Proceedings of Machine Learning Research vol 134:1–27, 2021 34th Annual Conference on Learning Theory

Learning in Matrix Games can be Arbitrarily Complex

Gabriel P. Andrade GABRIEL.ANDRADE@COLORADO.EDU
University of Colorado Boulder

Rafael Frongillo RAF@COLORADO.EDU
University of Colorado Boulder

Georgios Piliouras GEORGIOS@SUTD.EDU.SG
Singapore University of Technology and Design

Editors: Mikhail Belkin and Samory Kpotufe

Abstract

Many multi-agent systems with strategic interactions have their desired functionality encoded as the Nash equilibrium of a game, e.g. machine learning architectures such as Generative Adversarial Networks. Directly computing a Nash equilibrium of these games is often impractical or impossible in practice, which has led to the development of numerous learning algorithms with the goal of iteratively converging on a Nash equilibrium. Unfortunately, the dynamics generated by the learning process can be very intricate and instances failing to converge become hard to interpret. In this paper we show that, in a strong sense, this dynamic complexity is inherent to games. Specifically, we prove that replicator dynamics, the continuous-time analogue of Multiplicative Weights Update, even when applied in a very restricted class of games–known as finite matrix games–is rich enough to be able to approximate arbitrary dynamical systems. In the context of machine learning, our results are positive in the sense that they show the nearly boundless dynamic modelling capabilities of current machine learning practices, but also negative in implying that these capabilities may come at the cost of interpretability. As a concrete example, we show how replicator dynamics can effectively reproduce the well-known strange attractor of Lonrenz dynamics (the "butterfly effect", Fig 1) while achieving no regret.

Keywords: Game Theory, Online Learning, Replicator Dynamics, Regret, Nash Equilibria, Dynamical Systems, Attractors

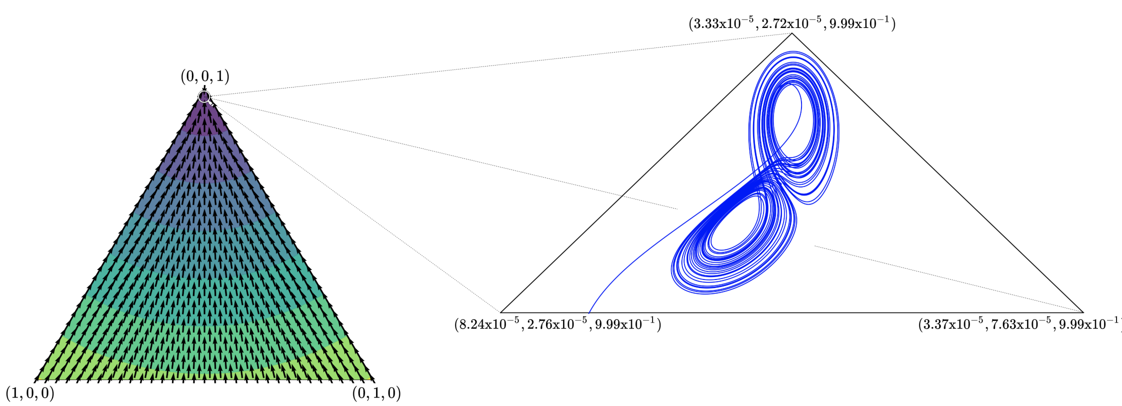

Figure 1: A no-regret strange attractor. The Lorenz system can be embedded in replicator dynamics on a finite matrix game. For more details see Sections §4 and Appendix A.3.

1. Introduction

Game theory has emerged as a powerful formalism for studying machine learning settings with multiple interacting loss functions. These settings are ubiquitous; they arise explicitly from the goal of machine learning tasks (e.g. chess, poker, Go) or implicitly in the design of architectures such as Generative Adversarial Networks (Goodfellow et al., 2014; Balduzzi et al., 2020). Rather than the classical algorithms which simply minimize a single loss function, problems with multiple loss functions require algorithms which converge to a Nash equilibrium. However, in practice, many games underlying multi-loss machine learning have an implicit complexity (e.g. from datasets and the deep networks used) that makes formal analyses of general settings infeasible using current techniques. In light of this complexity, since many well-studied classes of games equate to idealized multi-loss problems, studying the dynamics of learning in these simpler settings becomes an important stepping stone for machine learning theory. Inspired by this, recent work has addressed questions traditionally studied in evolutionary game theory by trying to derive and understand algorithms with strong equilibrium convergence guarantees in classic game theoretic settings.

Unfortunately, even for simple games, we lack learning algorithms that provably find such equilibria in general. In the presence of multiple interacting loss functions, the standard toolbox of learning algorithms often fails in unpredictable ways. Recent work has shown that, even under the simplifying assumption of perfect competition (zero-sum games and variants), instead of converging to Nash equilibria the dynamics of standard learning algorithms can cycle (Mertikopoulos et al., 2018) diverge (Bailey and Piliouras, 2018), or even be formally chaotic (Cheung and Piliouras, 2019). Moreover, when one broadens their scope to a more general class of games, experimental results suggest that chaos is in fact typical behaviour (Sanders et al., 2018) and can even emerge in low-dimensional systems (Palaiopanos et al., 2017). Considering the ubiquity of multi-loss ML settings alongside these negative results for relatively simple games, and others discussed in §1.1, an urgent question arises: *Is there any hope for a general understanding of the behaviours arising from optimization-driven dynamics in games?*

This paper provides evidence that the answer to this question is likely to be "no". We show that the dynamics of even, arguably, the most well-studied evolutionary learning algorithms, even in a simple and seemingly very constrained class of games, can approximate arbitrarily complex dynamical systems.

Informal Main Theorem. *Replicator learning dynamics on a matrix game can, after a transitionary period of time, approximate essentially* any C^1 *dynamical system with arbitrary precision.*

The significance of our result is clear when one considers that matrix games are a very restricted class of games and that replicator dynamics is a special case of Follow-the-Regularized-Leader (FTRL) dynamics, which captures multiple popular learning algorithms such as gradient descent, hedge, multiplicative weights, etc. as a special case and enjoys vanishing regret at a rate of $O(1/T)$ (see e.g. Mertikopoulos et al. (2018)). Since matrix games are a simple class of games and replicator dynamics is a special case of FTRL, then the dynamics of more general classes of games and learning algorithms cannot be any simpler than the case we have shown to be arbitrarily complex; framed like this, our result can be interpreted in the same fashion as a reduction in computational complexity theory. When understood in this way, our result implies that understanding learning dynamics in multi-agent machine learning settings is akin to a general understanding of dynamical systems and has multi-faceted implications depending on the context it is being interpreted in. We discuss a range of such implications in §5, including what our result implies for designing learning algorithms and the use of regret for measuring learning performance.

A formal statement of the main theorem requires carefully exploring strong notions of equivalence between dynamical systems, and how these notions can be used to meaningfully define approximations of dynamical systems. All of the requisite language and formalism is introduced in §2, while our notion of approximation and the main result is given in §3. Our proof establishes connections between game theory, topological dynamics, learning theory, and standard population models from mathematical ecology. In fact, the question explored in this paper mirrors one famously asked nearly fifty years ago by Smale (1976)–about whether it is possible to meaningfully understand and predict the behaviours of well-studied ecological models of competition. The class of systems considered in this paper are, relatively speaking, even more restricted than the ones considered by Smale in his construction. However, as we show through a sequence of transformations and embeddings, we can approximately capture the behaviour of any target system using replicator dynamics in finite dimensional matrix games.

1.1. Related Work

Optimization-driven learning in games, e.g., regret-minimizing dynamics, has been the subject of intense study. The standard approach focuses on their time-averaged behaviour and its convergence to coarse correlated equilibria in games, (see e.g. Roughgarden (2015); Stoltz and Lugosi (2007)). The analysis of the time-averaged behaviour, however, is unable to faithfully capture the day-to-day dynamics. In many cases, it has been shown that the emergent day-to-day behaviour is non-convergent in a strong formal sense (Mertikopoulos et al., 2018; Bailey and Piliouras, 2019). Perhaps even more alarming is the fact that strong time-average convergence guarantees may hold true regardless of whether the underlying system is convergent, recurrent, or even chaotic (Palaiopanos et al., 2017; Chotibut et al., 2020a,b; Cheung and Piliouras, 2019, 2020; Bailey et al., 2020). In fact, all FTRL dynamics, despite their optimal regret guarantees, fail to achieve (even local) asymptotic stability on *any* (even partially) mixed Nash equilibrium in effectively *all* games (Flokas et al., 2020).

With the proliferation of multi-agent architectures in machine learning, e.g., Generative Adversarial Networks (GANs), recent work has placed particular attention on the modes of failure arising in variants of zero-sum competition between learning agents (e.g. between two neural networks). In zero-sum games the dynamics of standard learning algorithms such as gradient descent do not converge to Nash equilibria. Instead, the resultant dynamics may lead to cycling (Mertikopoulos et al., 2018; Vlatakis-Gkaragkounis et al., 2019; Boone and Piliouras, 2019; Balduzzi et al., 2018), divergence (Bailey and Piliouras, 2018; Cheung, 2018), or formally chaotic behaviours (Cheung and Piliouras, 2019, 2020). In the face of such strong negative results for out-of-the-box optimization methods the development of tailored algorithmic solutions is incentivized, e.g. Daskalakis et al. (2018); Mertikopoulos et al. (2019); Gidel et al. (2019); Mescheder et al. (2018); Perolat et al. (2020); Yazıcı et al. (2019). However, even when these algorithms do equilibrate, they may stabilize at fixed points that are not Nash equilibria and thus not game theoretically meaningful (Adolphs et al., 2019; Daskalakis and Panageas, 2018).

