory and dynamical systems theory become available. This allows us to perform a general algorithmic analysis of commonly studied systems from machine learning and biology previously requiring individualized treatment.

From an algorithmic learning perspective, we focus on the most studied evolutionary learning dynamic: replicator, the continuous-time analogue of the multiplicative weights update. Remarkably, despite the chaotic coevolution of agents and games that forces agents 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 agents and games lying on recurrent orbits that come arbitrarily close to their initial conditions infinitely often

(Figure 1). As a crucial component of this result, we demonstrate the dynamics obey information-theoretic *conservation laws* that couple the behavior of all agents and games (Figure 3). Moreover, while the system never equilibrates, the conservation laws allow us to prove the *time-average behavior and utility of the agents converge* to the time-average Nash of their evolving games with bounded regret (Figures 11 and 14 in the Appendix). Finally, we provide a *polynomial time algorithm* that predicts these time-average quantities.

Related Work and Technical Novelty

Our work relates with the rich previous literature studying the emerging recurrent behavior of replicator dynamics in (network) zero-sum games (Piliouras et al., 2014; Piliouras and Shamma, 2014; Boone and Piliouras, 2019; Mertikopoulos et al., 2018; Nagarajan et al., 2020; Perolat et al., 2020). All these results are based on the surprising fact that the Kullback-Leibler (KL) divergence between the dynamics produced by the replicator equation and the Nash Equilibrium remains constant. Unfortunately this proof technique is an immediate dead-end for time-evolving zero-sum games (cf. Figure 7 in Appendix C which shows the KL divergence between the (evolving) strategies and (evolving) Nash equilibrium for the central RPS example from Mai et al. 2018). In particular, it is not even clear what the static concept of a Nash equilibrium means in this context. Despite this, Mai et al. (2018) managed to prove recurrence via constructing an invariant function for this specific example. However, their invariant function relies on the symmetries of the RPS game and has no deeper interpretation or obvious generalization. A key contribution of our work is the development of a novel characterization of a general class of time-evolving games that possess a number of regularities including recurrence, which we demonstrate by deriving an information theoretic invariant. In particular, this allows us to not only generalize the recurrence results of time-evolving games to a class with much richer and complex interactions than the one studied in Mai et al. (2018), but also provides a naturally interpretable invariant in such time-evolving games.

2 Preliminaries and Definitions

In this section, we formalize the concept of polymatrix games, define the replicator dynamics for this class of games, and provide background material on dynamical systems that is relevant to our results.

Polymatrix Games

An N -player *polymatrix game* is defined using an undirected graph $G = (V, E)$ where V corresponds to the set of agents (or players) and E corresponds to the set of edges between agents in which a *bimatrix game* is played between the endpoints (Cai and Daskalakis, 2011). Each agent $i \in V$ has a set of actions $\mathcal{A}_i = \{1, \dots, n_i\}$ 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 $\mathcal{X}_i = \Delta^{n_i-1} = \{x_i \in \mathbb{R}_{\geq 0}^{n_i} : \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha} = 1\}$ where $x_{i\alpha}$ denotes the probability mass on action $\alpha \in \mathcal{A}_i$. 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 $A_{\alpha\beta}^{ij}$ for $(\alpha, \beta) \in \mathcal{A}_i \times \mathcal{A}_j$ 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 an agent $i \in V$ plays a game defined by A^{ii} against itself. The *utility* or *payoff* of agent $i \in V$ 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 agent 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,

$$u_i(x) = \sum_{j:(i,j) \in E} x_i^\top A^{ij} x_j. \quad (1)$$

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 $\alpha \in \mathcal{A}_i$. 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*.

A common notion of equilibrium behavior in game theory is that of a Nash equilibrium, which is defined as a mixed strategy profile $x^* \in \mathcal{X}$ such that for each player $i \in V$,

$$u_i(x_i^*, x_{-i}^*) \geq u_i(x_i, x_{-i}^*), \quad \forall x_i \in \mathcal{X}_i. \quad (2)$$

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

Replicator Dynamics

In polymatrix games, *replicator dynamics* (Sandholm, 2010) for each $i \in V$ are given by

$$\dot{x}_{i\alpha} = x_{i\alpha}(u_{i\alpha}(x) - u_i(x)), \quad \forall \alpha \in \mathcal{A}_i. \quad (3)$$

We suppress the explicit dependence on time t in the system and do so throughout where clear from context to simplify notation. Moreover, we consider initial conditions on the interior of the simplex. The replicator dynamics are equivalently given in vector form for each $i \in V$ by the system

$$\dot{x}_i = x_i \cdot \left(\sum_{j:(i,j) \in E} A^{ij} x_j - \left(\sum_{j:(i,j) \in E} x_i^\top A^{ij} x_j \right) \cdot \mathbf{1} \right), \quad (4)$$

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) 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 $x_i = (x_{i1}, \dots, x_{in_i}) \mapsto (\ln \frac{x_{i2}}{x_{i1}}, \dots, \ln \frac{x_{in_i}}{x_{i1}})$ for each player $i \in V$.

Review of Topology of Dynamical Systems

We now review some concepts from dynamical systems theory that will help us prove Poincaré recurrence. Further background material can be found in the book of Alongi and Nelson (2007).

Flows: Consider a differential equation $\dot{x} = f(x)$ on a topological space X . The existence and uniqueness theorem for ordinary differential equations guarantees that there exists a unique continuous function $\phi : \mathbb{R} \times X \rightarrow X$, which is termed the *flow*, that satisfies (i) $\phi(t, \cdot) : X \rightarrow X$ —often denoted $\phi^t : X \rightarrow X$ —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 $x \in X$, and (iii) for each $x \in X$, $\frac{d}{dt}|_{t=0} \phi(t, x) = f(x)$. Since the replicator dynamics are Lipschitz continuous, a unique flow ϕ of the replicator dynamics exists.

Conservation of Volume: The flow ϕ of a system of 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. 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 (Poincaré, 1890). Given a flow ϕ^t on a topological space X , a point $x \in X$ 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 2.1 (Poincaré Recurrence (Poincaré, 1890)). *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 $\Omega(\phi^t) = Z$.*

3 Studying Doubly Evolutionary 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 takes the form of a population y of *species* which evolve dynamically in time as a function of itself and some *environment* parameters w whose evolution, in turn, depends on y . We now review models from each application and then 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.

Doubly Evolutionary Behavior in AI and ML

Evolutionary game theory methods for training generative adversarial networks commonly exhibit time-evolving dynamic behavior and there is a pair of predominant doubly evolutionary process models (Costa et al., 2020; Wang et al., 2019; Garciarena et al., 2018; Costa et al., 2019a; Miikkulainen et al., 2019). In the first formulation, Wang et al. (2019) 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 (Garciarena et al., 2018; Costa et al., 2019a; Miikkulainen et al., 2019). We connect further to AI and ML applications in the discussion where we highlight exciting future directions.

Doubly Evolutionary Behavior in Biology

There are also two common 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 a setting where the payoff matrix $A(w)$ depends on an environment variable w which, in turn, depends on the population via $\dot{w} = w(1-w)G(y)$ where $G(y)$ is a feedback mechanism describing when environmental degradation or enhancement occurs as a function of y (Weitz et al., 2016; Tilman et al., 2020, 2017; Lade et al., 2013); e.g., in Weitz et al. (2016), $G(y)$ takes the form $\theta y - (1-y)$ for some $\theta > 0$ which represents the ratio of the enhancement rate to 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, Mai et al. (2018) argue that the environment 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. The adopted model of the dynamic behavior with initial conditions on the interior of the simplex for both w and y is given for each $i \in \{1, \dots, n\}$ by

$$\begin{aligned}\dot{w}_i &= w_i \sum_{j=1}^n w_j (y_j - y_i) \\ \dot{y}_i &= y_i ((P(w)y)_i - y^\top P(w)y)\end{aligned}\tag{5}$$

where $P(w) = P + \mu W$ for $\mu > 0$ with P defined as the generalized 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 the environmental variations matrix

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

Reducing Time-Evolving RPS to a Polymatrix Game

Mai et al. (2018) studied the dynamical system in (5) and showed it exhibits a special type of cyclic behavior: *Poincaré recurrence*. By capturing the evolution of the environment (dynamics of the payoff matrix) as additional players that dynamically change their strategies, we reduce the coevolution of w and y to a *static polymatrix game* of greater dimensionality (greater number of players). Given this reduction, Theorem 4.1, which establishes the Poincaré recurrence of replicator dynamics in rescaled zero-sum polymatrix games, immediately captures the results of Mai et al. (2018) (see Corollary 4.1).

Proposition 3.1. *The time-evolving generalized rock-paper-scissors game from (5) is equivalent to replicator dynamics in a two-player rescaled zero-sum polymatrix game.*

Proof Sketch (see Appendix B.1 for formal proof). The initial condition $w(0)$ is on the interior of the simplex and $\sum_{i=1}^n \dot{w}_i = 0$. Consequently, $\sum_{i=1}^n w_i(0) = \sum_{i=1}^n w_i(t) = 1$, and we obtain

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

which is the replicator equation of a node w in a polymatrix game with payoff matrix $A^{wy} = -I$. Using a similar decomposition, we reformulate the y dynamics:

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

This corresponds to the replicator equation of node y playing against itself with $A^{yy} = P$ and against w with $A^{yw} = \mu I$. The game is rescaled zero-sum with $\eta_y = 1$ and $\eta_w = \mu$. \square

Generalized Reduction

The previous reduction generalizes to a class of time-evolving games 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 $k \in \{1, \dots, n_w\}$. Environments coevolve with only populations and not other environments, while any population coevolves only with environments and itself. Let \mathcal{N}_k^w be the set of populations which coevolve with w_k and \mathcal{N}_ℓ^y be the set of environments which coevolve with y_ℓ . The time-evolving dynamics for each environment k and population ℓ are given componentwise by

$$\dot{w}_{k,i} = w_{k,i} \sum_{\ell \in \mathcal{N}_k^w} \sum_j w_{k,j} ((A^{k,\ell} y_\ell)_i - (A^{k,\ell} y_\ell)_j), \quad (6)$$

$$\dot{y}_{\ell,i} = y_{\ell,i} ((P_\ell(w)y_\ell)_i - y_\ell^\top P_\ell(w)y_\ell), \quad (7)$$

where $P_\ell(w) = P_\ell + \sum_{k \in \mathcal{N}_\ell^y} W^{\ell,k}$ with $P_\ell \in \mathbb{R}^{n \times n}$ and $W^{\ell,k} \in \mathbb{R}^{n \times n}$ 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 proof of this result is in Appendix B.1.

Theorem 3.1. *Any time-evolving system defined by the dynamics in (6-7) is equivalent to replicator dynamics in a polymatrix game.*

The expressive power we gain from this reduction permits us to efficiently describe and characterize coevolutionary processes of higher complexity than past work since we can return to the familiar territory of analyzing dynamic agents in static games. In what follows we focus on providing theoretical results for the subclass of time-evolving systems which reduce to a rescaled zero-sum game. However, this reduction is of independent interest since it can prove useful for future work analyzing the class of general-sum games after the behavior of network zero-sum games and rescaled generalizations are well understood.

4 Poincaré Recurrence

In this section, we show that the 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 4.1. *The replicator dynamics given in (3) are Poincaré recurrent in any N -player rescaled zero-sum polymatrix game that has an interior Nash equilibrium.*

Boone and Piliouras (2019), the closest known result, prove replicator dynamics are Poincaré recurrent in N -player pairwise zero-sum polymatrix games with an interior Nash equilibria, which requires $A^{ij} = -(A^{ji})^\top$ for every $(i, j) \in E$. Our extension to N -player rescaled zero-sum polymatrix games is a far more general characterization of the Poincaré recurrence of replicator dynamics since there are no explicit restrictions on the edge games and the polymatrix game itself need not even be strictly zero-sum. The significance of this result is further enhanced by the connection developed in Section 3 between a class of time-evolving games and N -player rescaled zero-sum polymatrix games. As a concrete example, given the reduction of Proposition 3.1, Theorem 4.1 recovers the work of Mai et al. (2018).

Corollary 4.1. *The time-evolving generalized rock-paper-scissors game in (5) is Poincaré recurrent.*

The following proof sketch provides intuition that highlights our analysis techniques and we defer the finer points to Appendix B.2. It is worth noting that the technical results we prove in order to show the system is Poincaré recurrent, namely volume preservation and the bounded orbits property, are themselves independently important as they provide conservation laws that couple the behavior of agents. In fact, they are fundamental to showing that while the system never equilibrates, the time-average dynamics and utility converge to the Nash equilibrium and its utility.

Overview of Proof Methods

To prove Poincaré recurrence, we need to show the flow corresponding to the system of ordinary differential equations in (3) is volume preserving and has bounded orbits (cf. Theorem 2.1). Notice that the flow of (3) always has bounded orbits since $x_{i\alpha} \geq 0$ and $\sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}(t) = 1 \forall i \in V$, however proving the volume preserving property is not as straightforward. To show volume preservation, we transform the dynamics via a *canonical transformation*. Indeed, we prove Poincaré recurrence of the flow of a system of ordinary differential equations that is diffeomorphic to the flow of the replicator equation. Given $x \in \mathcal{X}$, consider the transformed variable $z \in \mathbb{R}^{n_1 + \dots + n_N - N}$ defined by

$$z_i = \left(\ln \frac{x_{i2}}{x_{i1}}, \dots, \ln \frac{x_{in_i}}{x_{i1}} \right), \quad \forall i \in V. \quad (8)$$

Given the vector z_i , the components of x_i are 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 $\alpha \in \mathcal{A}_i$ and all $i \in V$ by

$$\dot{z}_{i\alpha} = F_{i\alpha}(z) = \frac{\dot{x}_{i\alpha}}{x_{i\alpha}} - \frac{\dot{x}_{i1}}{x_{i1}} = \sum_{j \in V} \sum_{\beta \in \mathcal{A}_j} (A_{\alpha\beta}^{ij} - A_{1\beta}^{ij}) e^{z_{j\beta}} / \sum_{\ell=1}^{n_j} e^{z_{j\ell}}. \quad (9)$$

Observe that $F_{i1} = 0$, meaning $z_{i1} = 0$ for all time. To show Poincaré recurrence of (3), we prove two key properties: (i) the flow of \dot{z} is volume preserving, meaning the trace Jacobian of the respective 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 from Theorem 2.1.

Conservation of Volume

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 (9), is volume preserving.

Lemma 4.1. *For any N -player rescaled zero-sum polymatrix game,*

$$\text{tr}(DF(z)) = \sum_{i \in V} \sum_{\alpha \in \mathcal{A}_i} \frac{d}{dz_{i\alpha}} F_{i\alpha}(z) = 0.$$

The proof of Lemma 4.1 crucially relies on the fact the self-loops are antisymmetric, $(A^{ii})^\top = -A^{ii}$.

Bounded Orbits

In order to prove that the orbits from any initial interior point $z(0)$ are bounded, we show that for any initial interior point $x(0)$, the orbit produced by the replicator dynamics stays on the interior of the simplex, that is, there exists a fixed parameter $\epsilon > 0$ such that for any agent $i \in V$ and strategy $\alpha \in \mathcal{A}_i$, $\epsilon \leq x_{i\alpha} \leq 1 - \epsilon$. Then, $|z_{i\alpha}|$ is clearly bounded since $z_{i\alpha} = \ln(x_{i\alpha}/x_{1\alpha})$.

Lemma 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 admits an interior Nash Equilibrium x^* , then $\Phi(t) = \sum_{i \in V} \sum_{\alpha \in \mathcal{A}_i} \eta_i x_{i\alpha}^* \ln x_{i\alpha}$ is time-invariant, meaning $\Phi(t) = \Phi(0)$ for $t \geq 0$. Hence, orbits from any interior initial condition $x(0)$ remain on the interior of the simplex.*

From the preceding discussion, Lemma 4.2 guarantees orbits from any interior initial condition $z(0)$ remain bounded. The proof of Lemma 4.2 is the primary novelty in the proof of Theorem 4.1 and the techniques may be of independent interest. 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.

Cai and Daskalakis (2011) 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 $\alpha \in \mathcal{A}_j$ and any linear combination.

Proof of Theorem 4.1. The proof follows directly from Lemma 4.1, Lemma 4.2, and Theorem 2.1. Indeed, the dynamics in (9) are Poincaré recurrent since from Lemma 4.1 they are volume preserving and from Lemma 4.2 the orbits are bounded. This property in the cumulative payoff space carries over to the dynamics in the strategy space from (3) since the transformation is a diffeomorphism. \square

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 its connections to notions of equilibrium and regret. We prove that the enduring system behavior is guaranteed to satisfy a number of desirable game-theoretic metrics of consistency and optimality. Moreover, we design a polynomial time algorithm able to predict this behavior. The proofs of results from this section are in Appendix B.3.

While the replicator dynamics exhibit complex dynamics and never equilibrate 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 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) is such that i) $\lim_{t \rightarrow \infty} \frac{1}{t} \int_0^t x(\tau) d\tau = x^*$ and ii) $\lim_{t \rightarrow \infty} \frac{1}{t} \int_0^t u_i(x(\tau)) d\tau = u_i(x^*)$.*

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 (Hofbauer et al., 2009). We remark that our proof crucially relies on Lemma 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.

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 (Cai et al., 2016). The following result extends this formulation to rescaled zero-sum polymatrix games.

Theorem 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:*

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

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 5.1. *Any player following the replicator dynamics (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 ,*

$$\text{Reg}_i(t) := \max_{y \in \mathcal{X}_i} \frac{1}{t} \int_0^t [u_i(y, x_{-i}(s)) - u_i(x(s))] ds \leq \frac{\Omega_i}{t}.$$

The proof of this proposition mirrors closely more general arguments in Mertikopoulos et al. (2018). A standalone derivation is provided in the appendix sake of completeness.

6 Simulations

The goal of this section is to experimentally verify some of the key results, and to highlight other empirically observed properties outside the established theoretical results.¹

Theorem 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 1 from the time-evolving generalized RPS game of Section 3 are cyclic in nature. Specifically, Figure 1c shows the coevolution of the system for a fixed initial

¹Code is available at github.com/ryanndelion/egt-squared

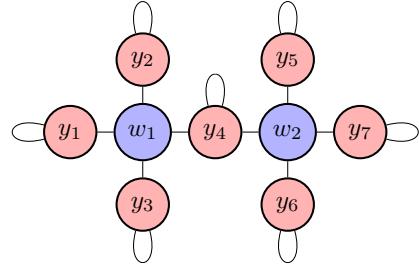


Figure 2: Two clusters of nodes that join together to form a ‘butterfly’ structure. Self-loops represent RPS self-play games, while edges between nodes represent $(I, -I)$. The *red* nodes denote a population of species, while the *blue* nodes stand for an environment.

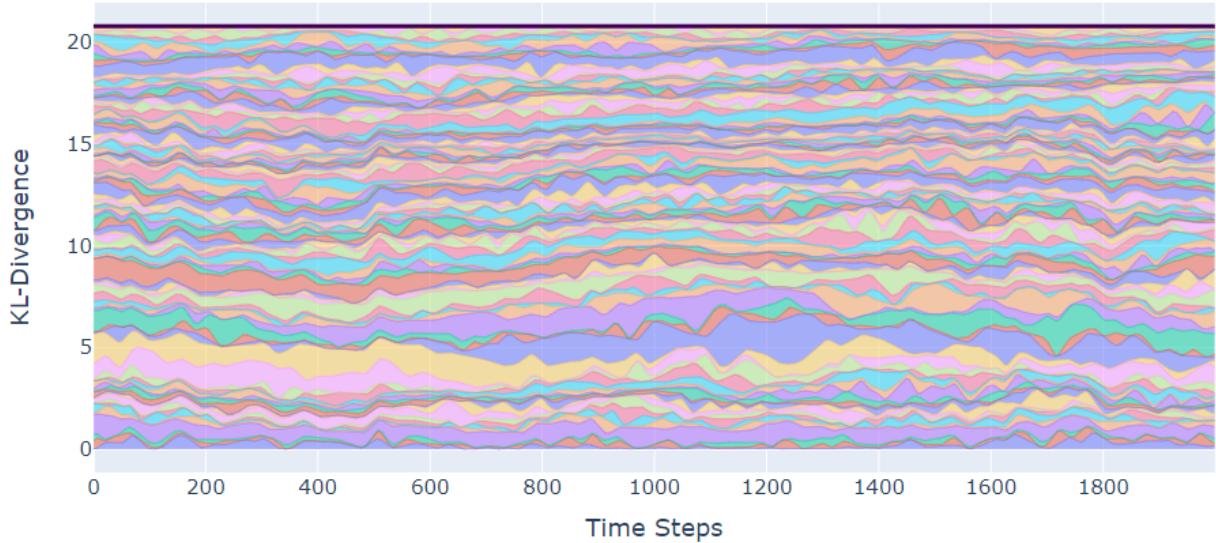


Figure 3: Weighted KL divergence for 25 cluster (100 player) time-evolving zero-sum game (for details see Appendix C.2).

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. In the supplementary code, we provide an animation of these dynamics for a range of initial conditions. 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 5.1.

Lemma 4.2 shows that for any *rescaled zero-sum game* there is a constant of motion, namely $\Phi(t)$. It is easy to see from the definition of $\Phi(t)$ that a weighted sum of KL-divergences between the strategy vectors produced by replicator dynamics and an interior Nash Equilibrium is also a constant of motion (see Corollary B.1 in the Appendix). We simulated an extension to the game depicted in Figure 2 in which many ‘butterfly’ clusters are joined in a toroid shape. Figure 3 depicts our claim: although each agent specific divergence term $\eta_i \text{KL}(x_i^* || x_i(t))$ fluctuates, the weighted sum $\sum_{i \in V} \eta_i \text{KL}(x_i^* || x_i(t))$ remains constant.

To generate Figure 4, we scale-up the game structure from Mai et al. (2018) 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

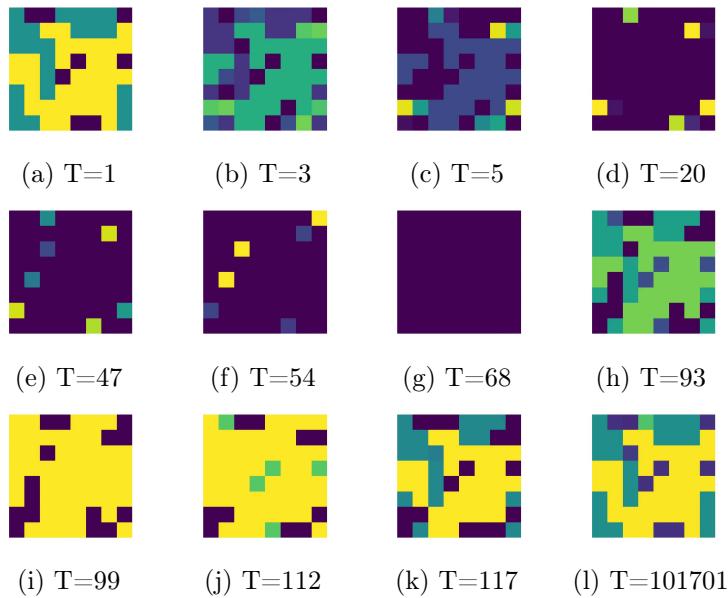


Figure 4: Sequence of Pikachu images showing approximate recurrence in an 8×8 zero-sum polymatrix game, where the changing color of each pixel on the grid represents the strategy of the player over time.

the expected time to see recurrence in such a system is exponential in the number of agents. As observed in Figure 4, 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. Further details on the simulation methodology and additional experiments can be found in Appendix C.

7 Discussion

We show 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. A future direction of theoretical research is on the 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 (Czarnecki et al., 2020; Balduzzi et al., 2019). 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 we cannot capture the system fully, we can create the following 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. For such systems, our results provide some theoretical justification for the preservation of diversity and for the satisfying empirical performance.

To conclude, we briefly touch on the connection to progressive training of generative adversarial networks (Kar-

ras et al., 2018). The basic idea is to start the training process with small generator and discriminator networks and, over time, periodically add layers to the networks of higher dimension to grow the resolution of the generated images. This process causes the zero-sum game (between generator and discriminator) to evolve with time. Importantly, as a consequence, the equilibrium is not fixed in the game. For instance, we can capture behavior of this process as a time evolving game in our model: the base game matrix P is sparse and of high dimension; as the environment w changes in time the nonzero values in the time-evolving payoff $P(w)$ ‘turn on’, progressively making the matrix dense. Despite the critical nature of the above AI architectures, which are both based on the guided evolution of zero-sum games, no model of them exists in the literature.

Acknowledgments

Stratis Skoulakis gratefully acknowledges NRF 2018 Fellowship NRF-NRFF2018-07. Tanner Fiez acknowledges support from the DoD NDSEG Fellowship. Ryann Sim gratefully acknowledges support from the SUTD President’s Graduate Fellowship (SUTD-PGF). Lillian Ratliff is supported by NSF CAREER Award number 1844729 and an Office of Naval Research Young Investigator Award. Georgios Piliouras gratefully acknowledges support from grant PIE-SGP-AI-2018-01, NRF2019-NRF-ANR095 ALIAS grant and NRF 2018 Fellowship NRF-NRFF2018-07.

References

- Erol Akçay and Joan Roughgarden. The evolution of payoff matrices: providing incentives to cooperate. *Royal Society B: Biological Sciences*, 278(1715):2198–2206, 2011.
- Abdullah Al-Dujaili, Tom Schmiedlechner, Una-May O’Reilly, et al. Towards distributed coevolutionary gans. *arXiv preprint arXiv:1807.08194*, 2018.
- John M Alongi and Gail Susan Nelson. *Recurrence and topology*, volume 85. American Mathematical Society, 2007.
- Robert J Aumann. Subjectivity and correlation in randomized strategies. *Journal of mathematical Economics*, 1(1):67–96, 1974.
- James P Bailey and Georgios Piliouras. Multiplicative weights update in zero-sum games. In *ACM Conference on Economics and Computation*, pages 321–338, 2018.
- D Balduzzi, S Racaniere, J Martens, J Foerster, K Tuyls, and T Graepel. The mechanics of n-player differentiable games. In *International Conference on Machine Learning*, volume 80, pages 363–372, 2018.
- D Balduzzi, M Garnelo, Y Bachrach, W Czarnecki, J Pérolat, M Jaderberg, and T Graepel. Open-ended learning in symmetric zero-sum games. In *International Conference on Machine Learning*, volume 97, pages 434–443, 2019.
- Chris Bauch and David Earn. Vaccination and the theory of games. *National Academy of Sciences*, 101:13391–4, 10 2004.
- Chris T. Bauch, Alison P. Galvani, and David J. D. Earn. Group interest versus self-interest in smallpox vaccination policy. *National Academy of Sciences*, 100(18):10564–10567, 2003.
- Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *International Conference on Machine Learning*, pages 41–48, 2009.
- 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*, pages 85–99, 2019.
- Samuel Bowles, Jung-Kyoo Choi, and Astrid Hopfensitz. The co-evolution of individual behaviors and social institutions. *Journal of Theoretical Biology*, 223(2):135–147, 2003.