Alongside studies of learning in zero-sum games, differential games (i.e. smooth games) have been the focus of recent research as a powerful, and more general, model of multi-agent machine learning (e.g. Balduzzi et al. (2018); Mazumdar et al. (2020)). Letcher et al. (2019) leveraged connections with Hamiltonian dynamics to design new algorithms for training GANs while "correcting" cyclic behaviours. In addition, Balduzzi et al. (2020) explored the structure of differential games and revealed promising training guarantees when relatively weak constraints are placed on the loss functions of agents in the model and the payoff structure of their interactions. Within the space of differential games, the dynamics of non-convex non-concave games have received particular attention and a number of distinct non-equilibrating failure modes have been catalogued (Vlatakis-Gkaragkounis et al., 2019; Hsieh et al., 2020). The impossibility of universal algorithmic solutions within the broad scope of differential games has also been reinforced by recent work that constructs a simple example where reasonable gradient-based methods cannot hope to converge (Letcher, 2021).

Even when one restricts their attention on matrix games, the difficulty of learning Nash equilibria grows significantly and swiftly when one broadens their scope to a more general class of games than just zero-sum games (Daskalakis et al., 2010; Kleinberg et al., 2011; Galla and Farmer, 2013; Papadimitriou and Piliouras, 2019). In fact, detailed experimental studies suggest that chaos is standard fare (Sanders et al., 2018) and emerges even in very low dimensional systems (Sato et al., 2002; Palaiopanos et al., 2017; Pangallo et al., 2017). This abundance of non-equilibrating results has inspired a program for linking game theory to topology of dynamical systems (Papadimitriou and Piliouras, 2018, 2019), specifically to Conley's fundamental theorem of dynamical systems (Conley, 1978). This approach shifts attention from Nash equilibria to a more general notion of recurrence, called chain recurrence, that is flexible enough to capture both cycling behavior as well as chaos. These tools have since found application in multi-agent ML settings (Omidshafiei et al., 2019; Rowland et al., 2019).

Thus far, almost all work on learning dynamics in games can be roughly broken into two streams: (i) designing algorithms that converge to desirable states, and (ii) characterizing the possible emergent behaviors from a given class of game dynamics. Our work differs from both these lines of inquiry by, in a sense, doing the *converse* of (ii). Roughly speaking, we ask the question "Given a target dynamical system, can we construct a game whose learning dynamics behave in a similar fashion?" To the best of our knowledge, our work is the first construction of this sort in the context of learning in games. Our approach is inspired by the work of Smale (1976) and Hirsch (1988) in mathematical ecology, which have developed constructions in the same spirit as ours to study the dynamics of population models.

2. Preliminaries

2.1. Game Theory

A *matrix game* (*finite* 2*-player normal form game*) is defined on a set of two agents $[2] = \{1, 2\}$. Agent $i \in [2]$ chooses actions from a finite action set S_i according to a distribution $\mathbf{x}_i$ in the probability $|S_i|$-simplex $\Delta^{|S_i|} = \{\mathbf{x}_i \in \mathbb{R}_+^{|S_i|} : \sum_{s \in S_i} x_{is} = 1\}$. The probability distribution $\mathbf{x}_i$ is known as i's *mixed strategy*. As the name indicates, agents in a matrix game receive payoffs according to a payoff matrix $A_{i,j} \in \mathbb{R}^{|S_i| \times |S_j|}$ where $i, j \in [2]$ and $i \neq j$. Given that mixed strategies $\mathbf{x}_1 \in \Delta^{|S_1|}$ and $\mathbf{x}_2 \in \Delta^{|S_2|}$ are chosen, agent 1 receives payoff $\mathbf{x}_1^\top A_{1,2}\mathbf{x}_2$ and agent 2 receives payoff $\mathbf{x}_2^\top A_{2,1}\mathbf{x}_1$. This gives rise to two optimization problems, one per agent, where agents act strategically and independantly to maximize their expected payoff over the other agent's mixed strategy, i.e.

$$\max_{\mathbf{x}_i \in \Delta^{|S_i|}} \mathbf{x}_i^\top A_{i,j}\mathbf{x}_j, \qquad i, j \in [2] \ \&\ i \neq j\ . \tag{1}$$

2.2. Follow-the-Regularized-Leader (FTRL) Learning and Replicator Dynamics

Arguably the most well known class of algorithms for online learning and optimization is Follow-the-Regularized-Leader (FTRL). Given initial payoff vector $\mathbf{y}_i(0)$, an agent i that plays against agent j in a matrix game $A_{i,j}$ updates their strategy at time t according to

$$\begin{aligned} \mathbf{y}_i(t) &= \mathbf{y}_i(0) + \int_0^t A_{i,j}\mathbf{x}_j(s)ds \\ \mathbf{x}_i(t) &= \arg\max_{\mathbf{x}_i \in \Delta^{|S_i|}} \{\langle \mathbf{x}_i, \mathbf{y}_i(t)\rangle - h_i(\mathbf{x}_i)\} \end{aligned} \tag{2}$$

where h_i is strongly convex and continuously differentiable. FTRL effectively performs a balancing act between exploration and exploitation. The accumulated payoff vector $\mathbf{y}_i(t)$ indicates the total payouts until time t, i.e. if agent i had played strategy $s_i \in S_i$ continuously from $t = 0$ until time t, agent i would receive a total reward of $\mathbf{y}_{is_i}(t)$. The two most well-known instantiations of FTRL dynamics are the online gradient descent algorithm when $h_i(\mathbf{x}_i) = ||\mathbf{x}_i||_2^2$, and the replicator dynamics (the continuous-time analogue of Multiplicative Weights Update Arora et al. (2012)) when $h_i(\mathbf{x}_i) = \sum_{s_i \in S_i} \mathbf{x}_{is_i} \ln \mathbf{x}_{is_i}$. FTRL dynamics in continuous time has bounded regret in arbitrary games (Mertikopoulos et al., 2018). For more information on FTRL dynamics and online optimization, see Shalev-Shwartz (2012).

In this paper, we will focusing on *replicator dynamics* (RD) as our main game dynamics. Aside from its role in optimization, RD is one of the key mathematical models of evolution and biological competition (Schuster and Sigmund, 1983; Taylor and Jonker, 1978). It is also the prototypical dynamic studied in the field of evolutionary game theory (Weibull, 1995; Sandholm, 2010). In this context, replicator dynamics can be thought of as a normalized form of the population models introduced in §2.4, and is studied given just a single payoff matrix A and a single probability distribution $\mathbf{x}$ that can be thought abstractly as capturing the proportions of different species/strategies in the current population. Species/strategies get randomly paired up and the resulting payoff determines which strategies will increase/decrease over time.

Formally, the dynamics are as follows. Let $A \in \mathbb{R}^{m \times m}$ be a matrix game and $\mathbf{x} \in \Delta^m$ be the mixed strategy played. RD on A are given by:

$$\dot{x}_i = \frac{dx_i}{dt} = x_i\left((A\mathbf{x})_i - \mathbf{x}^\top A\mathbf{x}\right), \qquad i \in [n] \tag{3}$$

Under the symmetry of $A_{i,j} = A_{j,i}$, and of initial conditions (i.e. $\mathbf{x}_i = \mathbf{x}_j$), it is immediate to see that under the $\mathbf{x}_i, \mathbf{x}_j$ solutions of (2) are identical to each other and to the solution of (3) with $A = A_{i,j} = A_{j,i}$. For our purposes, it will suffice to focus on exactly this setting of matrix games defined by a single payoff matrix A and a single probability distribution $\mathbf{x}$, which is actually the standard setting within evolutionary game theory.

2.3. Dynamical Systems Theory

Dynamical systems are mathematical models of time-evolving processes. The object undergoing change in a dynamical system is called its *state* and is often denoted by $\mathbf{x} \in \mathcal{X}$, where $\mathcal{X}$ is a topological space called a *state space*. We will be focusing on *continuous time* systems with time denoted by $t \in \mathbb{R}$. Change between states in a dynamical system is described by a *flow* $\Phi : \mathcal{X} \times \mathbb{R} \to \mathcal{X}$ satisfying two properties:

(i) For each $t \in \mathsf{R}$, $\Phi(\cdot, t) : \mathsf{X} \to \mathsf{X}$ is bijective, continuous, and has a continuous inverse.

(ii) For every $s, t \in \mathsf{R}$ and $\mathbf{x} \in \mathsf{X}$, $\Phi(\mathbf{x}, s+t) = \Phi(\Phi(\mathbf{x}, t), s)$.

Intuitively, flows serve the purpose of describing the evolution of states in the dynamical system. Given a time $t \in \mathsf{R}$, the flow describes the relative movement of every point $\mathbf{x} \in \mathsf{X}$; we will denote this by the map $\Phi^t : \mathsf{X} \to \mathsf{X}$. Similarly, given a point $\mathbf{x} \in \mathsf{X}$, the flow captures the trajectory of $\mathbf{x}$ as a function of time; in an abuse of notation, we will denote this by $\Phi^t(\mathbf{x})$ where t is changing.