- Yang Cai and Constantinos Daskalakis. On minmax theorems for multiplayer games. In *Symposium of Discrete Algorithms*, pages 217–234, 2011.
- Yang Cai, Ozan Candogan, Constantinos Daskalakis, and Christos H. Papadimitriou. Zero-sum polymatrix games: A generalization of minmax. *Mathematics of Operations Research*, 41(2):648–655, 2016.
- 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*, pages 921–930, 2019.
- Matteo Cavaliere, Sean Sedwards, Corina E Tarnita, Martin A Nowak, and Attila Csikász-Nagy. Prosperity is associated with instability in dynamical networks. *Journal Theoretical Biology*, 299:126–138, 2012.
- Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.
- 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*, pages 807–834, 2019.
- Michael H Cortez and Stephen P Ellner. Understanding rapid evolution in predator-prey interactions using the theory of fast-slow dynamical systems. *The American Naturalist*, 176(5):E109–E127, 2010.
- Victor Costa, Nuno Lourenço, João Correia, and Penousal Machado. Coegan: evaluating the coevolution effect in generative adversarial networks. In *Genetic and Evolutionary Computation Conference*, pages 374–382, 2019a.
- Victor Costa, Nuno Lourenço, and Penousal Machado. Coevolution of generative adversarial networks. In *International Conference on the Applications of Evolutionary Computation*, pages 473–487. Springer, 2019b.
- 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*, pages 562–577. Springer, 2020.
- Wojciech Czarnecki, Gauthier Gidel, Brendan Tracey, Karl Tuyls, Shayegan Omidshafiei, David Balduzzi, and Max Jaderberg. Real world games look like spinning tops. In *Advances in Neural Information Processing Systems*, 2020.
- Constantinos Daskalakis, Andrew Ilyas, Vasilis Syrgkanis, and Haoyang Zeng. Training gans with optimism. In *International Conference on Learning Representations*, 2018.
- Benoit Duvocelle, Panayotis Mertikopoulos, Mathias Staudigl, and Dries Vermeulen. Learning in time-varying games. *arXiv preprint arXiv:1809.03066*, 2018.
- Ilan Eshel, Ethan Akin, et al. Coevolutionary instability of mixed nash solutions. *Journal of Mathematical Biology*, 18(2):123–133, 1983.
- Yoav Freund and Robert E Schapire. Adaptive game playing using multiplicative weights. *Games and Economic Behavior*, 29(1-2):79–103, 1999.
- Daniel Friedman. Evolutionary games in economics. *Econometrica*, pages 637–666, 1991.
- Alison Galvani, Timothy Reluga, and Gretchen Chapman. Long-standing influenza vaccination policy is in accord with individual self-interest but not with the utilitarian optimum. *National Academy of Sciences*, 104:5692–7, 04 2007.
- Unai Garcíarena, Roberto Santana, and Alexander Mendiburu. Evolved gans for generating pareto set approximations. In *Genetic and Evolutionary Computation Conference*, pages 434–441, 2018.
- Gauthier Gidel, Reyhane Askari Hemmat, Mohammad Pezeshki, Rémi Le Priol, Gabriel Huang, Simon Lacoste-Julien, and Ioannis Mitliagkas. Negative momentum for improved game dynamics. In *International Conference on Artificial Intelligence and Statistics*, pages 1802–1811, 2019.

- Mikko Heino, Johan AJ Metz, and Veijo Kaitala. The enigma of frequency-dependent selection. *Trends in Ecology & Evolution*, 13(9):367–370, 1998.
- Josef Hofbauer. Evolutionary dynamics for bimatrix games: A hamiltonian system? *Journal of Mathematical Biology*, 34(5):675, 1996.
- Josef Hofbauer, Karl Sigmund, et al. *Evolutionary games and population dynamics*. Cambridge university press, 1998.
- Josef Hofbauer, Sylvain Sorin, and Yannick Viossat. Time average replicator and best-reply dynamics. *Mathematics of Operations Research*, 34(2):263–269, 2009.
- Ling Huang, Anthony D Joseph, Blaine Nelson, Benjamin IP Rubinstein, and J Doug Tygar. Adversarial machine learning. In *ACM workshop on Security and artificial intelligence*, pages 43–58, 2011.
- Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018.
- Rolf Kümmerli and Sam Brown. Molecular and regulatory properties of a public good shape the evolution of cooperation. *National Academy of Sciences*, 107:18921–6, 10 2010.
- Steven J Lade, Alessandro Tavoni, Simon A Levin, and Maja Schlüter. Regime shifts in a social-ecological system. *Theoretical Ecology*, 6(3):359–372, 2013.
- Thodoris Lykouris, Vasilis Syrgkanis, and Éva Tardos. Learning and efficiency in games with dynamic population. In *Symposium of Discrete Algorithms*, pages 120–129, 2016.
- Tung Mai, Milena Mihail, Ioannis Panageas, Will Ratcliff, Vijay Vazirani, and Peter Yunker. Cycles in Zero-Sum Differential Games and Biological Diversity. In *ACM Conference on Economics and Computation*, page 339–350, 2018.
- Panayotis Mertikopoulos, Christos Papadimitriou, and Georgios Piliouras. Cycles in adversarial regularized learning. In *Symposium of Discrete Algorithms*, pages 2703–2717, 2018.
- Risto Miikkulainen, Jason Liang, Elliot Meyerson, Aditya Rawal, Daniel Fink, Olivier Francon, Bala Raju, Hormoz Shahrzad, Arshak Navruzyan, Nigel Duffy, et al. Evolving deep neural networks. In *Artificial Intelligence in the Age of Neural Networks and Brain Computing*, pages 293–312. Elsevier, 2019.
- Paul Milgrom and Ilya Segal. Envelope theorems for arbitrary choice sets. *Econometrica*, 70(2):583–601, 2002.
- Charles Mullon, Laurent Keller, and Laurent Lehmann. Social polymorphism is favoured by the co-evolution of dispersal with social behaviour. *Nature ecology & evolution*, 2(1):132–140, 2018.
- 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*, pages 7186–7196, 2020.
- John Nash. Non-cooperative games. *Annals of Mathematics*, pages 286–295, 1951.
- N Nisan, T Roughgarden, E Tardos, and VV Vazirani. *Algorithmic Game Theory*. Cambridge university press, 2007.
- Julien Perolat, Remi Munos, Jean-Baptiste Lespiau, Shayegan Omidshafiei, Mark Rowland, Pedro Ortega, Neil Burch, Thomas Anthony, David Balduzzi, Bart De Vylder, Georgios Piliouras, Marc Lanctot, and Karl Tuyls. From Poincaré Recurrence to Convergence in Imperfect Information Games: Finding Equilibrium via Regularization. *arXiv preprint arXiv:2002.08456*, 2020.
- Georgios Piliouras and Jeff S Shamma. Optimization despite chaos: Convex relaxations to complex limit sets via Poincaré recurrence. In *Symposium of Discrete Algorithms*, pages 861–873, 2014.

- Georgios Piliouras, Carlos Nieto-Granda, Henrik I Christensen, and Jeff S Shamma. Persistent patterns: Multi-agent learning beyond equilibrium and utility. In *International Conference on Autonomous Agents and Multi-Agent Systems*, pages 181–188, 2014.
- Henri Poincaré. Sur le problème des trois corps et les équations de la dynamique. *Acta mathematica*, 13(1), 1890.
- Adin Ross-Gillespie, Zoé Dumas, and Rolf Kümmerli. Evolutionary dynamics of interlinked public goods traits: An experimental study of siderophore production in *pseudomonas aeruginosa*. *Journal of Evolutionary Biology*, 28, 11 2014.
- William H Sandholm. *Population games and evolutionary dynamics*. MIT press, 2010.
- Shai Shalev-Shwartz et al. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2012.
- Yoav Shoham and Kevin Leyton-Brown. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press, 2008.
- Kenneth O Stanley and Risto Miikkulainen. Evolving neural networks through augmenting topologies. *Evolutionary Computation*, 10(2):99–127, 2002.
- Alexander J Stewart and Joshua B Plotkin. Collapse of cooperation in evolving games. *Proceedings of the National Academy of Sciences*, 111(49):17558–17563, 2014.
- Andrew R Tilman, James R Watson, and Simon Levin. Maintaining cooperation in social-ecological systems. *Journal of Theoretical Biology*, 10(2):155–165, 2017.
- Andrew R Tilman, Joshua B Plotkin, and Erol Akçay. Evolutionary games with environmental feedbacks. *Nature communications*, 11(1):1–11, 2020.
- Jamal Toutouh, Erik Hemberg, and Una-May O'Reilly. Spatial evolutionary generative adversarial networks. In *Genetic and Evolutionary Computation Conference*, pages 472–480, 2019.
- 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*, pages 10450–10461, 2019.
- Chaoyue Wang, Chang Xu, Xin Yao, and Dacheng Tao. Evolutionary generative adversarial networks. *IEEE Transactions on Evolutionary Computation*, 23(6):921–934, 2019.
- 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. *National Academy of Sciences*, 113(47), 2016.
- Stuart West and A. Buckling. Cooperation, virulence and siderophore production in bacterial parasites. *Proc. R. Soc. B*, 270:37–44, 01 2002.
- Stuart West, Ashleigh Griffin, Andy Gardner, and Steve Diggle. Social evolution theory for microbes. *Nature reviews. Microbiology*, 4:597–607, 09 2006.
- Stuart A West, Stephen P Diggle, Angus Buckling, Andy Gardner, and Ashleigh S Griffin. The social lives of microbes. *Annual Review of Ecology, Evolution, and Systematics*, 38:53–77, 2007.
- Lee Worden and Simon A Levin. Evolutionary escape from the prisoner's dilemma. *Journal of Theoretical Biology*, 245(3):411–422, 2007.
- Yan Wu, Jeff Donahue, David Balduzzi, Karen Simonyan, and Timothy Lillicrap. LOGAN: Latent Optimisation for Generative Adversarial Networks. *arXiv preprint arXiv:1912.00953*, 2019.

Appendices

We provide a detailed overview of related work in Appendix A, omitted proofs in Appendix B, and further experimental results and details in Appendix C.

A Related Work

We now cover a broader class of related work. The related work can be categorized into the following topics: (i) learning in zero-sum games, Poincaré recurrence, and cycles, (ii) learning in time-evolving games, and (iii) experimental works.

Learning in Zero-Sum Games, Poincaré Recurrence, and Cycles. Classical works in evolutionary game theory have long explored the interface between dynamical systems theory and learning in games with the goal of understanding when cycling or other non-convergent behaviors emerge (Hofbauer et al., 1998; Sandholm, 2010). For specific classes of games such as zero-sum or partnership bimatrix games, constants of motion are known to exist (Hofbauer, 1996), and volume preservation properties of the replicator dynamics have been shown (Eshel et al., 1983; Hofbauer et al., 1998). More recently, applications of dynamical systems tools to the analysis of learning algorithms has lead to new insights about non-convergent behavior and its interpretation (Vlatakis-Gkaragkounis et al., 2019; Boone and Piliouras, 2019; Piliouras and Shamma, 2014; Piliouras et al., 2014; Mertikopoulos et al., 2018; Perolat et al., 2020). For instance, several works demonstrate the surprising property that replicator dynamics are Poincaré recurrent in pairwise zero-sum polymatrix games without self-loops by showing both the existence of a constant of motion and volume preservation (Piliouras and Shamma, 2014; Piliouras et al., 2014). Boone and Piliouras (2019) extend this analysis to pairwise zero-sum polymatrix games with self loops. Mai et al. (2018) consider a biologically-inspired time-evolving game in which the payoff of a collection of species playing against itself depends on a dynamically changing environmental variable. The dynamics are shown to be Poincaré recurrent for certain parameter regimes, which is interpreted as promoting diversity.

Learning in Time-Evolving Games. Following a similar theme, there has been renewed interest in learning in games in which the payoffs change in time or are affected by a feedback mechanism from the environment. Such reciprocal feedback between strategies and environment variables arises in a number of applications including biology (Akçay and Roughgarden, 2011; Cortez and Ellner, 2010; Tilman et al., 2020), ecology (Worden and Levin, 2007; Lade et al., 2013), sociology (Bowles et al., 2003; Tilman et al., 2017), economics (Friedman, 1991), and more recently machine learning and artificial intelligence (Cardoso et al., 2019; Duvocelle et al., 2018; Lykouris et al., 2016). For instance, Cardoso et al. (2019) design algorithms with *small regret*—tantamount to the long-term payoff of both players being close to minimax optimal in hindsight—for a class of repeated play zero-sum games, termed *online matrix games*, such that players’ payoff matrices may change in each round. In related work, in the class of continuous games, Duvocelle et al. (2018) analyze the long-run behavior of regret minimizing players in time-evolving games which are executed in a sequence of concave, monotone *stage games*. In other work (Lykouris et al., 2016; Cavaliere et al., 2012), dynamically changing environments are modeled via a dynamic player population in which players leave the game with some probability and new players enter.

Closer to the class of time-evolving games we consider, another body of work captures various natural dynamical processes via action-dependent games (West et al., 2006, 2007; West and Buckling, 2002; Kümmel and Brown, 2010; Ross-Gillespie et al., 2014; Bauch et al., 2003; Bauch and Earn, 2004; Galvani et al., 2007; Weitz et al., 2016; Mai et al., 2018; Akçay and Roughgarden, 2011; Stewart and Plotkin, 2014; Mullon

et al., 2018). In such settings, the actions of the players (or in terms of evolutionary game-theory, the frequencies of species within a population), may affect the environment and thus change the game’s payoffs. For instance, the dynamics of the vaccinated human population are efficiently captured by such action dependent-games. Parents decide to vaccinate their newborns by weighing the risk of a potential disease to the risk of morbidity of vaccination. However, as the unvaccinated population increases so does the cost of the *do not vaccinate action* (Kümmerli and Brown, 2010; Ross-Gillespie et al., 2014; Bauch et al., 2003; Bauch and Earn, 2004; Galvani et al., 2007). Similar instances appear in the dynamics of bacteria and microbe populations since it is common for certain types of bacteria to cause certain environmental changes (e.g., increase of nutrient-scavenging enzymes or fixation of inorganic nutrients) that affect differently the various individuals of the population (West et al., 2006, 2007; West and Buckling, 2002).

Experimental Works. Motivated by the observation of cycling behavior in applications of game-theoretic learning dynamics to machine learning dynamics, there has been a push to better understand recurrence and to potentially see it as a solution concept. Indeed, standard gradient dynamics are known to result in cycling or recurrent behavior in continuous time (Piliouras and Shamma, 2014; Mertikopoulos et al., 2018) and chaotic and divergent behavior in discrete time (Bailey and Piliouras, 2018; Cheung and Piliouras, 2019; Gidel et al., 2019). Daskalakis et al. (2018) and Mertikopoulos et al. (2018) explore adaptations to follow-the-regularized learning (FTRL) dynamics that enable convergence in bilinear and more general nonconvex-nonconcave problems, respectively. Each work shows that versions of optimistic mirror descent can successfully train generative adversarial networks on challenging datasets. Similarly, Balduzzi et al. (2018) design dynamics that adjust for components of the gradient dynamics that cause cycling by drawing connections to Hamiltonian dynamics.

As opposed to trying to mitigate cycling behavior and converge to fixed points via carefully designed learning dynamics, a separate line of work instead makes use of the fact that in convex-concave games the time-average of standard gradient dynamics converge to the interior equilibrium (Freund and Schapire, 1999). In particular, Yazıcı et al. (2019) show that training generative adversarial networks and then averaging the parameters of the networks uniformly or by an exponential moving average is an empirically effective method. Gidel et al. (2019) explore a similar perspective of uniform averaging for simultaneous and alternating gradient updates using negative momentum. Moreover, Vlatakis-Gkaragkounis et al. (2019) show for a class of nonconvex-nonconcave minimax games, which generalize bilinear zero-sum games, that the time-average of gradient dynamics converges to an equilibrium for certain problem instances and initial conditions. This provides further evidence for the efficacy of recurrence as a solution concept that is relevant to machine learning applications such as generative adversarial networks. A final line of work explores evolutionary algorithms as a training method for generative adversarial networks (Costa et al., 2020; Karras et al., 2018; Wang et al., 2019; Costa et al., 2019a,b; Garciarena et al., 2018; Al-Dujaili et al., 2018; Toutouh et al., 2019).

B Proofs

We organize the proofs in the order that the results appeared in the paper. Proofs for results from Sections 3, 4, and 5 of the paper can be found in Appendix B.1, B.2, B.3, respectively.

In Appendix B.1, we begin by proving Proposition 3.1, which shows a reduction from the time-evolving generalized rock-paper-scissors game presented by Mai et al. (2018) to a rescaled zero-sum polymatrix game. Following that proof, we prove Theorem 3.1, which shows the reduction of Proposition 3.1 extends to a general class of time-evolving dynamical systems. This result demonstrates the breadth of time-evolving games that can in fact be studied as rescaled zero-sum polymatrix games.

In Appendix B.2, we prove Lemma 4.1 and Lemma 4.2. Recall that Lemma 4.1 and Lemma 4.2 show that the replicator dynamics are volume preserving and have bounded orbits in rescaled zero-sum polymatrix games, respectively. Moreover, as shown in Section 4 via the proof of Theorem 4.1, Lemma 4.1 and Lemma 4.2 nearly immediately imply Theorem 4.1, which guarantees the replicator dynamics are Poincaré recurrent in rescaled zero-sum polymatrix games with interior Nash equilibria.

Appendix B.3 contains the proofs of Theorem 5.1 and Proposition 5.1, which show time-average equilibria and utility convergence of replicator dynamics in rescaled zero-sum polymatrix games along with the bounded regret property in general polymatrix games, respectively. The proof of Theorem 5.2, which provides a linear program to compute Nash equilibrium in rescaled zero-sum polymatrix games can also be found in Appendix B.3.

B.1 Proofs of Reductions from Time-Varying Games to Polymatrix Games from Section 3

In Appendix B.1.1, we provide the proof of Proposition 3.1 from Section 3. This result shows a reduction from a time-evolving generalized rock-paper-scissors game to an appropriate rescaled zero-sum polymatrix game. Moreover, in Appendix B.1.2, we provide the proof of Theorem 3.1 from Section 3, which generalizes the reduction of Proposition 3.1 to a broad class of dynamical systems that represent multiple evolving populations interacting with multiple evolving environments.

B.1.1 Proof of Proposition 3.1

Let y and w denote the mixed strategies of player 1 and player 2, respectively, which correspond to the population and the environment. In what follows, we show that both the population and environment dynamics can be simplified so that it is clear each player is following replicator dynamics in a static rescaled zero-sum polymatrix game.

Environment Dynamics. We begin by considering the environment dynamics. The dynamics of player 2 (w -player) for each action $i \in \{1, \dots, n\}$ with an initial condition on the interior of the simplex are given by

$$\dot{w}_i = w_i \sum_{j=1}^n w_j (y_j - y_i). \quad (10)$$

Now observe that

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

Since $\sum_{i=1}^n \dot{w}_i = 0$ as shown above and the given initial condition is such that $w(0) \in \Delta^{n-1}$, we conclude $w(t) \in \Delta^{n-1}$ and $\sum_{j=1}^n w_j(t) = 1$ for any $t \geq 0$. From a series of algebraic manipulations and the fact that $\sum_{j=1}^n w_j = 1$, we obtain an equivalent form of the dynamics given in (10) for each action $i \in \{1, \dots, n\}$ as follows:

$$\begin{aligned} \dot{w}_i &= w_i \sum_{j=1}^n w_j (y_j - y_i) \\ &= w_i \sum_{j=1}^n w_j y_j - w_i \sum_{j=1}^n w_j y_i \\ &= w_i \sum_{j=1}^n w_j y_j - w_i y_i \\ &= w_i \left(-y_i + \sum_{j=1}^n w_j y_j \right). \end{aligned} \quad (11)$$

The dynamics for player 2 (w -player) from (11) in vector form are then given by

$$\dot{w} = w \cdot (-Iy + w^\top Iy). \quad (12)$$

We now see that the dynamics in (12) are replicator dynamics in which player 2 (w -player) plays against player 1 (y -player) with the payoff matrix $A^{w,y} = -I$, where the superscript indices (w, y) indicate the players.

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

$$\dot{y}_i = y_i ((P(w)y)_i - y^\top P(w)y). \quad (13)$$

From an expansion of the payoff matrix $P(w)$ in (13), the dynamics of player 1 (y -player) for each action $\{1, \dots, n\}$ are equivalently

$$\dot{y}_i = y_i ((Py)_i - y^\top Py) + y_i \left(\mu \sum_{j=1}^n (w_i - w_j) y_j - \mu \sum_{\ell=1}^n y_\ell \sum_{j=1}^n (w_\ell - w_j) y_j \right). \quad (14)$$

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 (14) simplify to the form

$$\dot{y}_i = y_i ((Py)_i - y^\top Py) + y_i \left(\mu \sum_{j=1}^n (w_i - w_j) y_j \right). \quad (15)$$

Finally, $y(0) \in \Delta^{n-1}$ so that $\sum_{j=1}^n y_j(t) = 1$ for any $t \geq 0$ since clearly \dot{y} is replicator dynamics with the payoff matrix $P(w)$ in (13). Accordingly, for each action $i \in \{1, \dots, n\}$, we simplify the dynamics in (15) as follows:

$$\begin{aligned} \dot{y}_i &= y_i ((Py)_i - y^\top Py) + y_i \left(\mu \sum_j (w_i - w_j) y_j \right) \\ &= y_i ((Py)_i - y^\top Py) + y_i \left(\mu w_i \sum_{j=1}^n y_j - \sum_{j=1}^n w_j y_j \right) \\ &= y_i ((Py)_i - y^\top Py) + y_i \left(\mu w_i - \sum_{j=1}^n \mu w_j y_j \right). \end{aligned} \quad (16)$$

The dynamics for player 1 (y -player) from (16) in vector form are then given by

$$\dot{y} = y \cdot (Py + y^\top Py \cdot \mathbf{1}) + y \cdot (\mu Iw + \mu y^\top Iw \cdot \mathbf{1}). \quad (17)$$

We now see that the dynamics in (17) 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 again the superscript indices indicate the players in the payoff matrix.



Figure 5: Basic interaction structure in the time-evolving systems that reduce to rescaled zero-sum polymatrix games. The *red* nodes denotes a population of species, while the *blue* node is an environment.

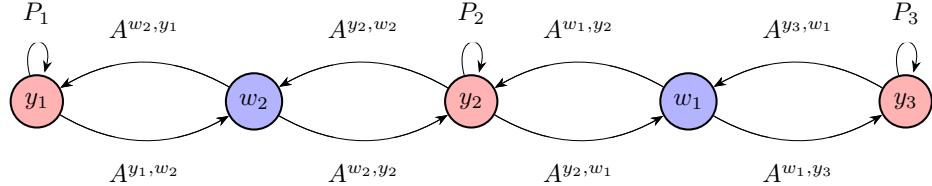


Figure 6: Example polymatrix game that can be formed from a reduction of a time-evolving dynamical system. The *red* nodes denote a population of species, while the *blue* nodes stand for an environment.

Static Polymatrix Game. We have shown that the dynamics of (13) and (10) correspond to replicator dynamics for a two-player polymatrix game in which player 1 (y -player) has utility $u_y(y, w) = y^\top P y + \mu y^\top I w$ 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 utility $\eta_y u_y(y, w) + \eta_w u_w(y, w) = 0$ for every strategy (y, w) . This allows us to conclude by definition that the time-evolving generalized rock-paper-scissors game is equivalent to replicator dynamics in a two-player rescaled zero-sum polymatrix game.

B.1.2 Proof of Theorem 3.1

In this section, we provide the proof of Theorem 3.1. The theorem shows that the reduction from Proposition 3.1 extends to more general dynamical systems. Before giving the proof, we provide some intuition for the underlying structure.

Time-Evolving Games that Admit Reduction to Polymatrix Games. The class of time-evolving systems that admit a reduction are such that the basic interaction structure is of the form in Figure 5. The key component of any general structure formed from the building block is that each environment w_k is only connected to populations, and each population y_ℓ is only connected to environments or themselves via a self-loop. As an example of the type of generalization that is possible for the reduction, consider the polymatrix game in Figure 6. Of course the game graph does not have to be a line, but population nodes should be separated by environment nodes.

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}_ℓ^y be the set of environments which coevolve with the population y_ℓ via the building block structure from Figure 5. The time-evolving dynamics for each population ℓ are given componentwise by

$$\dot{y}_{\ell,i} = y_{\ell,i} ((P_\ell(w)y_\ell)_i - y_\ell^\top P_\ell(w)y_\ell), \quad (18)$$

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

$$W^{\ell,k} = \begin{pmatrix} 0 & (A^{\ell,k}w_k)_1 - (A^{\ell,k}w_k)_2 & \cdots & (A^{\ell,k}w_k)_1 - (A^{\ell,k}w_k)_n \\ (A^{\ell,k}w_k)_2 - (A^{\ell,k}w_k)_1 & 0 & \cdots & (A^{\ell,k}w_k)_2 - (Aw_k)_n \\ \vdots & \vdots & \vdots & \vdots \\ (A^{\ell,k}w_k)_n - (A^{\ell,k}w_k)_1 & (A^{\ell,k}w_k)_n - (A^{\ell,k}w_k)_2 & \cdots & 0 \end{pmatrix}.$$

Further, the time-evolving dynamics for each environment k are given componentwise by

$$\dot{w}_{k,i} = w_{k,i} \sum_{\ell \in \mathcal{N}_k^w} \sum_{j=1}^n w_{k,j} ((A^{k,\ell}y_\ell)_i - (A^{k,\ell}y_\ell)_j). \quad (19)$$

We now prove that any time-evolving game defined by the dynamics in (18–19) 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 3.1, 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 (19) is

$$\begin{aligned} \dot{w}_{k,i} &= w_{k,i} \sum_{\ell \in \mathcal{N}_k^w} \sum_{j=1}^n w_{k,j} ((A^{k,\ell}y_\ell)_i - (A^{k,\ell}y_\ell)_j) \\ &= w_{k,i} \sum_{\ell \in \mathcal{N}_k^w} \left(\sum_{j=1}^n w_{k,j} (A^{k,\ell}y_\ell)_i - \sum_{j=1}^n w_{k,j} (A^{k,\ell}y_\ell)_j \right) \\ &= w_{k,i} \sum_{\ell \in \mathcal{N}_k^w} \left((A^{k,\ell}y_\ell)_i - \sum_{j=1}^n w_{k,j} (A^{k,\ell}y_\ell)_j \right) \\ &= w_{k,i} \sum_{\ell \in \mathcal{N}_k^w} ((A^{k,\ell}y_\ell)_i - w_k^\top A^{k,\ell}y_\ell). \end{aligned}$$

It is now clear that the dynamics from (19) 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 (18) are equivalent to

$$\begin{aligned} \dot{y}_{\ell,i} &= y_{\ell,i} ((P_\ell y_\ell)_i - y_\ell^\top P_\ell y_\ell) \\ &\quad + y_{\ell,i} \left(\sum_{k \in \mathcal{N}_\ell^y} \sum_{j=1}^n ((A^{\ell,k}w_k)_i - (A^{\ell,k}w_k)_j) y_{\ell,j} - \sum_{k \in \mathcal{N}_\ell^y} \sum_{s=1}^n y_{\ell,s} \sum_{j=1}^n ((A^{\ell,k}w_k)_s - (A^{\ell,k}w_k)_j) y_{\ell,j} \right). \end{aligned}$$

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

$$\begin{aligned}
\dot{y}_{\ell,i} &= y_{\ell,i} ((P_\ell y_\ell)_i - y_\ell^\top P_\ell y_\ell) + y_{\ell,i} \sum_{k \in \mathcal{N}_\ell^y} \sum_{j=1}^n ((A^{\ell,k} w_k)_i - (A^{\ell,k} w_k)_j) y_{\ell,j} \\
&= y_{\ell,i} ((P_\ell y_\ell)_i - y_\ell^\top P_\ell y_\ell) + y_{\ell,i} \sum_{k \in \mathcal{N}_\ell^y} \left(\sum_{j=1}^n (A^{\ell,k} w_k)_i y_{\ell,j} - \sum_{j=1}^n (A^{\ell,k} w_k)_j y_{\ell,j} \right) \\
&= y_{\ell,i} ((P_\ell y_\ell)_i - y_\ell^\top P_\ell y_\ell) + y_{\ell,i} \sum_{k \in \mathcal{N}_\ell^y} \left((A^{\ell,k} w_k)_i - \sum_{j=1}^n (A^{\ell,k} w_k)_j y_{\ell,j} \right) \\
&= y_{\ell,i} ((P_\ell y_\ell)_i - y_\ell^\top P_\ell y_\ell) + y_{\ell,i} \sum_{k \in \mathcal{N}_\ell^y} ((A^{\ell,k} w_k)_i - y_\ell^\top A^{\ell,k} w_k).
\end{aligned}$$

The final equation shows that the dynamics from (18) 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 (18–19) 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, Cai et al. (2016) 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 B.1 (Theorem 8 (Cai et al., 2016)). *Let $G = (V, E)$ be a polymatrix game. For any player $i \in V$, pure strategy $\alpha \in \mathcal{A}_i$, and joint strategy x_{-i} of the rest of the players, denote by $W(\alpha, x_{-i}) = \sum_{j \in V} u_j(\alpha, x_{-i})$ 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*

$$\max_{x_{-i}} W(\beta, x_{-i}) - W(\alpha, x_{-i})$$

equals zero for all $i \in V$ and $\alpha, \beta \in \mathcal{A}_i$. Moreover, this condition can be checked in polynomial time in the number of players and strategies.