When $\mathbf{x}$ changes according to a continuous function in t the dynamical system is often given as a system of *ordinary differential equations* (ODEs). Systems of ODEs describe a *vector field* $V : \mathsf{X} \to T\mathsf{X}$ which assigns to each $\mathbf{x} \in \mathsf{X}$ a vector in the tangent space of X at $\mathbf{x}$. This fact is particularly important in this paper for the case that X is Δ^n, in which case the tangent space $T\Delta^n$ at each $\mathbf{x} \in \Delta^n$ is: $\{\mathbf{y} \in \mathsf{R}^n : \|\mathbf{y}\|_1 = 0\}$ for $\mathbf{x}$ in the interior of Δ^n, and additionally "pointing inwards" for $\mathbf{x}$ on the boundary of Δ^n (i.e. $y_i \geq 0$ if $x_i = 0$). A system of ODEs is said to *generate* (resp. *give*) a flow Φ if Φ describes a solution of the ODEs at each point $\mathbf{x} \in \mathsf{X}$. Throughout this paper we will assume that all dynamical systems discussed can be given by a system of ODEs. As such, we will use the term *dynamical system* to refer to the system of ODEs, the associated vector field, and a generated flow interchangeably. Note that, for Lipschitz-continuous systems of ODEs, the generated flow is unique (see Perko (1991); Meiss (2007)) and using these terms interchangeably is well defined.

An important notion in this paper, and dynamical systems theory in general, is that of a *global attracting set* of the dynamical system. Let Φ be a flow generated by some dynamical system on X. We say $\mathsf{Y} \subseteq \mathsf{X}$ is *forward invariant* for the flow Φ if $\Phi^t(\mathbf{y}) \in \mathsf{Y}$ for every $t \geq 0$, $\mathbf{y} \in \mathsf{Y}$. We say $\mathsf{Y} \subseteq \mathsf{X}$ is *globally attracting* for the flow Φ if Y is nonempty, forward invariant, and

$$\mathsf{Y} \supseteq \bigcap_{t>0} \{\Phi^t(\mathbf{x}) : \mathbf{x} \in \mathsf{X}\} . \tag{4}$$

Intuitively speaking, if Y is globally attracting it will capture the dynamics of Φ starting from any point in X after some transitionary period of time. In §3 we also use the notion of *stationary* dynamics, which is often considered "uninteresting" in dynamical systems theory since, in a sense, it describes dynamical systems that are not dynamic. For our purposes, we say a dynamical system is stationary if the ODEs of that system are identically zero, i.e. the ODEs describe a system whose solutions are stuck in their initial state.

Now let X and Y be two topological spaces. We say that a function $f : \mathsf{X} \to \mathsf{Y}$ is a *homeomorphism* if (i) f is bijective, (ii) f is continuous, and (iii) f has a continuous inverse. Furthermore, two flows $\Phi : \mathsf{X} \times \mathsf{R} \to \mathsf{X}$ and $\Psi : \mathsf{Y} \times \mathsf{R} \to \mathsf{Y}$ are *homeomorphic* if there exists a homeomorphism $g : \mathsf{X} \to \mathsf{Y}$ such that for each $\mathbf{x} \in \mathsf{X}$ and $t \in \mathsf{R}$ we have $g(\Phi(\mathbf{x}, t)) = \Psi(g(\mathbf{x}), t)$. If additionally g is C^1 and has a C^1 inverse, then we say g is a *diffeomorphism* and that the flows Φ and Ψ are *diffeomorphic*. Note that every diffeomorphism is also a homeomorphism, and thus every pair of diffeomorphic flows are also homeomorphic. Homeomorphisms (resp. diffeomorphisms) are a strong, and typical, notion of equivalence between dynamical systems. In essence, two dynamical systems are homeomorphic if their trajectories can be mapped to one another by stretching and bending space.

2.4. Ecological Population Models

Throughout this paper we make use of tools developed in mathematical ecology for studying the growth and decline of populations of species. As is typically done in ecological models, consider vectors $\mathbf{x} \in \mathbb{R}^n_+$ where n is the number of "species" and x_i represents the population of the i^{th} space. Suppose that the dynamics of each population is given by the system of ODEs

$$\dot{x}_i = \frac{dx_i}{dt} = x_i M_i(\mathbf{x}), \qquad i \in [n]\ . \tag{5}$$

We call any dynamical systems given by eq. 5 a *population system*. Furthermore, for each $i \in [n]$, M_i is called the i^{th} species' *fitness function*.

Two well studied special cases of population systems will be particularly relevant to our analysis: (i) when the fitness functions are affine and (ii) when the fitness functions are multivariate generalized polynomials. In case (i)–when the fitness function M_i is affine for every $i \in [n]$–the system of ODEs is known as the *Lotka-Volterra* (LV) equations and is given by the system of ODEs

$$\dot{x}_i = \frac{dx_i}{dt} = x_i \left(\hat{\lambda}_i + \sum_{j \in [n]} \hat{A}_{ij} x_j \right), \qquad i \in [n] \tag{6}$$

where $\hat{\lambda} \in \mathbb{R}^n$ and $\hat{A} \in \mathbb{R}^{n \times n}$. In case (ii)–when the fitness function M_i is a multivariate *generalized polynomial* for every $i \in [n]$–the system of ODEs is known as the *generalized Lotka-Volterra* (GLV) equations and is given by the system of ODEs

$$\dot{x}_i = \frac{dx_i}{dt} = x_i \left(\lambda_i + \sum_{j \in [m]} A_{ij} \prod_{k \in [n]} x_k^{B_{jk}} \right), \qquad i \in [n] \tag{7}$$

where m is some positive integer, $\lambda \in \mathbb{R}^n$, $A \in \mathbb{R}^{n \times m}$, and $B \in \mathbb{R}^{m \times n}$.

3. Main Result: Universality of Replicator Dynamics in Matrix Games

In this section we formally state and prove our main result. Specifically, we show that replicator dynamics in finite matrix games can emulate the behaviour of *any* finite dimensional C^1 dynamical system defined on a space diffeomorphic to the probability simplex. In order to state our result, we introduce a notion of approximately embedding one dynamical system into another.

Definition 1 *A flow Ψ on topological space X is (ϵ, T)*-approximately embedded *in a flow Θ on topological space Z if there exists $\mathsf{Z}' \subseteq \mathsf{Z}$ and topological space Y satisfying the following:*

(i) The diameter of Y is 1 with respect to $\|\cdot\|_\infty$, i.e. $\sup\{\|\mathbf{y} - \hat{\mathbf{y}}\|_\infty : \mathbf{y}, \hat{\mathbf{y}} \in \mathsf{Y}\} = 1$.

(ii) There exists diffeomorphisms $g : \mathsf{X} \to \mathsf{Y}$ and $f : \mathrm{relint}(\mathsf{Y}) \to \mathsf{Z}'$.

(iii) For every $\mathbf{y} \in \mathrm{relint}(\mathsf{Y})$ and $t \in [0, T]$ we have

$$\|g(\Psi^t(g^{-1}(\mathbf{y}))) - f^{-1}(\Theta^t(f(\mathbf{y})))\|_\infty < \epsilon\ .$$

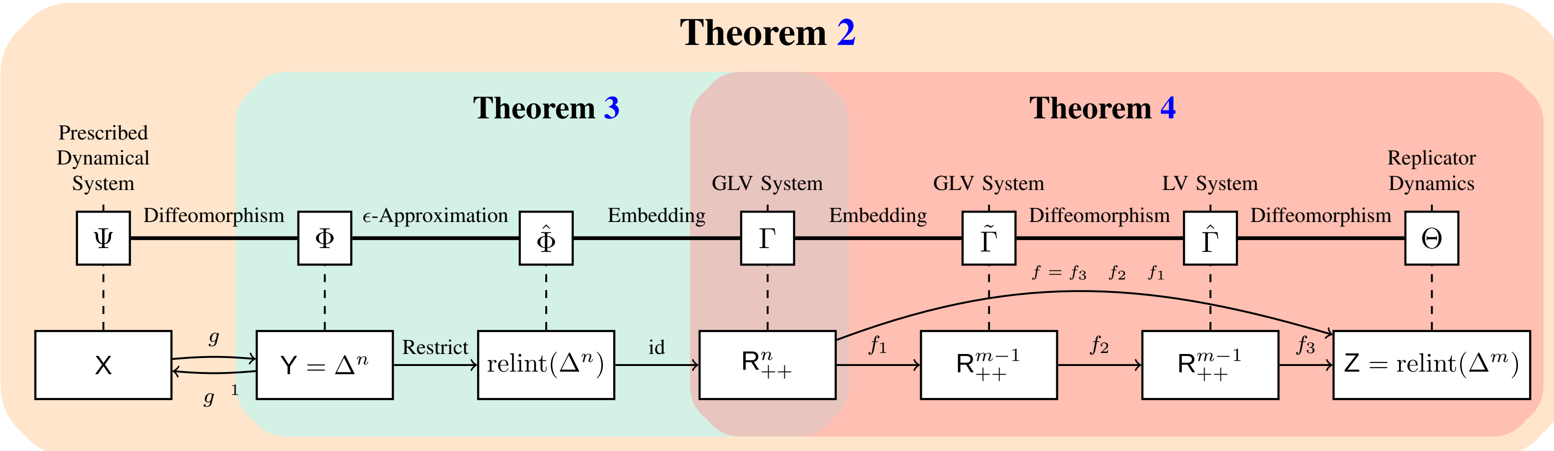

Figure 2: A diagram highlighting the relationship between Theorems 2, 3, and 4, along with the steps used to construct the (ϵ, T)-approximate embedding of Ψ in Θ. All embeddings are injective smooth maps and ensure the original dynamical system's approximation can be recovered from the higher dimensional embedding spaces. The functions f_1, f_2, and f_3 are diffeomorphisms defined in Appendix A.2. Furthermore, in Theorem 2, the subspace $\mathsf{Z}' \subseteq \mathsf{Z}$ for the (ϵ, T)-approximate embedding is $f \circ \mathrm{id}(\mathrm{relint}(\Delta^n))$.