B.2 Proof of Poincaré Recurrence in Rescaled Zero-Sum Polymatrix Games from Section 4

We now provide the proofs of Lemma 4.1 and Lemma 4.2 from Section 4 in Appendix B.2.1 and Appendix B.2.2, respectively. Recall that Lemma 4.1 and Lemma 4.2 show that the replicator dynamics are volume preserving and have bounded orbits in rescaled zero-sum polymatrix games, respectively. Moreover, as shown in Section 4 via the proof of Theorem 4.1, Lemma 4.1 and Lemma 4.2 nearly immediately imply Theorem 4.1, which guarantees the replicator dynamics are Poincaré recurrent in rescaled zero-sum polymatrix games with interior Nash equilibria.

B.2.1 Proof of Lemma 4.1

We need to show

$$\sum_{i \in V} \sum_{\alpha \in \mathcal{A}_i} \frac{d}{dz_{i\alpha}} F_{i\alpha}(z) = 0.$$

Recall from (9) that for each $\alpha \in \mathcal{A}_i$ and $i \in V$,

$$F_{i\alpha}(z) = \sum_{j \in V} \sum_{\beta \in \mathcal{A}_j} A_{\alpha\beta}^{ij} \frac{e^{z_{j\beta}}}{\sum_{\ell \in \mathcal{A}_j} e^{z_{j\ell}}} - \sum_{j \in V} \sum_{\beta \in \mathcal{A}_j} A_{1\beta}^{ij} \frac{e^{z_{j\beta}}}{\sum_{\ell \in \mathcal{A}_j} e^{z_{j\ell}}}.$$

It follows that for any agent $i \in V$,

$$\sum_{\alpha \in \mathcal{A}_i} \frac{d}{dz_{i\alpha}} F_{i\alpha}(z) = \sum_{\alpha \in \mathcal{A}_i} \sum_{j \in V} \sum_{\beta \in \mathcal{A}_j} A_{\alpha\beta}^{ij} \frac{d}{dz_{i\alpha}} \frac{e^{z_{j\beta}}}{\sum_{\ell \in \mathcal{A}_j} e^{z_{j\ell}}} - \sum_{\alpha \in \mathcal{A}_i} \sum_{j \in V} \sum_{\beta \in \mathcal{A}_j} A_{1\beta}^{ij} \frac{d}{dz_{i\alpha}} \frac{e^{z_{j\beta}}}{\sum_{\ell \in \mathcal{A}_j} e^{z_{j\ell}}}.$$

Moreover, observe that for $i \neq j$,

$$\frac{d}{dz_{i\alpha}} \frac{e^{z_{j\beta}}}{\sum_{\ell \in \mathcal{A}_j} e^{z_{j\ell}}} = 0.$$

Consequently, we get that

$$\sum_{\alpha \in \mathcal{A}_i} \frac{d}{dz_{i\alpha}} F_{i\alpha}(z) = \sum_{\alpha \in \mathcal{A}_i} \sum_{\beta \in \mathcal{A}_i} A_{\alpha\beta}^{ii} \frac{d}{dz_{i\alpha}} \frac{e^{z_{i\beta}}}{\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}}} - \sum_{\alpha \in \mathcal{A}_i} \sum_{\beta \in \mathcal{A}_i} A_{1\beta}^{ii} \frac{d}{dz_{i\alpha}} \frac{e^{z_{i\beta}}}{\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}}}.$$

We now separate each sum over $\beta \in \mathcal{A}_i$ into a pair of sums over $\beta \neq \alpha$ and $\beta = \alpha$ for $\alpha \in \mathcal{A}_i$ and any $i \in V$ to get that

$$\begin{aligned} \sum_{\alpha \in \mathcal{A}_i} \frac{d}{dz_{i\alpha}} F_{i\alpha}(z) &= \sum_{\alpha \in \mathcal{A}_i} \sum_{\beta \neq \alpha} A_{\alpha\beta}^{ii} \frac{d}{dz_{i\alpha}} \frac{e^{z_{i\beta}}}{\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}}} - \sum_{\alpha \in \mathcal{A}_i} A_{\alpha\alpha}^{ii} \frac{d}{dz_{i\alpha}} \frac{e^{z_{i\alpha}}}{\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}}} \\ &\quad - \sum_{\alpha \in \mathcal{A}_i} \sum_{\beta \neq \alpha} A_{1\beta}^{ii} \frac{d}{dz_{i\alpha}} \frac{e^{z_{i\beta}}}{\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}}} - \sum_{\alpha \in \mathcal{A}_i} A_{1\alpha}^{ii} \frac{d}{dz_{i\alpha}} \frac{e^{z_{i\alpha}}}{\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}}}. \end{aligned} \quad (20)$$

Recall that the self-loops are antisymmetric, which means that $A_{\alpha\alpha}^{ii} = 0$ for any $\alpha \in \mathcal{A}_i$ and $i \in V$. From this property of the game class,

$$\sum_{\alpha \in \mathcal{A}_i} A_{\alpha\alpha}^{ii} \frac{d}{dz_{i\alpha}} \frac{e^{z_{i\alpha}}}{\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}}} = 0.$$

Accordingly, an equivalent form of (20) is the expression

$$\sum_{\alpha \in \mathcal{A}_i} \frac{d}{dz_{i\alpha}} F_{i\alpha}(z) = \sum_{\alpha \in \mathcal{A}_i} \sum_{\beta \neq \alpha} A_{\alpha\beta}^{ii} \frac{d}{dz_{i\alpha}} \frac{e^{z_{i\beta}}}{\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}}} - \sum_{\alpha \in \mathcal{A}_i} \sum_{\beta \neq \alpha} A_{1\beta}^{ii} \frac{d}{dz_{i\alpha}} \frac{e^{z_{i\beta}}}{\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}}} - \sum_{\alpha \in \mathcal{A}_i} A_{1\alpha}^{ii} \frac{d}{dz_{i\alpha}} \frac{e^{z_{i\alpha}}}{\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}}}. \quad (21)$$

The derivatives in (21) are given by

$$\frac{d}{dz_{i\alpha}} \frac{e^{z_{i\beta}}}{\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}}} = \begin{cases} \frac{\sum_{\ell \in \mathcal{A}_i} e^{z_{i\alpha} + z_{i\ell} - z_{i\beta}}}{(\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}})^2}, & \alpha = \beta \\ -e^{z_{i\beta} + z_{i\alpha}} / (\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}})^2, & \alpha \neq \beta. \end{cases}$$

Evaluating the derivatives in (21), we get that

$$\begin{aligned} \sum_{\alpha \in \mathcal{A}_i} \frac{d}{dz_{i\alpha}} F_{i\alpha}(z) &= - \sum_{\alpha \in \mathcal{A}_i} \sum_{\beta \neq \alpha} A_{\alpha\beta}^{ii} \frac{e^{z_{i\beta} + z_{i\alpha}}}{(\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}})^2} \\ &\quad + \sum_{\alpha \in \mathcal{A}_i} \sum_{\beta \neq \alpha} A_{1\beta}^{ii} \frac{e^{z_{i\beta} + z_{i\alpha}}}{(\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}})^2} - \sum_{\alpha \in \mathcal{A}_i} A_{1\alpha}^{ii} \frac{\sum_{\beta \in \mathcal{A}_i} e^{z_{i\beta} + z_{i\alpha}} - e^{z_{i\alpha} + z_{i\alpha}}}{(\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}})^2}. \end{aligned} \quad (22)$$

Moreover, from a series of algebraic manipulations, we find that

$$\begin{aligned}
& \sum_{\alpha \in \mathcal{A}_i} \sum_{\beta \neq \alpha} A_{1\beta}^{ii} \frac{e^{z_{i\beta} + z_{i\alpha}}}{(\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}})^2} - \sum_{\alpha \in \mathcal{A}_i} A_{1\alpha}^{ii} \frac{\sum_{\beta \in \mathcal{A}_i} e^{z_{i\beta} + z_{i\alpha}} - e^{z_{i\alpha} + z_{i\alpha}}}{(\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}})^2} \\
&= \sum_{\alpha \in \mathcal{A}_i} \sum_{\beta \neq \alpha} A_{1\beta}^{ii} \frac{e^{z_{i\beta} + z_{i\alpha}}}{(\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}})^2} + \sum_{\alpha \in \mathcal{A}_i} A_{1\alpha}^{ii} \frac{e^{z_{i\alpha} + z_{i\alpha}}}{(\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}})^2} - \sum_{\alpha \in \mathcal{A}_i} \sum_{\beta \in \mathcal{A}_i} A_{1\alpha}^{ii} \frac{e^{z_{i\beta} + z_{i\alpha}}}{(\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}})^2} \\
&= \sum_{\alpha \in \mathcal{A}_i} \sum_{\beta \in \mathcal{A}_i} A_{1\beta}^{ii} \frac{e^{z_{i\beta} + z_{i\alpha}}}{(\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}})^2} - \sum_{\alpha \in \mathcal{A}_i} \sum_{\beta \in \mathcal{A}_i} A_{1\alpha}^{ii} \frac{e^{z_{i\beta} + z_{i\alpha}}}{(\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}})^2} \\
&= 0.
\end{aligned}$$

It follows that (22) is equivalent to

$$\sum_{\alpha \in \mathcal{A}_i} \frac{d}{dz_{i\alpha}} F_{i\alpha}(z) = - \sum_{\alpha \in \mathcal{A}_i} \sum_{\beta \neq \alpha} A_{\alpha\beta}^{ii} \frac{e^{z_{i\beta} + z_{i\alpha}}}{(\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}})^2}. \quad (23)$$

Finally, we reorganize the sums in (23) 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

$$\sum_{\alpha \in \mathcal{A}_i} \frac{d}{dz_{i\alpha}} F_{i\alpha}(z) = - \sum_{\alpha \in \mathcal{A}_i} \sum_{\beta \neq \alpha} A_{\alpha\beta}^{ii} \frac{e^{z_{i\beta} + z_{i\alpha}}}{(\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}})^2} = \sum_{(\alpha, \beta): \beta \neq \alpha} (-A_{\alpha\beta}^{ii} - A_{\beta\alpha}^{ii}) \frac{e^{z_{i\alpha}} + e^{z_{i\beta}}}{(\sum_{\ell \in \mathcal{A}_i} e^{z_{i\ell}})^2} = 0.$$

So, by summing (24) over $i \in V$, we obtain

$$\sum_{i \in V} \sum_{\alpha \in \mathcal{A}_i} \frac{d}{dz_{i\alpha}} F_{i\alpha}(z) = 0.$$

B.2.2 Proof of Lemma 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

$$\Phi(t) = \sum_{i \in V} \sum_{\alpha \in \mathcal{A}_i} \eta_i x_{i\alpha}^* \ln x_{i\alpha} \quad (24)$$

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 (24) 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) given by

$$\dot{x}_{i\alpha} = x_{i\alpha}(u_{i\alpha}(x) - u_i(x)), \quad \forall \alpha \in \mathcal{A}_i. \quad (25)$$

We simplify the time derivative of $\Phi(t)$ using the structure of the dynamics given in (25) as follows:

$$\begin{aligned}
\frac{d\Phi(t)}{dt} &= \sum_{i \in V} \eta_i \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}^* \frac{d \ln x_{i\alpha}}{dt} \\
&= \sum_{i \in V} \eta_i \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}^* \frac{\dot{x}_{i\alpha}}{x_{i\alpha}} \\
&= \sum_{i \in V} \eta_i \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}^* (u_{i\alpha}(x) - u_i(x)) \\
&= \sum_{i \in V} \eta_i \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}^* u_{i\alpha}(x) - \sum_{i \in V} \eta_i \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}^* u_i(x).
\end{aligned} \quad (26)$$

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 $\alpha \in \mathcal{A}_i$, which can be represented by $x_i = e_\alpha$, when the rest of the agents play x_{-i} . Then, from the fact that $A_{\alpha\alpha}^{ii} = 0$ for all $\alpha \in \mathcal{A}_i$ and $i \in V$, we obtain

$$\begin{aligned} \sum_{i \in V} \eta_i \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}^* u_{i\alpha}(x) &= \sum_{i \in V} \eta_i \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}^* \sum_{j:(i,j) \in E} (A^{ij} x_j)_\alpha \\ &= \sum_{i \in V} \eta_i \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}^* (A_{\alpha}^{ii} e_\alpha)_\alpha + \sum_{i \in V} \eta_i \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}^* \sum_{j:(i,j) \in E'} (A^{ij} x_j)_\alpha \\ &= \sum_{i \in V} \eta_i \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}^* A_{\alpha\alpha}^{ii} + \sum_{i \in V} \eta_i \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}^* \sum_{j:(i,j) \in E'} (A^{ij} x_j)_\alpha \\ &= \sum_{i \in V} \eta_i (x_i^*)^\top \sum_{j:(i,j) \in E'} A^{ij} x_j. \end{aligned} \quad (27)$$

Moreover, since $\sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}^* = 1$ 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

$$\sum_{i \in V} \eta_i \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}^* u_i(x) = \sum_{i \in V} \eta_i u_i(x) \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}^* = \sum_{i \in V} \eta_i u_i(x) = 0. \quad (28)$$

Combining (26), (27), and (28), we have

$$\frac{d\Phi(t)}{dt} = \sum_{i \in V} \eta_i (x_i^*)^\top \sum_{j:(i,j) \in E'} A^{ij} x_j. \quad (29)$$

For the interior Nash equilibrium x^* under consideration,

$$\begin{aligned} \sum_{i \in V} \eta_i u_i(x^*) &= \sum_{i \in V} \eta_i (x_i^*)^\top \sum_{j:(i,j) \in E} A^{ij} x_j^* \\ &= \sum_{i \in V} \eta_i (x_i^*)^\top \sum_{j:(i,j) \in E'} A^{ij} x_j^* \\ &= 0. \end{aligned} \quad (30)$$

Note that (30) 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 (31) as a result of the polymatrix game being rescaled zero-sum. We continue by subtracting (30) from (29) since it is equal to zero and obtain

$$\frac{d\Phi(t)}{dt} = \sum_{i \in V} \eta_i (x_i^*)^\top \sum_{j:(i,j) \in E'} A^{ij} x_j - \sum_{i \in V} \eta_i (x_i^*)^\top \sum_{j:(i,j) \in E'} A^{ij} x_j^* = \sum_{i \in V} \eta_i \sum_{j:(i,j) \in E'} (x_i^*)^\top A^{ij} (x_j - x_j^*). \quad (32)$$

We now prove that (32) is equal to zero. To do so, we rely on the results of Cai and Daskalakis (2011, 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 (2011) 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 (Cai and Daskalakis, 2011)):

Property 1. $\eta_i A_{\alpha\beta}^{ij} - \eta_i A_{\alpha\gamma}^{ij} = B_{\alpha\beta}^{ij} - B_{\alpha\gamma}^{ij}$ for any $\alpha \in \mathcal{A}_i$ and $\beta, \gamma \in \mathcal{A}_j$.

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 $\gamma \in \mathcal{A}_j$, we can express the summand of (32) using Property 1 as follows:

$$\begin{aligned} (x_i^*)^\top \eta_i A^{ij} (x_j - x_j^*) &= \sum_{\alpha \in \mathcal{A}_i} \sum_{\beta \in \mathcal{A}_j} x_{i\alpha}^* \eta_i A_{\alpha\beta}^{ij} (x_{j\beta} - x_{j\beta}^*) \\ &= \sum_{\alpha \in \mathcal{A}_i} \sum_{\beta \in \mathcal{A}_j} x_{i\alpha}^* (B_{\alpha\beta}^{ij} - B_{\alpha\gamma}^{ij} + \eta_i A_{\alpha\gamma}^{ij}) (x_{j\beta} - x_{j\beta}^*) \\ &= (x_i^*)^\top B^{ij} (x_j - x_j^*) + \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha} (\eta_i A_{\alpha\gamma}^{ij} - B_{\alpha\gamma}^{ij}) \sum_{\beta \in \mathcal{A}_j} (x_{j\beta} - x_{j\beta}^*). \end{aligned} \quad (33)$$

Moreover, observe that since both x_j and x_j^* are on the simplex, $\sum_{\beta \in \mathcal{A}_j} (x_{j\beta} - x_{j\beta}^*) = 0$, and consequently

$$\sum_{\alpha \in \mathcal{A}_i} x_{i\alpha} (-B_{\alpha\gamma}^{ij} + \eta_i A_{\alpha\gamma}^{ij}) \sum_{\beta \in \mathcal{A}_j} (x_{j\beta} - x_{j\beta}^*) = 0. \quad (34)$$

Then, relating (33) and (34), 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, (32) is equivalently

$$\frac{d\Phi(t)}{dt} = \sum_{i \in V} \sum_{j:(i,j) \in E'} (x_i^*)^\top \eta_i A^{ij} (x_j - x_j^*). \quad (35)$$

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

$$\frac{d\Phi(t)}{dt} = \sum_{j \in V} \sum_{i:(j,i) \in E'} (x_j - x_j^*)^\top (c^{ji} \mathbf{1}_{n_j \times n_i} - B^{ji}) x_i^* \quad (36)$$

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 $\sum_{\alpha \in \mathcal{A}_j} x_{j\alpha} = \sum_{\alpha \in \mathcal{A}_j} x_{j\alpha}^* = \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}^* = 1$. Accordingly, we simplify (36) and get

$$\frac{d\Phi(t)}{dt} = - \sum_{j \in V} \sum_{i:(j,i) \in E'} (x_j - x_j^*)^\top B^{ji} x_i^*. \quad (37)$$

Following a similar argument as above, we analyze the summand in (37) for some $j \in V$. Using Property 1

and fixing any strategy $\gamma_i \in \mathcal{A}_i$ for each $i \in V \setminus \{j\}$, we have that

$$\begin{aligned} \sum_{i:(j,i) \in E'} (x_j - x_j^*)^\top B^{ji} x_i^* &= \sum_{i:(j,i) \in E'} \sum_{\alpha \in \mathcal{A}_j} \sum_{\beta \in \mathcal{A}_i} z_{j\alpha} B_{\alpha\beta}^{ji} x_{i\beta}^* \\ &= \sum_{i:(j,i) \in E'} \sum_{\alpha \in \mathcal{A}_j} \sum_{\beta \in \mathcal{A}_i} z_{j\alpha} (\eta_j A_{\alpha\beta}^{ji} - \eta_j A_{\alpha\gamma_i}^{ji} + B_{\alpha\gamma_i}^{ji}) x_{i\beta}^* \\ &= \sum_{i:(j,i) \in E'} \eta_j (x_j - x_j^*)^\top A^{ji} x_i^* + \sum_{i:(j,i) \in E'} \sum_{\alpha \in \mathcal{A}_j} \sum_{\beta \in \mathcal{A}_i} z_{j\alpha} (-\eta_j A_{\alpha\gamma_i}^{ji} + B_{\alpha\gamma_i}^{ji}) x_{i\beta}^*. \end{aligned} \quad (38)$$

where $z_{j\alpha} := x_{j\alpha} - x_{j\alpha}^*$. We now examine the last term in the equation overhead, and use the fact $\sum_{\beta \in \mathcal{A}_i} x_{i\beta}^* = 1$ since $x_i^* \in \mathcal{X}_i^* = \Delta^{n_i-1}$ to get that

$$\begin{aligned} \sum_{i:(j,i) \in E'} \sum_{\alpha \in \mathcal{A}_j} \sum_{\beta \in \mathcal{A}_i} (x_{j\alpha} - x_{j\alpha}^*) (-\eta_j A_{\alpha\gamma_i}^{ji} + B_{\alpha\gamma_i}^{ji}) x_{i\beta}^* &= \sum_{i:(j,i) \in E'} \sum_{\alpha \in \mathcal{A}_j} (x_{j\alpha} - x_{j\alpha}^*) (-\eta_j A_{\alpha\gamma_i}^{ji} + B_{\alpha\gamma_i}^{ji}) \sum_{\beta \in \mathcal{A}_i} x_{i\beta}^* \\ &= \sum_{\alpha \in \mathcal{A}_j} (x_{j\alpha} - x_{j\alpha}^*) \sum_{i:(j,i) \in E'} (-\eta_j A_{\alpha\gamma_i}^{ji} + B_{\alpha\gamma_i}^{ji}) \end{aligned} \quad (39)$$

For each $\alpha \in \mathcal{A}_j$, the terms $\sum_{i:(j,i) \in E'} \eta_j A_{\alpha\gamma_i}^{ji}$ and $\sum_{i:(j,i) \in E'} B_{\alpha\gamma_i}^{ji}$ 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 $\gamma_i \in \mathcal{A}_i$. From Property 3, we conclude for each $\alpha \in \mathcal{A}_j$ that

$$\sum_{i:(j,i) \in E'} (-\eta_j A_{\alpha\gamma_i}^{ji} + B_{\alpha\gamma_i}^{ji}) = 0. \quad (40)$$

Relating (40) back to (39) and then (38), for each $j \in V$, we obtain

$$\sum_{i:(j,i) \in E'} (x_j - x_j^*)^\top B^{ji} x_i^* = \sum_{i:(j,i) \in E'} \eta_j (x_j - x_j^*)^\top A^{ji} x_i^*. \quad (41)$$

Finally, combining (41) and (37), 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 $\alpha \in \mathcal{A}_j$ and any linear combination thereof. Consequently, we conclude that $\Phi(t) = \Phi(0)$ for all $t \geq 0$.

Orbits remain bounded away from the boundary. To complete the proof, we use the constant of motion to show that the orbits of the replicator dynamics for rescaled zero sum polymatrix games remain bounded away from the boundary. Indeed, let x be an interior point which is not an equilibrium. That is, each $x_i \in \text{int}(\Delta^{n-1})$. Let $\gamma(x)$ be the forward orbit of x i.e.,

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

Then, Lemma 4.2 implies that for any $y \in \gamma(x)$,

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

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

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

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

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

Let

$$\varepsilon = \min_{i \in V, \beta \in \mathcal{A}_i} \exp(-c/x_{i\beta}^*)$$

so that for any $y \in \gamma(x)$ and any player $i \in V$ and strategy $\beta \in \mathcal{A}_i$, $y_{i\beta} \geq \varepsilon > 0$ which, in turn, implies that $\gamma(x)$ is bounded away from the boundary.

KL-Divergence Constant of Motion. The constant of motion from Lemma 4.2 is equivalently given as

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

where $\mathcal{I}(\cdot)$ denotes the entropy and $\text{KL}(\cdot || \cdot)$ denotes the Kullback-Leibler (KL) divergence. Since $\sum_{i \in V} \eta_i \mathcal{I}(x_i^*)$ 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 B.1. *Under the assumptions of Lemma 4.2, $\Psi(t) = \sum_{i \in V} \eta_i (\text{KL}(x_i^* || x_i))$ is also a constant of motion.*

B.3 Proofs of Time-Average Convergence, Equilibrium Computation, & Bounded Regret from Section 5

We now provide the proofs of Theorem 5.1, Theorem 5.2, and Proposition 5.1 from Section 5. Appendix B.3.1 contains the proof of Theorem 5.1, which shows the time-average equilibria and utility convergence of the replicator dynamics in rescaled zero-sum polymatrix games. The proof of Theorem 5.2, which provides a linear program to compute Nash equilibrium in rescaled zero-sum polymatrix games is in Appendix B.3.2. Finally, the proof of Proposition 5.1, which shows that the replicator dynamics achieve bounded regret, is given in Appendix B.3.3.

B.3.1 Proof of Theorem 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 4.2. Integrating the replicator dynamics from (3) given by

$$\dot{x}_{i\alpha}(t) = x_{i\alpha}(t)(u_{i\alpha}(x(t)) - u_i(x(t)))$$

for each agent $i \in V$ and each strategy $\alpha \in \mathcal{A}_i$, 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,

$$\frac{1}{T} \int_{x(0)}^{x(T)} \frac{1}{x_{i\alpha}(\tau)} dx(\tau) = \frac{1}{T} (\log x_{i\alpha}(T) - \log x_{i\alpha}(0))$$

so that

$$\frac{1}{T} \int_0^T (u_{i\alpha}(x(\tau)) - u_i(x(\tau))) d\tau = \frac{1}{T} (\log x_{i\alpha}(T) - \log x_{i\alpha}(0)). \quad (42)$$

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 $\alpha \in \mathcal{A}_i$, let $\bar{x}_i = (\bar{x}_1, \dots, \bar{x}_{n_i})$ 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 (42) 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

$$\lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T (u_{i\alpha}(x(\tau)) - u_i(x(\tau))) d\tau = 0, \quad \forall \alpha \in \mathcal{A}_i. \quad (43)$$

Rearranging (43), for each $i \in V$, we have that

$$\begin{aligned} \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T u_i(x(\tau)) d\tau &= \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T u_{i\alpha}(x(\tau)) d\tau, \quad \forall \alpha \in \mathcal{A}_i \\ &= \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T \sum_{j:(i,j) \in E} (A^{ij} x_j(\tau))_\alpha d\tau, \quad \forall \alpha \in \mathcal{A}_i \\ &= \sum_{j:(i,j) \in E} (A^{ij} \bar{x}_j)_\alpha, \quad \forall \alpha \in \mathcal{A}_i, \end{aligned} \quad (44)$$

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 $\alpha \in \mathcal{A}_i$, we have that

$$\begin{aligned} u_i(\bar{x}) &= \sum_{\alpha \in \mathcal{A}_i} \bar{x}_{i\alpha} \sum_{j:(i,j) \in E} (A^{ij} \bar{x}_j)_\alpha \\ &= \sum_{\alpha \in \mathcal{A}_i} \bar{x}_{i\alpha} \left(\lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T u_i(x(\tau)) d\tau \right) \\ &= \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T u_i(x(\tau)) d\tau \end{aligned}$$

where the last equality holds since $\sum_{\alpha \in \mathcal{A}_i} \bar{x}_{i\alpha} = 1$. In turn, the above implies that

$$u_i(\bar{x}) = \sum_{j:(i,j) \in E} (A^{ij} \bar{x}_j)_\alpha = u_{i\alpha}(\bar{x}), \quad \forall \alpha \in \mathcal{A}_i$$

so that $\bar{x} = (\bar{x}_1, \dots, \bar{x}_N)$ 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 (44), we have that

$$\lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T u_i(x(\tau)) d\tau = u_{i\alpha}(x^*) = u_i(x^*)$$

which proves (ii).