This definition can be seen as an extension of embeddings traditionally studied in differential topology; in fact, the function f is an embedding of $\mathrm{relint}(\mathsf{Y})$ into Z in the traditional sense. Intuitively, a flow Ψ is said to be (ϵ, T)-approximately embedded in a flow Θ if, on some subspace Z', Θ stays within ϵ from a diffeomorphic copy of Ψ for at least T time. Definition 1 stipulates that the approximation is for every $\mathbf{y} \in \mathrm{relint}(\mathsf{Y})$, instead of on the entire space Y. The importance of this distinction lies in the fact that the flow Θ restricted to Z' should be diffeomorphic to a flow that is well defined in Y, but the boundary of Y may not be well defined in the embedding space.

With this definition, we can state our main result. For expository purposes we state Theorem 2 in terms of the convex hull of $n+1$ affinely independent points in R^n, since this captures most settings of interest to machine learning practitioners. Theorem 2 trivially extends to *any* space diffeomorphic to the simplex since diffeomorphisms are closed under composition.

Theorem 2 *Let X be the convex hull of a set of $n+1$ affinely independent points in R^n and Ψ be any flow on X given by a finite dimensional C^1 system of ODEs. For any $\epsilon, T > 0$, there exists $m \geq 0$ and a matrix $A \in \mathsf{R}^{m \times m}$ such that Ψ is (ϵ, T)-approximately embedded in the flow given by replicator dynamics on A.*

A proof of Theorem 2 follows immediately from Theorems 3 and 4 stated below. The basic intuition of how Theorems 2, 3, and 4 are proved and relate to one another is summarized in Figure 2. The remainder of this section is dedicated to formally proving Theorem 2. We begin by stating Theorems 3 and 4 along with their proof sketches–the full proofs are given in Appendices A.1 and A.2 respectively. We then conclude by demonstrating how these Theorems come together to prove Theorem 2. It is worth noting that in some cases our proof techniques can be used to actually construct a matrix game that emulates the behaviour of a prescribed dynamical system under RD; a concrete example is given in §4 where a matrix game giving rise to the iconic Lorenz system (Lorenz, 1963) is constructed.

Theorem 3 *Let Φ be a flow on Δ^n that is generated by a C^1 system of ODEs. For any $\epsilon, T > 0$, there exists a flow Γ on $\mathbb{R}^n_{++}$ given by a system of GLV equations (eq. 7) such that:*

(i) A subspace of $\mathrm{relint}(\Delta^n)$ is a global attracting set of Γ.

(ii) For every $\boldsymbol{y} \in \mathrm{relint}(\Delta^n)$ and $t \in [0, T]$ we have

$$\|\Phi^t(\boldsymbol{y}) - \Gamma^t(\boldsymbol{y})\|_\infty < \epsilon\,.$$

Formally constructing a flow Γ with the properties stated in Theorem 3 requires some technical legwork, but the intuition behind our construction of Γ is rather straightforward. First we get a polynomial approximation of the ODEs generating Φ from the well known Stone-Weierstrass theorem, which we call $\mathbf{p} = (p_1, \ldots, p_n)$. In our construction we ensure that $\mathbf{p}$ generates a forward invariant flow $\hat{\Phi}$ on Δ^n and has a subspace of $\mathrm{relint}(\Delta^n)$ as a global attracting set. Then, for each $i \in [n]$, we divide p_i by y_i and add the resultant generalized polynomials to the polynomial $\pi(\mathbf{y}) = (1 - \|\mathbf{y}\|_1)$, which yields a new generalized polynomial $\pi + \frac{1}{y_i}p_i$ for each $i \in [n]$. By setting these new generalized polynomials, $\pi + \frac{1}{y_i}p_i$, as the fitness functions of a population system on $\mathbb{R}^n_{++}$ we get the system generating Γ. The role of π is to define logistic equation dynamics between the ODEs so that the dynamics of the system as a whole approaches Δ^n. Since the logistic equation ensures $\|\mathbf{y}\|_1 \to 1$ as $t \to \infty$, though the dynamics outside Δ^n may be different from those on Δ^n, this construction ensures that the probability simplex Δ^n is attracting all of the dynamics. Furthermore, not only is Δ^n forward invariant under the construction, but $\pi(\mathbf{x}) = 0$ for $\mathbf{x} \in \Delta^n$ and so the flow is exactly generated by the polynomials $\mathbf{p}$ that approximate Φ. A full proof of Theorem 3 can be found in Appendix A.1.

Theorem 4 *Let $\overline{\lambda} \in \mathbb{R}^n$, $\overline{A} \in \mathbb{R}^{n \times (m-1)}$, and $\overline{B} \in \mathbb{R}^{(m-1) \times n}$ define a system of GLV equations (eq. 7) on $\mathbb{R}^n_{++}$, where $m - 1 \geq n$. Let Γ on $\mathbb{R}^n_{++}$ be the flow generated by this system of GLV equations. There exists a flow Θ on $\mathrm{relint}(\Delta^m)$ and a diffeomorphism $f : \mathbb{R}^n_{++} \to \mathsf{P} \subset \mathrm{relint}(\Delta^m)$ such that:*

(i) The flow Θ on $\mathrm{relint}(\Delta^m)$ is given by RD on a matrix game with payoff matrix $A \in \mathbb{R}^{m \times m}$.

(ii) The flow $\Theta|_{\mathsf{P}} = f(\Gamma)$ and $\Gamma = f^{-1}(\Theta|_{\mathsf{P}})$, where $\Theta|_{\mathsf{P}}$ is the flow given by Θ restricted to P.

A full proof of Theorem 4 appears in Appendix A.2. The result follows from our construction of the payoff matrix $A \in \mathbb{R}^{m \times m}$, which requires an intermediary step where the system of GLV equations on $\mathbb{R}^n_{++}$ is embedded into a system of LV equations on $\mathbb{R}^{m-1}_{++}$. This embedding is guaranteed to exist due to a trick introduced by Brenig and Goriely (1989). First, the embedding trick adds dummy dimensions to the GLV system by padding $\overline{\lambda}$, $\overline{A}$, and $\overline{B}$ to define a qualitatively equivalent system of GLV equations on $\mathbb{R}^{m-1}_{++}$–this step ensures the new GLV system is always stationary on the $m - n - 1$ newly introduced dimensions and is identical to the original system on a submanifold of $\mathbb{R}^{m-1}_{++}$. Next, the embedding trick uses a diffeomorphism to transform the enlarged GLV equations on $\mathbb{R}^{m-1}_{++}$ into a system of LV equations on $\mathbb{R}^{m-1}_{++}$. As summarized in Figure 2, the original GLV equations on $\mathbb{R}^n_{++}$ generate a flow Γ and we use the embedding trick to place it into a flow

$\hat{\Gamma}$ given by the LV equations. Using a diffeomorphism by Hofbauer and Sigmund (1998), which maps trajectories of LV equations in $\mathbb{R}^{m-1}_{++}$ to trajectories of RD in $\mathrm{relint}(\Delta^m)$, we construct the game matrix $A \in \mathbb{R}^{m \times m}$. This matrix A under RD generates the flow Θ on $\mathrm{relint}(\Delta^m)$ that we are ultimately interested in. Finally, to find the subspace $\mathsf{P} \subseteq \mathrm{relint}(\Delta^m)$ and diffeomorphism f, we note that the embedding constructed by the embedding trick is an injective smooth map from $\mathbb{R}^n_{++}$ to $\mathbb{R}^{m-1}_{++}$; define $\hat{f} : \mathbb{R}^n_{++} \to \mathrm{relint}(\Delta^m)$ as the composition of this embedding map with the diffeomorphism by Hofbauer and Sigmund (1998). The subspace P is precisely $\hat{f}(\mathbb{R}^n_{++})$ and the diffeomorphism f is obtained by restricting the range of $\hat{f}$ to P.