B.3.2 Proof of Theorem 5.2

Let OPT denote the optimal value of the linear program

$$\begin{aligned} \min_{x \in \mathcal{X}} \quad & \sum_{i=1}^n \eta_i v_i \\ \text{s.t.} \quad & v_i \geq u_{i\alpha}(x), \quad \forall i \in V, \forall \alpha \in \mathcal{A}_i. \end{aligned} \quad (45)$$

We begin by proving that $\text{OPT} \leq 0$. Since a Nash equilibrium always exists (Nash, 1951), there exists a strategy profile x such that $\max_{\alpha \in \mathcal{A}_i} u_{i\alpha}(x) = u_i(x)$. That is,

$$\max_{\alpha \in \mathcal{A}_i} \sum_{j:(i,j) \in E} (A^{ij} x_j)_\alpha = \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha} \sum_{j:(i,j) \in E} (A^{ij} x_j)_\alpha, \quad \forall i \in V. \quad (46)$$

Let $v_i = \max_{\alpha \in \mathcal{A}_i} \sum_{j:(i,j) \in E} (A^{ij}x_j)_\alpha$ for all $i \in V$. Then, the pair of vectors (v, x) forms a feasible solution for the linear program in (45). As a result, using (46), we have that

$$\text{OPT} \leq \sum_{i \in V} \eta_i v_i = \sum_{i \in V} \eta_i \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha} \sum_{j:(i,j) \in E} (A^{ij}x_j)_\alpha = 0$$

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 (45). We now prove that x^* is a Nash equilibrium using the fact that $\text{OPT} \leq 0$. 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 $\alpha \in \mathcal{A}_i$ satisfying

$$\max_{\alpha \in \mathcal{A}_i} u_{i\alpha}(x^*) = \max_{\alpha \in \mathcal{A}_i} \sum_{j:(i,j) \in E} (A^{ij}x_j^*)_\alpha > \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}^* \sum_{j:(i,j) \in E} (A^{ij}x_j^*)_\alpha = u_i(x^*). \quad (47)$$

Moreover, since (v^*, x^*) is the optimal solution of the linear program in (45), we know that $v_i^* \geq u_{i\alpha}(x^*)$ for all $i \in V$ and $\alpha \in \mathcal{A}_i$, which then implies $v_i^* \geq \max_{\alpha \in \mathcal{A}_i} u_{i\alpha}(x^*)$ for all $i \in V$. As a direct result, we obtain the inequality

$$\text{OPT} = \sum_{i \in V} \eta_i v_i^* \geq \sum_{i \in V} \eta_i \max_{\alpha \in \mathcal{A}_i} \sum_{j:(i,j) \in E} (A^{ij}x_j^*)_\alpha \quad (48)$$

Finally, combining (47) and (48), we get that

$$\text{OPT} > \sum_{i \in V} \eta_i \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha}^* \sum_{j:(i,j) \in E} (A^{ij}x_j^*)_\alpha = 0,$$

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 $\text{OPT} \leq 0$, which means x^* must be a Nash equilibrium.

B.3.3 Proof of Proposition 5.1

We begin by presenting preliminaries and notation needed for an intermediate technical result. Denote by $v_i(x) = (u_{i\alpha}(x))_{\alpha \in \mathcal{A}_i}$ the payoff vector for any agent $i \in V$ that includes the utility of each pure strategy $\alpha \in \mathcal{A}_i$ 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

$$\begin{aligned} y_i(t) &= \int_0^t v_i(x(s)) ds \\ x_i(t) &= Q_i(y_i(t)) \end{aligned} \quad (49)$$

characterize the ‘‘Follow the Regularized Leader’’ updates for player $i \in V$ at time $t \geq 0$. The so-called choice map $Q_i : \mathbb{R}^{n_i} \rightarrow \mathcal{X}_i$ 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 $h_i : \mathcal{X}_i \rightarrow \mathbb{R}$. 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

$$h_i(x_i) = \sum_{\alpha \in \mathcal{A}_i} x_{i\alpha} \log(x_{i\alpha})$$

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 inequality, 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., (Shalev-Shwartz et al., 2012, 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 5.1.

Lemma B.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), then independent of the rest of the players in the game,*

$$\max_{x_i \in \mathcal{X}_i} \int_0^t \langle v_i(x(s)), x_i \rangle ds - \int_0^t \langle v_i(x(s)), x_i(s) \rangle ds \leq h_{\max,i} - h_{\min,i}.$$

Proof of Lemma B.1. We begin by deriving a bound for every fixed $x_i \in \mathcal{X}_i$ on the expression

$$\int_0^t \langle v_i(x(s)), x_i \rangle ds. \quad (50)$$

From the definition of the utility dynamics given by

$$y_i(t) = \int_0^t v_i(x(s)) ds,$$

an equivalent representation of (50) is

$$\int_0^t \langle v_i(x(s)), x_i \rangle ds = \langle y_i(t), x_i \rangle. \quad (51)$$

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 (51), we get

$$\int_0^t \langle v_i(x(s)), x_i \rangle ds \leq h_i^*(y_i(t)) + h_{\max,i}. \quad (52)$$

We now work on obtaining a bound for $h_i^*(y_i(t))$. Observe that by definition

$$\begin{aligned} h_i^*(y_i(t)) &= \langle y_i(t), Q_i(y_i(t)) \rangle - h_i(Q_i(y_i(t))) \\ &= \int_0^t \langle v_i(x(s)), Q_i(y_i(s)) \rangle ds - h_i(Q_i(y_i(t))). \end{aligned}$$

Now define the function

$$\phi : (z, t) \mapsto \int_0^t \langle v_i(z(s)), z(s) \rangle ds - h(z(t)).$$

For any fixed $t \geq 0$, we can verify by the maximizing argument (see, e.g., (Shalev-Shwartz et al., 2012, §2.7)) that $Q_i(y_i(t))$ maximizes $\phi(\cdot, t)$, so we can apply the envelope theorem (Milgrom and Segal, 2002) 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

$$h_i^*(y_i(t)) - h_i^*(y_i(0)) = \int_0^t \langle v_i(x(s)), Q_i(y_i(s)) \rangle ds.$$

Since $h_i^*(y_i(0)) = -h_{\min,i}$, we get the bound

$$h_i^*(y_i(t)) \leq \int_0^t \langle v_i(x(s)), Q_i(y_i(s)) \rangle ds - h_{\min,i}.$$

Finally, combining the previous equation with (52), we conclude the stated result of

$$\max_{x_i \in \mathcal{X}_i} \int_0^t \langle v_i(x(s)), x_i \rangle ds - \int_0^t \langle v_i(x(s)), x_i(s) \rangle ds \leq h_{\max,i} - h_{\min,i}.$$

□

We now return to proving Proposition 5.1. By definition $u_i(x) = \langle v_i(x), x_i \rangle$, which means we can directly apply Lemma B.1 to the regret definition. We now do so and obtain the stated result:

$$\begin{aligned} \text{Reg}_i(t) &= \max_{x_i \in \mathcal{X}_i} \frac{1}{t} \int_0^t (u_i(x_i, x_{-i}(s)) - u_i(x(s))) ds \\ &= \max_{x_i \in \mathcal{X}_i} \frac{1}{t} \int_0^t \langle v_i(x(s)), x_i - x_i(s) \rangle ds \\ &\leq \frac{h_{\max,i} - h_{\min,i}}{t} = \frac{\Omega_i}{t}. \end{aligned}$$

C Supplementary Experiments and Details

The focus of this section is to provide supplementary simulations and details on our experimental methodology. We provide further simulations of the time-evolving generalized rock-paper-scissors game in Appendix C.1. Then, in Appendix C.2, we present simulations on a 5-player rescaled zero-sum polymatrix game. In Appendix C.3, we provide simulations for larger systems. Finally, Appendix C.4 contains a description of our simulation environment and methods to allow for easy reproduction of the experimental results.

C.1 Simulations of Time-Evolving Generalized Rock-Paper-Scissors Game

As mentioned in the introduction, there is a breadth of work studying the emergence of recurrent behavior of replicator dynamics in network zero-sum games (Piliouras et al., 2014; Piliouras and Shamma, 2014; Boone and Piliouras, 2019; Mertikopoulos et al., 2018; Nagarajan et al., 2020; Perolat et al., 2020). The canonical 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 since the Nash equilibrium of the game is not fixed. This is in fact a key technical challenge that we overcome in this work. To illustrate this, we consider the time-evolving generalized rock-paper-scissors game proposed by Mai et al. (2018). We show the evolution of the Nash equilibrium over time in Figure 7a, the evolution of the population strategy vector in Figure 7b, and the KL divergence between the evolving equilibrium and the replicator trajectory in Figure 7c. Clearly, the Nash equilibrium is no longer static and furthermore the KL divergence is not a constant of motion. This precludes the opportunity to follow standard proof techniques for showing replicator dynamics are recurrent in time-evolving games.

We solve this technical challenge by reducing time-evolving dynamics to a static polymatrix game and then proving a constant of motion. Indeed, for the time-evolving generalized rock-paper-scissors game, we can verify that the constant of motion from Corollary B.1 holds empirically. Figure 8 shows the weighted sum of KL-divergences from the equilibrium of the rescaled zero-sum game we obtain from our reduction to the strategy of each player along the replicator trajectory is constant. In other words, while a constant of motion exists for the time-evolving generalized rock-paper-scissors game, it is not the obvious choice.

In Figure 9a, 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 9b, 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.

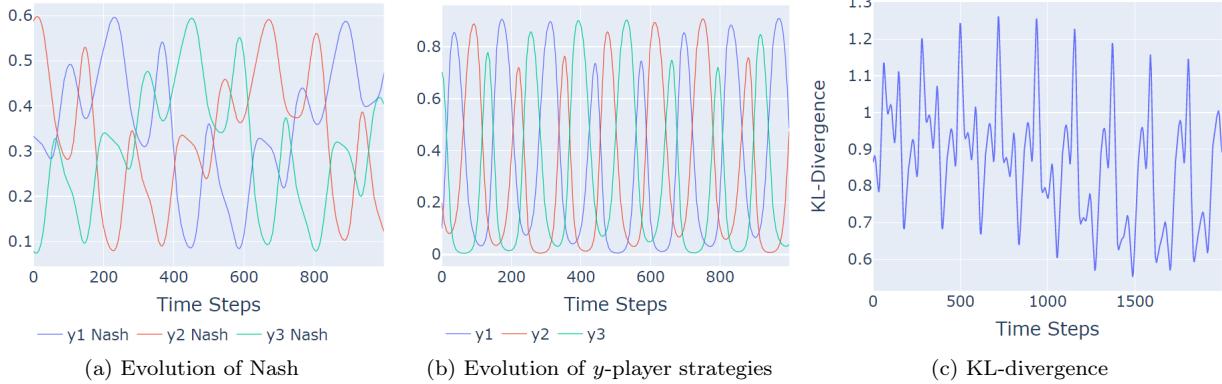


Figure 7: Nash equilibrium of the time-evolving game, evolving strategies and the KL divergence between the Nash and strategies, from left to right, under replicator dynamics for the time-evolving generalized RPS mode (Mai et al., 2018).

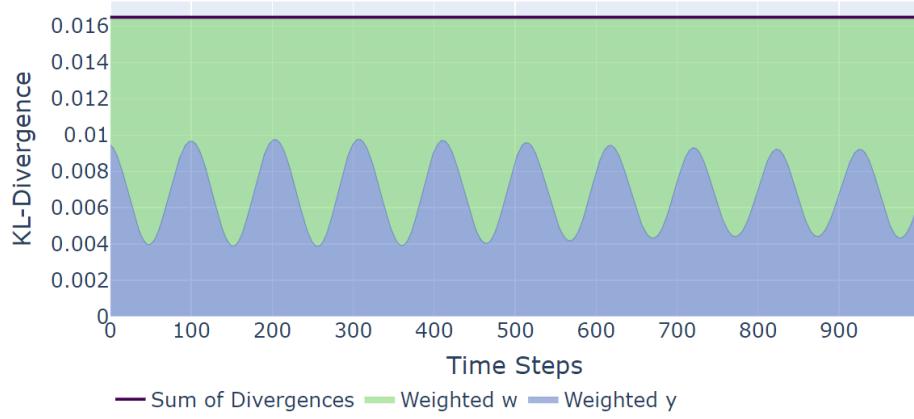


Figure 8: Constant sum of KL-divergence for time-evolving generalized rock-paper-scissors game.

To visualize the multidimensional system behavior while retaining the maximum amount of information, we generated Figure 10, which transforms the 3-dimensional data for players y and w respectively to two dimensions. To be precise, the transformations are given as follows:

$$\begin{aligned} y'_1 &= \frac{\sqrt{2}}{2}y_3 - \frac{\sqrt{2}}{2}y_2, & y'_2 &= -\frac{1}{\sqrt{6}}y_3 - \frac{1}{\sqrt{6}}y_2 + \frac{\sqrt{2}}{\sqrt{3}}y_1 \\ w'_1 &= \frac{\sqrt{2}}{2}w_3 - \frac{\sqrt{2}}{2}w_2, & w'_2 &= -\frac{1}{\sqrt{6}}w_3 - \frac{1}{\sqrt{6}}w_2 + \frac{\sqrt{2}}{\sqrt{3}}w_1 \end{aligned}$$

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 10 and the corresponding animation are analogous to Figure 1c and its corresponding animation, but the transformation method allows for visualization of all 6 dimensions instead of a subset of 4 dimensions.