With Theorems 3 and 4 stated we are now ready to prove Theorem 2. Let X be a topological space with a diffeomorphism $g : \mathsf{X} \to \Delta^n$ and let Ψ be *any* flow on X given by a finite dimensional C^1 system of ODEs. Define the flow $\Phi = g(\Psi)$ on Δ^n, i.e. the dynamical system diffeomorphic to Ψ via g. From Theorem 3 we know that for any $\epsilon, T > 0$ there exists a flow Γ given by a system of GLV equations on $\mathbb{R}^n_{++}$ such that $\|\Phi^t(\mathbf{y}) - \Gamma^t(\mathbf{y})\|_\infty < \epsilon$ for every $\mathbf{y} \in \mathrm{relint}(\Delta^n)$ and $t \in [0, T]$. From Theorem 4, for $m \geq n$, we know there exists a flow Θ on $\mathrm{relint}(\Delta^m)$ and diffeomorphism $f : \mathbb{R}^n_{++} \to \mathsf{P} \subseteq \mathrm{relint}(\Delta^m)$ such that Θ restricted to P is diffeomorphic to Γ via f. Let $\Theta|_{\mathsf{P}}$ be the flow given by Θ restricted to P. Since $g(\Psi) = \Phi$, $f^{-1}(\Theta|_{\mathsf{P}}) = \Gamma$, and $f(\mathrm{relint}(\Delta^n)) \subseteq \mathsf{P} = f(\mathbb{R}^n_{++})$, it follows that $\|g(\Psi^t(g^{-1}(\mathbf{y}))) - f^{-1}(\Theta^t(f(\mathbf{y})))\|_\infty < \epsilon$ for every $\mathbf{y} \in \mathrm{relint}(\Delta^n)$ and $t \in [0, T]$. Thus, by setting $\mathsf{Z} = \mathrm{relint}(\Delta^m)$, $\mathsf{Z}' = f(\mathrm{relint}(\Delta^n))$, and $\mathsf{Y} = \Delta^n$, we have shown that Ψ is (ϵ, T)-approximately embedded in Θ. Furthermore, from Theorem 4 we know that Θ is the flow given by replicator dynamics on a matrix game with payoff matrix $A \in \mathbb{R}^{m \times m}$. The convex hull of $n+1$ affinely independent points in $\mathbb{R}^n$ is a special case of X, so we have proven Theorem 2.

4. The Lorenz Game

To demonstrate how the construction in §3 can be applied, we will highlight the construction of a matrix game whose dynamics under RD embeds the iconic system of Lorenz (1963); the full construction of this matrix game can be found in Appendix A.3. The Lorenz system's strange attractor, the "butterfly", has nearly become synonymous with chaotic flows and is given by the following three dimensional the system of ODEs in $\mathbb{R}^3$

$$
\begin{aligned}
\dot{x}_1 &= \sigma(x_2 - x_1) \\
\dot{x}_3 &= x_1(\rho - x_3) - x_2 \\
\dot{x}_3 &= x_1 x_2 - \beta x_3 \,,
\end{aligned}
$$

where $\sigma, \rho, \beta > 0$ are constants. Due to the fame of the Lorenz attractor it has been studied extensively and analyses of its dynamics under various settings of its parameters can be found in many sources (see e.g. Hateley (2019)). We will focus on the setting first studied by Lorenz, given by $\rho = 28$, $\sigma = 10$, and $\beta = 8/3$. Given these parameters, it is straightforward to show that, for sufficiently large $r > 0$, the sphere $R = \{(x_1, x_2, x_3) : x_1^2 + x_2^2 + (x_3 - \rho - \sigma)^2 = r\}$ is globally attracting and forward invariant under the Lorenz system. (Moreover, all initial conditions converge to R exponentially fast.)

Shifting the solutions of the Lorenz equation by r in the positive direction for all three dimensions, and then rearranging terms, we arrive at the following GLV system on $\mathbb{R}^3_{++}$:

$$
\begin{aligned}
\dot{x}_1 &= \sigma\left((x_2 - r) - (x_1 - r)\right) & & & \dot{x}_1 &= x_1\left(\sigma x_2 x_1^{-1} - \sigma\right) \\
\dot{x}_3 &= (x_1 - r)\left(\rho - (x_3 - r)\right) - (x_2 - r) & &\Longrightarrow & \dot{x}_2 &= x_2\left(\eta x_1 x_2^{-1} - x_1 x_3 x_2^{-1} + r x_3 x_2^{-1} + \alpha x_2^{-1} - 1\right) \\
\dot{x}_3 &= (x_1 - r)(x_2 - r) - \beta(x_1 - r) & & & \dot{x}_3 &= x_3\left(x_1 x_2 x_3^{-1} - r x_1 x_3^{-1} - r x_2 x_3^{-1} + \mu x_3^{-1} - \beta\right)
\end{aligned}
$$

where $\eta = \rho + r$, $\alpha = r - \rho r - r^2$, and $\mu = r^2 + \beta r$. Since we can rewrite the shifted Lorenz system in this GLV form, there is no need do derive the approximation highlighted in Theorem 3 and we can immediately apply Theorem 4.

From the construction used to prove Theorem 4, we get the game matrix $A \in \mathbb{R}^{11 \times 11}$ that can be written as

$$
A = \begin{bmatrix}
-\sigma & \eta & -1 & r & \alpha & 0 & 0 & 0 & 0 & (\sigma - 1) & 0 \\
\sigma & -\eta & 1 & -r & -\alpha & 0 & 0 & 0 & 0 & (1 - \sigma) & 0 \\
\sigma & -\eta & 1 & -r & -\alpha & 1 & -r & -r & \mu & (1 - \sigma - \beta) & 0 \\
0 & -\eta & 1 & -r & -\alpha & 1 & -r & -r & \mu & (1 - \beta) & 0 \\
0 & -\eta & 1 & -r & -\alpha & 0 & 0 & 0 & 0 & 1 & 0 \\
\sigma & \eta & -1 & r & \alpha & -1 & r & r & -\mu & (\beta - \sigma - 1) & 0 \\
\sigma & 0 & 0 & 0 & 0 & -1 & r & r & -\mu & (\beta - \sigma) & 0 \\
0 & \eta & -1 & r & \alpha & -1 & r & r & -\mu & (\beta - 1) & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & r & r & -\mu & \beta & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}.
$$

The solution of RD on A is plotted in Figure 1. It is worth noting that the last row and column are all zeros since they correspond to a compactifying dimension added during our construction for normalizing each dimension. Similarly, the second to last row of zeros and its corresponding column serves the role of keeping track of the constants in the shifted Lorenz system. In addition to A, we have a diffeomorphism $f : \mathbb{R}^3_{++} \to \mathbb{P} \subset \mathrm{relint}(\Delta^{11})$ from $\mathbf{x} \in \mathbb{R}^3_{++}$ to $\mathbf{p} \in \mathrm{relint}(\Delta^{11})$ that is written as

$$
f(\mathbf{x}) = \left(\frac{x_1^{-1}x_2^{1}}{N}, \frac{x_1^{1}x_2^{-1}}{N}, \frac{x_1^{1}x_2^{-1}x_3^{1}}{N}, \frac{x_2^{-1}x_3^{1}}{N}, \frac{x_2^{-1}}{N}, \frac{x_1^{1}x_2^{1}x_3^{-1}}{N}, \frac{x_1^{1}x_3^{-1}}{N}, \frac{x_2^{1}x_3^{-1}}{N}, \frac{x_3^{-1}}{N}, \frac{1}{N}, \frac{1}{N}\right),
$$

where N is a normalization factor given by the sum of the numerators in f. Since we were able to rewrite the Lorenz system in GLV form exactly, without an approximation step, RD on this game is a true embedding of the Lorenz system's strange attractor.

5. Discussion

In this paper we show that learning dynamics in finite matrix games can be as complex as any system of ODEs defined on a set diffeomorphic to the probability simplex. This result has multiple implications for both multi-agent machine learning and algorithmic game theory. We discuss some of these implications here, along with extensions and future directions.

5.1. On the Hardness of Nash Equilibria

Our results offer an interesting conclusion to a progression of results from algorithmic game theory which have established the hardness of computing Nash equilibria. First, it was shown that computing Nash equilibria is impractical or impossible in general, as it is a PPAD-hard problem (Daskalakis

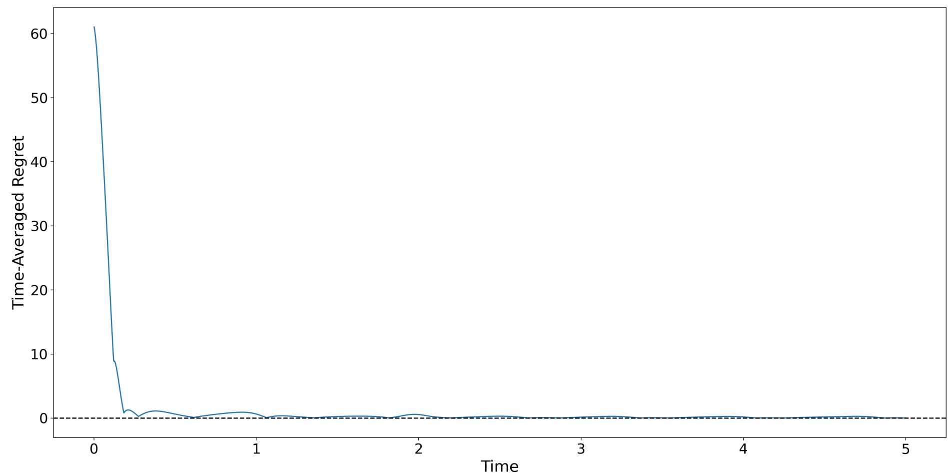

Figure 3: Time-averaged regret of the trajectory shown in Figure 1. As theoretical predictions suggest (Mertikopoulos et al., 2018), we see that regret (solid blue line) quickly converges to zero (dashed black line).

et al., 2006). Next, it was shown that learning dynamics do not converge to equilibria in general (Daskalakis et al., 2010; Sato et al., 2002; Mertikopoulos et al., 2018; Flokas et al., 2020). Recently it was revealed that, not only is convergence to equilibria not guaranteed, learning dynamics in games can even be provably chaotic (Palaiopanos et al., 2017; Chotibut et al., 2020a,b; Cheung and Piliouras, 2019, 2020). In this paper, we show that, indeed, learning dynamics can effectively simulate *any* behavior even in the special case of finite matrix games.

5.2. No-Regret Strange Attractors

A popular measure of performance for online learning algorithms is *regret*, which measures the difference between an algorithm's *average* performance against the performance of the best *fixed* strategy in hindsight. When the regret of an algorithm tends to zero as $t \to \infty$ for all sets of input, the algorithm is said to be *no-regret*. Though analyzing an algorithm's regret provides useful insights and knowing that an algorithm has no-regret is a good guarantee to have, our result shows that having a no-regret algorithm provides effectively no insight into the system's day-to-day behaviour. The arbitrary behaviour of no-regret learning algorithms in games is perhaps best exemplified by our construction of the Lorenz game in §4. Since RD is known to have no-regret in arbitrary games (e.g., Mertikopoulos et al. (2018)) and we have embedded the Lorenz system's strange attractor into RD on a matrix game, it follows that we have constructed a game where it is possible to have no-regret while the day-to-day dynamics move along a strange attractor–this is demonstrated in Figure 3. To the best of our knowledge this is the first instance where this possibility has been formally established.

5.3. Extensions and Future Directions

In this paper, we study learning in games from a different perspective than prior work; rather than focusing on the dynamics generated by a specific game setting, we focus on which games generate specific dynamics. Approaching the problem from this angle introduces a natural way of formalizing intuitions about multi-loss learning relative to the complexity of other dynamical systems. Although great strides have been made towards answering fundamental questions about learning dynamics in games, the area is still rich with challenging problems and issues. We believe that our main results, along with the novel techniques they required, provide useful insights into old questions and introduce interesting new directions. We conclude by highlighting some of these below.

Taming dynamic complexity by designing games, not just algorithms. The complex dynamics that arise from training multi-loss machine learning models, such as Generative Adversarial Networks (GANs), has recently become the object of intense study. As highlighted in §1.1, this study has led to several results reporting possible modes of failure and algorithms seeking to correct pathological training behaviours. Our main result, Theorem 2, shows that essentially any dynamics can arise in highly simplistic games and thus formalizes the idea that some dynamic complexity in multi-agent learning can be inherent to the underlying game. As the games encountered in machine learning problems are generally more complex than matrix games, our result highlights an interesting direction for future work: rather than just focusing on algorithms, we should also identify properties of games that facilitate desirable learning dynamics and tackle difficult learning problems by designing underlying games with these properties.

This idea of designing games is closely related to a research program recently proposed by Leibo et al. (2019), which treats underlying interaction structures encoded in games as an "autocurricula" to be harnessed for training more robust models. The paradigm of designing games has already led to promising results on the emergent behaviours of certain multi-agent systems (Balduzzi et al., 2020; Chang et al., 2020), but many natural approaches remain largely unexplored. For example, studying settings where the games themselves should evolve over time may help guide multi-agent learning (Skoulakis et al., 2021).

Time-average convergence, other algorithms, and relations to day-to-day dynamics. As highlighted in §5.2, the Lorenz game (§4) demonstrates the possibility of learning algorithms having no-regret while their trajectory is on a strange chaotic attractor. The existence of these no-regret strange attractors is in spite of guarantees that replicator dynamics' trajectories will converge on the set of coarse correlated equilibria in a time-average sense. A precise understanding of the relationship between an algorithm's day-to-day dynamics and its time-average convergence could provide useful insights for future work, and embedding specific dynamics into the day-to-day behaviours of replicator dynamics could be a potent tool for exploring this relationship formally. In addition, as our proof is for replicator dynamics, extending the result to other popular algorithms, including discrete-time algorithms, would be invaluable for tackling this line of inquiry more broadly.

A notion of reduction for learning dynamics. The techniques used in this paper are similar in spirit to a reduction in computational complexity theory. By mapping essentially arbitrary dynamical systems onto specific instances of replicator dynamics in games, Theorem 2 immediately shows that no positive dynamics results are possible for any class of learning dynamics that include replicator dynamics as a special case, e.g. the family of FTRL dynamics and, even more broadly, the class of all regret-minimizing dynamics. In fact, since replicator dynamics have strong no-regret guarantees, our result implies that "fast convergence" to the set of coarse correlated equilibria is insufficient for proving positive topological results about learning dynamics in games. In this way, as with traditional reductions, restricting ourselves to specific settings (e.g. replicator dynamics) can serve to strengthen the negative result. Extending these ideas and formalizing a notion of reduction for learning dynamics could serve as a framework capable of providing rigorous insights about general settings. Not only would such a framework be useful for proving negative results like ours, but it could also be used to derive strong positive guarantees. For example, FTRL dynamics on zero-sum games are known to be Hamiltonian systems (Bailey and Piliouras, 2019), which are not guaranteed to have well-behaved dynamics unless they are fully *integrable*; a reduction between *integrable* Hamiltonian systems and special cases of FTRL on zero-sum games would be a positive result of significant interest.

We hope that our work inspires further investigations in each of these directions, as we keep exploring the impressive expressive power of multi-agent learning dynamics.

Acknowledgements

This research/project is supported in part by NRF2019-NRF-ANR095 ALIAS grant, grant PIE-SGP-AI-2018-01, NRF 2018 Fellowship NRF-NRFF2018-07, AME Programmatic Fund (Grant No. A20H6b0151) from the Agency for Science, Technology and Research (A*STAR) and the National Research Foundation, Singapore under its AI Singapore Program (AISG Award No: AISG2-RP-2020-016).

References

Leonard Adolphs, Hadi Daneshmand, Aurélien Lucchi, and Thomas Hofmann. Local saddle point optimization: A curvature exploitation approach. In *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, pages 486–495, 2019. URL http://proceedings.mlr.press/v89/adolphs19a.html.

Sanjeev Arora, Elad Hazan, and Satyen Kale. The multiplicative weights update method: a meta-algorithm and applications. *Theory of Computing*, 8(1):121–164, 2012.

James P. Bailey and Georgios Piliouras. Multiplicative weights update in zero-sum games. In *ACM Conference on Economics and Computation*, 2018.

James P. Bailey and Georgios Piliouras. Multi-agent learning in network zero-sum games is a Hamiltonian system. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '19, page 233–241, Richland, SC, 2019. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9781450363099.

James P. Bailey, Gauthier Gidel, and Georgios Piliouras. Finite regret and cycles with fixed step-size via alternating gradient descent-ascent. In *Conference on Learning Theory*, pages 391–407. PMLR, 2020.

David Balduzzi, Sébastien Racanière, James Martens, Jakob N. Foerster, Karl Tuyls, and Thore Graepel. The mechanics of n-player differentiable games. In Jennifer G. Dy and Andreas

Krause, editors, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 363–372. PMLR, 2018. URL http://proceedings.mlr.press/v80/balduzzi18a.html.

David Balduzzi, Wojciech M. Czarnecki, Tom Anthony, Ian Gemp, Edward Hughes, Joel Leibo, Georgios Piliouras, and Thore Graepel. Smooth markets: A basic mechanism for organizing gradient-based learners. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=B1xMEerYvB.

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. Springer, 2019.

Leon Brenig and Alain Goriely. Universal canonical forms for time-continuous dynamical systems. *Phys. Rev. A*, 40:4119–4122, Oct 1989. doi: 10.1103/PhysRevA.40.4119. URL https://link.aps.org/doi/10.1103/PhysRevA.40.4119.

Michael Chang, Sid Kaushik, S. Matthew Weinberg, Tom Griffiths, and Sergey Levine. Decentralized reinforcement learning: Global decision-making via local economic transactions. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1437–1447. PMLR, 13–18 Jul 2020. URL http://proceedings.mlr.press/v119/chang20b.html.

Yun Kuen Cheung. Multiplicative weights updates with constant step-size in graphical constant-sum games. In *Advances in Neural Information Processing Systems*, pages 3528–3538, 2018.

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 *COLT*, 2019.

Yun Kuen Cheung and Georgios Piliouras. Chaos, extremism and optimism: Volume analysis of learning in games. In *NeurIPS*, 2020.

Thiparat Chotibut, Fryderyk Falniowski, Michał Misiurewicz, and Georgios Piliouras. The route to chaos in routing games: When is price of anarchy too optimistic? *Advances in Neural Information Processing Systems*, 33, 2020a.

Thiparat Chotibut, Fryderyk Falniowski, Michal Misiurewicz, and Georgios Piliouras. Family of chaotic maps from game theory. *Dynamical Systems*, 2020b. doi: 10.1080/14689367.2020.1795624. https://doi.org/10.1080/14689367.2020.1795624.

Charles C. Conley. *Isolated invariant sets and the Morse index*. Number 38. American Mathematical Soc., 1978.