Another important property of the considered population/environment dynamics is that both the time-average vector produced by the replicator dynamics and the time-average utilities converge to the equilibrium values (see Theorem 5.1 of Section 4). In Figures 11a and 11b we plot the time-average of the population y and environment w in the time-evolving generalized rock-paper-scissors game, all of which converge to $1/3$ which is

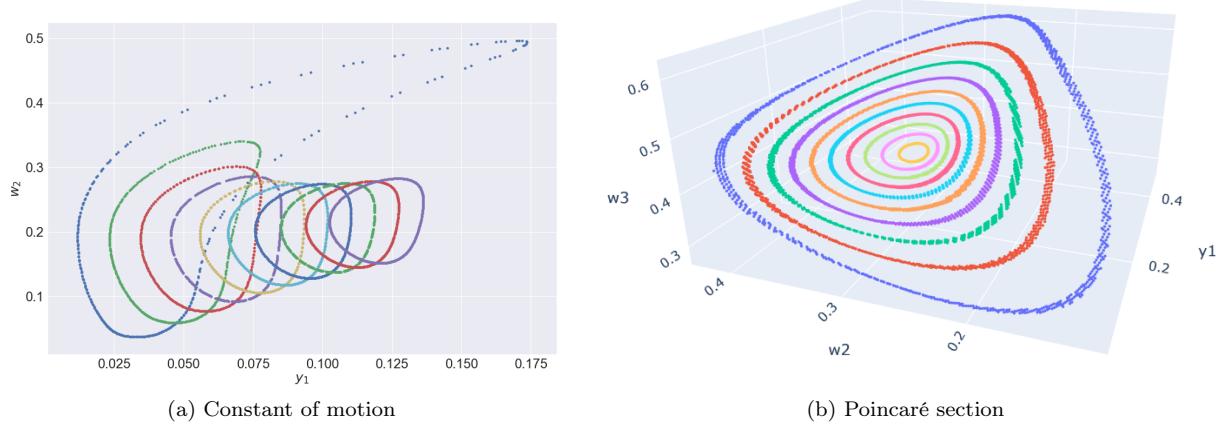


Figure 9: (a) 2D Poincaré section at $y_2 - y_1 - w_2 + w_1 = 0$ with 10 trajectories of initial conditions $\{[0.5, 0.01k, 0.5 - 0.01k, 0.5, 0.25, 0.25]\}_{k=1}^{10}$, (b) Side view of Poincaré section at $y_2 + y_1 + w_2 + w_1 = 4/3$ with 10 trajectories of initial conditions $\{[1/3, 0.03k, 2/3 - 0.03k, 1/3, 1/3, 1/3]\}_{k=1}^{10}$.

the equilibrium strategy. Similarly in Figures 11c and 11d we plot the time-average utilities of the population y and environment w converging to the equilibrium utility which in the considered instance is zero.

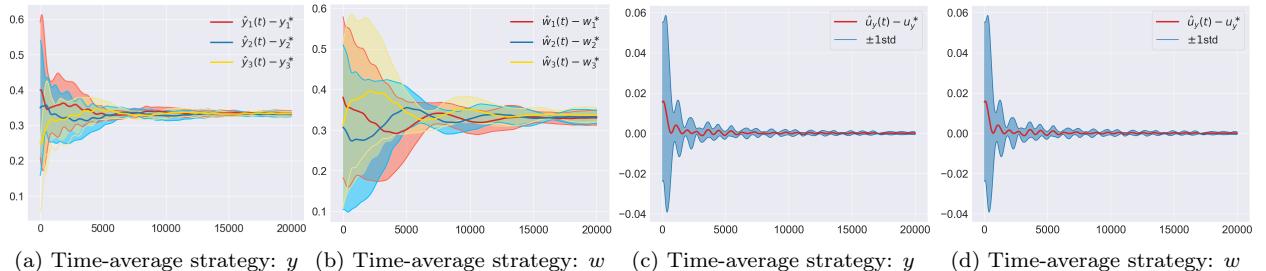


Figure 11: (a-b) Time-average for y and w converging to Nash, (c-d) Time-average utility converging with bounded regret.

C.2 Simulations on 5-player Rescaled Zero-Sum Polymatrix Game

We simulate the rescaled zero-sum polymatrix game depicted in Figure 12 where each player has 3 actions. As shown in Figure 13, the weighted sum of Kullback-Leibler (KL) divergences of each agent's strategy from the equilibrium is a constant of motion, demonstrating the bounded orbits property of Lemma 4.2. In the simulation $\mu_1 = 0.1, \mu_2 = 0.5, \mu_3 = 0.8, \mu_4 = 0.5$ and the initial condition was $[0.3, 0.4, 0.3, 0.2, 0.1, 0.7, 0.5, 0.3, 0.2, 0.7, 0.2, 0.1, 0.4, 0.2, 0.4]$. We also include the time averages of the trajectories and utility for player x_3 in Figure 14. 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.

C.3 Simulations on Large-Scale 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.

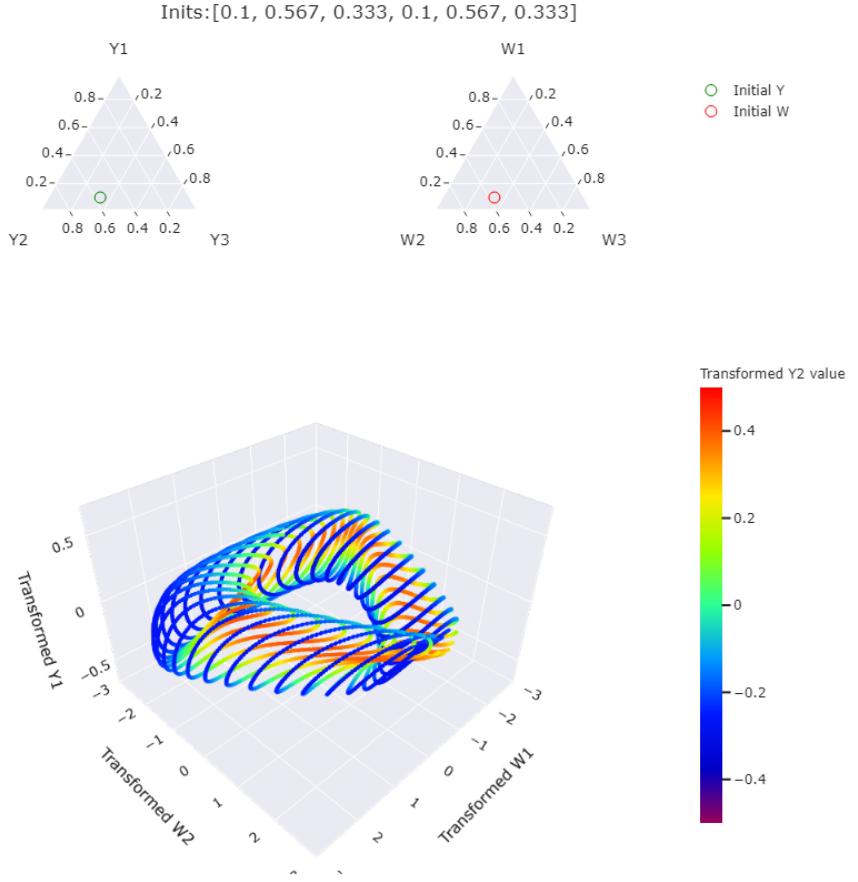


Figure 10: 4D embedding of trajectories for a range of initial conditions.

Showing Poincaré recurrence in large-scale games. 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.

Due to the presence of the sigmoid function, we expect to see a mostly dark or mostly bright screen whenever the strategies are far from the central value of 0.5, and after creating several animations, this hypothesis is confirmed. Indeed, in Figure 4 we see that the grid very quickly transforms into something that does not resemble the original at all. After a number of iterations, the recurrence property causes the Pikachu (or at least, something that looks similar to 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 scaled 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.

Constant KL-divergence with complex graph structures. 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 6. However, our theory extends to more

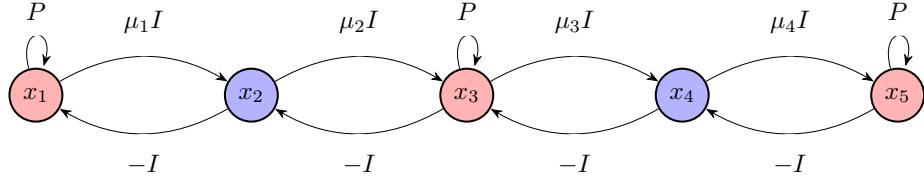


Figure 12: Each node represents a player, with different initial strategies and values of μ_i .



Figure 13: Weighted KL divergence for five player time-evolving RPS game for 1000 iterations.

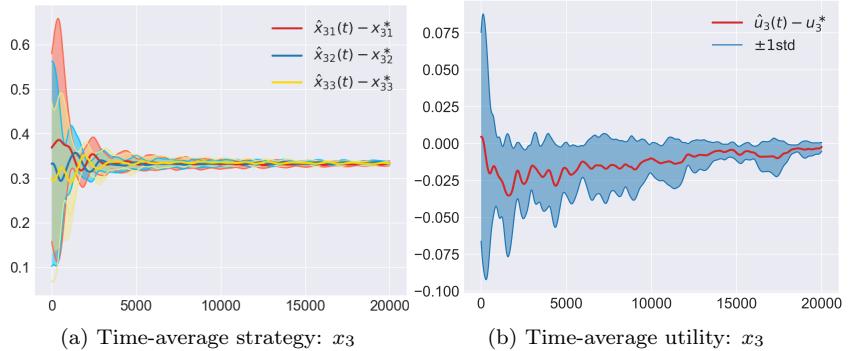


Figure 14: (a) Time-average trajectories for x_3 showing convergence to Nash. (b) Time-average utility convergence for x_3 player with bounded regret. Initial conditions are fixed values chosen uniformly at random on the simplex for x_1, x_2, x_4, x_5 and for x_3 , they take values in $(z, 0.75 - z, 0.25)$ for each $z \in \{0.1 + \frac{2k}{30}, k \in \{0, \dots, 9\}\}$.

than just graphs of that form. So long as the graphs are formed from the basic building blocks in Figure 5, we will see similar results. We performed experiments using extensions of the ‘butterfly’ graph shown in Figure 2, where each red node is a population of species and each blue node is an environment. 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 and 16, we see that despite the much more complex graph structure and many nodes, the weighted sum of divergences again sums up to a constant value. In the supplementary code, we also

present an animation that shows a grid where each element represents the strategy of a node. Similarly to the Pikachu example above, we expect to see the same image after some number of iterations, but unfortunately due to the density of the graph, this would take a far larger number of iterations than our integration allows in order to achieve recurrence.

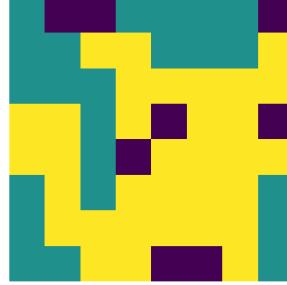


Figure 15: 8×8 grid of colors generated by sigmoid function

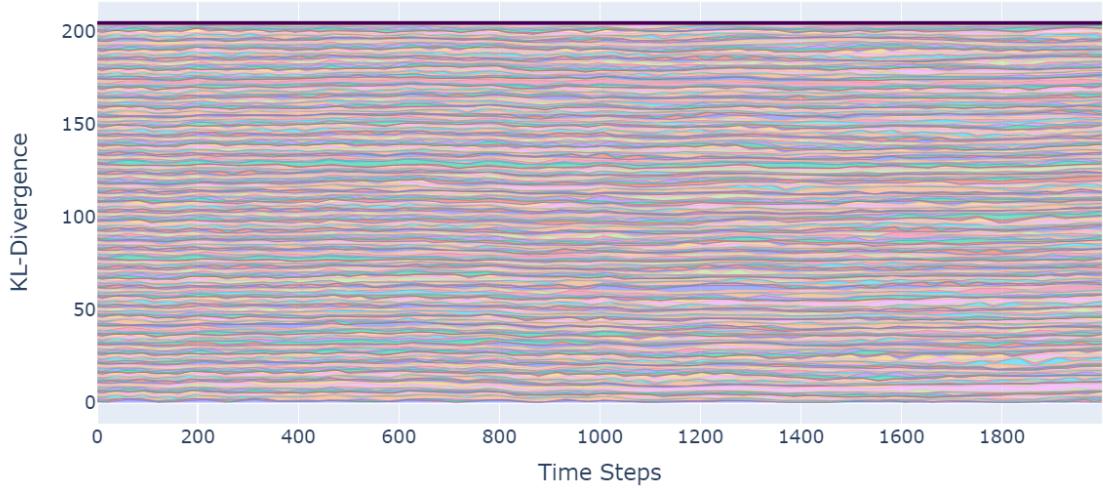


Figure 16: KL Divergences in 400-player rescaled zero-sum RPS game. Note that the sum of divergences is still constant despite the large number of nodes/players.

C.4 Implementation Details

The code used to generate the simulations in this paper has been compiled into a Jupyter notebook for ease of viewing. A HTML render of the notebook can be found at the following [link](#), while the full repository is [here](#). Our code is in Python 3.6 and only requires basic scientific computing packages such as NumPy and SciPy and data visualization packages such as Matplotlib and Plotly. Most of the code in the submission has been edited so that it can easily be executed on any standard computer in a matter of minutes as it is not computationally intensive.