Constantinos Daskalakis and Ioannis Panageas. The limit points of (optimistic) gradient descent in min-max optimization. In *Advances in Neural Information Processing Systems*, pages 9256–9266, 2018.

Constantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou. The complexity of computing a nash equilibrium. In *Proceedings of the Thirty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '06, page 71–78, New York, NY, USA, 2006. Association for Computing Machinery. ISBN 1595931341. doi: 10.1145/1132516.1132527. URL https://doi.org/10.1145/1132516.1132527.

Constantinos Daskalakis, Rafael Frongillo, Christos H. Papadimitriou, George Pierrakos, and Gregory Valiant. On learning algorithms for nash equilibria. In Spyros Kontogiannis, Elias Koutsoupias, and Paul G. Spirakis, editors, *Algorithmic Game Theory*, pages 114–125, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg. ISBN 978-3-642-16170-4.

Constantinos Daskalakis, Andrew Ilyas, Vasilis Syrgkanis, and Haoyang Zeng. Training GANs with optimism. In *ICLR*, 2018.

Lampros Flokas, Emmanouil-Vasileios Vlatakis-Gkaragkounis, Thanasis Lianeas, Panayotis Mertikopoulos, and Georgios Piliouras. No-regreet learning and mixed nash equilibria: They do not mix. In *NeurIPS*, 2020.

Tobias Galla and J. Doyne Farmer. Complex dynamics in learning complicated games. *Proceedings of the National Academy of Sciences*, 110(4):1232–1236, 2013.

Gauthier Gidel, Hugo Berard, Gaëtan Vignoud, Pascal Vincent, and Simon Lacoste-Julien. A variational inequality perspective on generative adversarial networks. In *ICLR*, 2019.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 2672–2680, Cambridge, MA, USA, 2014. MIT Press.

James Hateley. The lorenz system, September 2019. URL https://web.math.ucsb.edu/~jhateley/paper/lorenz.pdf.

Benito Hernández-Bermejo and Víctor Fairén. Lotka-volterra representation of general nonlinear systems. *Mathematical Biosciences*, 140(1):1 – 32, 1997. ISSN 0025-5564. doi: https://doi.org/10.1016/S0025-5564(96)00131-9. URL http://www.sciencedirect.com/science/article/pii/S0025556496001319.

Morris W. Hirsch. Systems of differential equations which are competitive or cooperative: III. competing species. *Nonlinearity*, 1(1):51–71, feb 1988. doi: 10.1088/0951-7715/1/1/003. URL https://doi.org/10.1088%2F0951-7715%2F1%2F1%2F003.

Josef Hofbauer and Karl Sigmund. *Evolutionary Games and Population Dynamics*. Cambridge University Press, 1998. doi: 10.1017/CBO9781139173179.

Ya-Ping Hsieh, Panayotis Mertikopoulos, and Volkan Cevher. The limits of min-max optimization algorithms: Convergence to spurious non-critical sets. *arXiv preprint arXiv:2006.09065*, 2020.

Robert Kleinberg, Katrina Ligett, Georgios Piliouras, and Éva Tardos. Beyond the Nash equilibrium barrier. In *Symposium on Innovations in Computer Science (ICS)*, 2011.

Joel Z. Leibo, Edward Hughes, Marc Lanctot, and Thore Graepel. Autocurricula and the emergence of innovation from social interaction: A manifesto for multi-agent intelligence research, 2019.

Alistair Letcher. On the impossibility of global convergence in multi-loss optimization, 2021.

Alistair Letcher, David Balduzzi, Sébastien Racanière, James Martens, Jakob N. Foerster, Karl Tuyls, and Thore Graepel. Differentiable game mechanics. *J. Mach. Learn. Res.*, 20:84:1–84:40, 2019. URL http://jmlr.org/papers/v20/19-008.html.

Edward N. Lorenz. Deterministic nonperiodic flow. *Journal of Atmospheric Sciences*, 20(2):130 – 141, 03 1963. doi: 10.1175/1520-0469(1963)020⟨0130:DNF⟩2.0.CO;2. URL https://journals.ametsoc.org/view/journals/atsc/20/2/1520-0469_1963_020_0130_dnf_2_0_co_2.xml.

Eric Mazumdar, Lillian J. Ratliff, and S. Shankar Sastry. On gradient-based learning in continuous games. *SIAM Journal on Mathematics of Data Science*, 2(1):103–131, 2020.

James D. Meiss. *Differential Dynamical Systems (Monographs on Mathematical Modeling and Computation)*. Society for Industrial and Applied Mathematics, USA, 2007. ISBN 0898716357.

Panayotis Mertikopoulos, Christos Papadimitriou, and Georgios Piliouras. Cycles in adversarial regularized learning. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '18, page 2703–2717, USA, 2018. Society for Industrial and Applied Mathematics. ISBN 9781611975031.

Panayotis Mertikopoulos, Bruno Lecouat, Houssam Zenati, Chuan-Sheng Foo, Vijay Chandrasekhar, and Georgios Piliouras. Optimistic mirror descent in saddle-point problems: Going the extra(-gradient) mile. In *ICLR*, 2019. URL https://openreview.net/forum?id=Bkg8jjC9KQ.

Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *International conference on machine learning*, pages 3481–3490. PMLR, 2018.

Shayegan Omidshafiei, Christos Papadimitriou, Georgios Piliouras, Karl Tuyls, Mark Rowland, Jean-Baptiste Lespiau, Wojciech M Czarnecki, Marc Lanctot, Julien Perolat, and Remi Munos. α-rank: Multi-agent evaluation by evolution. *arXiv preprint arXiv:1903.01373*, 2019.

Gerasimos Palaiopanos, Ioannis Panageas, and Georgios Piliouras. Multiplicative weights update with constant step-size in congestion games: Convergence, limit cycles and chaos. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 5874–5884, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Marco Pangallo, James Sanders, Tobias Galla, and Doyne Farmer. A taxonomy of learning dynamics in 2 x 2 games. *arXiv e-prints*, art. arXiv:1701.09043, Jan 2017.

Christos Papadimitriou and Georgios Piliouras. From nash equilibria to chain recurrent sets: An algorithmic solution concept for game theory. *Entropy*, 20(10), 2018. ISSN 1099-4300.

Christos Papadimitriou and Georgios Piliouras. Game dynamics as the meaning of a game. *ACM SIGecom Exchanges*, 16(2):53–63, 2019.

Lawrence Perko. *Differential Equations and Dynamical Systems*. Springer-Verlag, Berlin, Heidelberg, 1991. ISBN 0387974431.

Julien Perolat, Remi Munos, Jean-Baptiste Lespiau, Shayegan Omidshafiei, Mark Rowland, Pedro Ortega, Neil Burch, Thomas Anthony, David Balduzzi, Bart De Vylder, et al. From Poincaré recurrence to convergence in imperfect information games: Finding equilibrium via regularization. *arXiv preprint arXiv:2002.08456*, 2020.

Tim Roughgarden. Intrinsic robustness of the price of anarchy. *J. ACM*, 62(5), November 2015. ISSN 0004-5411. doi: 10.1145/2806883. URL https://doi.org/10.1145/2806883.

Mark Rowland, Shayegan Omidshafiei, Karl Tuyls, Julien Pérolat, Michal Valko, Georgios Piliouras, and Rémi Munos. Multiagent evaluation under incomplete information. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 12270–12282, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/510f2318f324cf07fce24c3a4b89c771-Abstract.html.

James B. T. Sanders, J. Doyne Farmer, and Tobias Galla. The prevalence of chaotic dynamics in games with many players. *Scientific reports*, 8(1):1–13, 2018.

William H. Sandholm. *Population Games and Evolutionary Dynamics*. MIT Press, 2010.

Yuzuru Sato, Eizo Akiyama, and J. Doyne Farmer. Chaos in learning a simple two-person game. *Proceedings of the National Academy of Sciences*, 99(7):4748–4751, 2002. ISSN 0027-8424. doi: 10.1073/pnas.032086299. URL https://www.pnas.org/content/99/7/4748.

Peter Schuster and Karl Sigmund. Replicator dynamics. *Journal of Theoretical Biology*, 100(3):533 – 538, 1983. ISSN 0022-5193. doi: http://dx.doi.org/10.1016/0022-5193(83)90445-9. URL http://www.sciencedirect.com/science/article/pii/0022519383904459.

Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and TrendsÂ® in Machine Learning*, 4(2):107–194, 2012. ISSN 1935-8237. doi: 10.1561/2200000018. URL http://dx.doi.org/10.1561/2200000018.

Stratis Skoulakis, Tanner Fiez, Ryann Sim, Georgios Piliouras, and Lillian J. Ratliff. Evolutionary game theory squared: Evolving agents in endogenously evolving zero-sum games. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 11343–11351. AAAI Press, 2021. URL https://ojs.aaai.org/index.php/AAAI/article/view/17352.

Stephen Smale. On the differential equations of species in competition. *Journal of Mathematical Biology*, 3:5–7, 1976.

Gilles Stoltz and Gábor Lugosi. Learning correlated equilibria in games with compact sets of strategies. *Games and Economic Behavior*, 59(1):187 – 208, 2007. ISSN 0899-8256. doi: https://doi.org/10.1016/j.geb.2006.04.007. URL http://www.sciencedirect.com/science/article/pii/S0899825606000911.

Peter D. Taylor and Leo B. Jonker. Evolutionary stable strategies and game dynamics. *Mathematical Biosciences*, 40(1):145 – 156, 1978. ISSN 0025-5564. doi: https://doi.org/10.1016/0025-5564(78)90077-9. URL http://www.sciencedirect.com/science/article/pii/0025556478900779.

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.

Jörgen W. Weibull. *Evolutionary Game Theory*. MIT Press; Cambridge, MA: Cambridge University Press., 1995.

Yasin Yazıcı, Chuan-Sheng Foo, Stefan Winkler, Kim-Hui Yap, Georgios Piliouras, and Vijay Chandrasekhar. The unusual effectiveness of averaging in gan training. In *ICLR*, 2019.

Appendix A. Proofs

A.1. Proof of Theorem 3

Theorem 2 *Let Φ be a flow on Δ^n that is generated by a C^1 system of ODEs. For any $\epsilon, T > 0$, there exists a flow Γ on $\mathbb{R}^n_{++}$ given by a system of GLV equations (eq. 7) such that:*

(i) A subspace of $\mathrm{relint}(\Delta^n)$ is a global attracting set of Γ.

(ii) For every $\mathbf{y} \in \mathrm{relint}(\Delta^n)$ and $t \in [0, T]$ we have

$$\|\Phi^t(\mathbf{y}) - \Gamma^t(\mathbf{y})\|_\infty < \epsilon\,.$$

Proof Suppose the flow Φ is given by a system of ODEs $\mathbf{h} : \Delta^n \to \mathbb{R}^n$, i.e. $\dot{y}_i = h_i(\mathbf{y})$. We will first construct a flow $\hat{\Phi}$ that approximates Φ within ϵ on Δ^n for any $T, \epsilon > 0$. Importantly, our construction ensures that $\hat{\Phi}$ is given by a system of polynomials that is well defined on Δ^n. To construct $\hat{\Phi}$ in a way where these properties are satisfied, we find a polynomial approximation of Φ and then add correction terms to the approximation that ensure the resultant polynomials are well behaved on the boundary of Δ^n. We conclude the proof by constructing the flow Γ in the Theorem statement, which has $\hat{\Phi}$ embedded on a globally attracting set of Γ. Note that the construction of the population system giving Γ is related to the construction in Smale (1976), where it is shown that some additional bookkeeping guarantees that Γ will satisfy properties commonly used in mathematical ecology for modeling species in competition.

The Stone-Weierstrass Theorem famously implies that any continuous function on a compact topological space can be approximated to an arbitrary degree of accuracy with a continuous sequence of polynomials. It follows that, by the Stone-Weierstrass Theorem, for any $\delta > 0$ and every

$i \in [n-1]$ there exists a polynomial $\hat{p}_i$ such that for every $\mathbf{y} \in \Delta^n$ we have $|h_i(\mathbf{y}) - \hat{p}_i(\mathbf{y})| < \frac{\delta}{n^2}$. Furthermore, since $\mathbf{h}$ is defined on Δ^n and therefore has the tangent space $T\Delta^n$ defined in §2.3, we know that $h_n = -\sum_{i \in [n-1]} h_i$, and setting $\hat{p}_n = -\sum_{i \in [n-1]} \hat{p}_i$ guarantees that

$$|h_n(\mathbf{y}) - \hat{p}_n(\mathbf{y})| \leq \sum_{i \in [n-1]} |\hat{p}_i(\mathbf{y}) - h_i(\mathbf{y})| < \delta/n \,.$$

It is mentioned in §2.3 that a dynamical system must "point inwards" on the boundary for it to be well defined on Δ^n. Therefore let us now consider the behaviour of $\hat{p}_i$ on the boundary of Δ^n, i.e. $\mathbf{y} \in \Delta^n$ such that some $y_i = 0$. We know that, for each $i \in [n-1]$, $h_i(\mathbf{y}) \geq 0$ for $\mathbf{y} \in \Delta^n$ such that $y_i = 0$, therefore $\hat{p}_i(\mathbf{y}) > -\delta/n^2$. Similarly, $h_n(\mathbf{y}) \geq 0$ for $\mathbf{y} \in \Delta^n$ such that $y_n = 0$, therefore $\hat{p}_n(\mathbf{y}) > -\delta/n$. It follows that to use $\hat{p}_i$ to construct a polynomial approximation on Δ^n of the flow Φ, we will need to add an appropriate correction term to each $\hat{p}_i$. Define $p_i(\mathbf{y}) = \hat{p}_i(\mathbf{y}) + \delta(\frac{1}{n} - y_i)$ for $i \in [n-1]$, and $p_n(\mathbf{y}) = -\sum_{i \in [n-1]} p_i(\mathbf{y})$ as before. Observe that we have $p_n(\mathbf{y}) = -\sum_{i \in [n-1]} (\hat{p}_i(\mathbf{y}) + \delta(\frac{1}{n} - y_i)) = \hat{p}_n - \delta\frac{n-1}{n} + \delta \sum_{i \in [n-1]} y_i = \hat{p}_n + \delta(\frac{1}{n} - y_n)$. It follows that, for $i \in [n]$ and $\mathbf{y} \in \Delta^n$, we have

$$|h_i(\mathbf{y}) - p_i(\mathbf{y})| = |h_i(\mathbf{y}) - \hat{p}_i(\mathbf{y}) - \delta(\tfrac{1}{n} - y_i)| < 2\delta \,.$$

Furthermore, for $i \in [n]$, when $\mathbf{y}$ is on the boundary of Δ^n this construction ensures that $p_i(\mathbf{y}) > 0$ when $y_i = 0$ and that $p_i(\mathbf{y}) < 0$ when $y_i = 1$. We therefore know the dynamical system given by $\mathbf{p} = (p_1, \ldots, p_n)$ is well defined on Δ^n and has a subspace of $\mathrm{relint}(\Delta^n)$ as a global attracting set.

Let $\hat{\Phi}$ be the flow given by the approximating polynomials $\mathbf{p} = (p_1, \ldots, p_n)$. By definition $\frac{d\Phi^t}{dt}(\mathbf{y}) = \mathbf{h}(\Phi^t(\mathbf{y}))$ and $\frac{d\hat{\Phi}^t}{dt}(\mathbf{y}) = \mathbf{p}(\hat{\Phi}^t(\mathbf{y}))$ for every $\mathbf{y} \in \Delta^n$ and $t \in \mathbb{R}$. Furthermore, since $\mathbf{h}$ is C^1 and Δ^n is compact, we know that $\mathbf{h}$ is Lipschitz continuous. Letting L denote the Lipschitz constant for $\mathbf{h}$ with respect to $\|\cdot\|_1$, it follows that for every $\mathbf{y} \in \Delta^n$ and $t \in \mathbb{R}$,

$$\begin{aligned}
\|\Phi^t(\mathbf{y}) - \hat{\Phi}^t(\mathbf{y})\|_1 &= \int_0^t \left\|\frac{d\Phi^s}{ds}(\mathbf{y}) - \frac{d\hat{\Phi}^s}{ds}(\mathbf{y})\right\|_1 ds \\
&= \int_0^t \|\mathbf{h}(\Phi^s(\mathbf{y})) - \mathbf{p}(\hat{\Phi}^s(\mathbf{y}))\|_1 \, ds \\
&\leq \int_0^t \|\mathbf{h}(\Phi^s(\mathbf{y})) - \mathbf{h}(\hat{\Phi}^s(\mathbf{y}))\|_1 \, ds + \int_0^t \|\mathbf{h}(\hat{\Phi}^s(\mathbf{y})) - \mathbf{p}(\hat{\Phi}^s(\mathbf{y}))\|_1 \, ds \\
&\leq \int_0^t L\|\Phi^s(\mathbf{y}) - \hat{\Phi}^s(\mathbf{y})\|_1 \, ds + 2t\delta \,.
\end{aligned}$$

Defining $R(t) = \int_0^t L\|\Phi^s(\mathbf{y}) - \hat{\Phi}^s(\mathbf{y})\|_1 \, ds + 2t\delta$, we have

$$\dot{R}(t) = L\|\hat{\Phi}^t(\mathbf{y}) - \Phi^t(\mathbf{y})\|_1 + 2\delta \leq LR(t) + 2\delta \,.$$

Letting $z(0) = R(0) = 0$ and $\dot{z} = Lz + 2\delta \geq \dot{R}$, and solving for $z(t)$, we have

$$L\|\hat{\Phi}^t(\mathbf{y}) - \Phi^t(\mathbf{y})\|_1 + 2\delta = \dot{R}(t) \leq z(t) = 2\delta e^{Lt} \,.$$

Thus, for every $t \in [0, T]$ and $\mathbf{y} \in \Delta^n$, we have

$$\|\hat{\Phi}^t(\mathbf{y}) - \Phi^t(\mathbf{y})\|_1 \leq \frac{2\delta}{L}\left(e^{Lt} - 1\right) < \epsilon \,, \tag{8}$$

where we set $\epsilon = \frac{\delta}{L}\left(e^{LT} - 1\right)$.

We will now embed $\hat{\Phi}$, restricted to $\mathrm{relint}(\Delta^n)$, inside of a flow Γ on $\mathbb{R}^n_{++}$ given by a population system with generalized polynomial fitness functions. Letting $\pi(\mathbf{y}) = (1 - \|\mathbf{y}\|_1)$, consider the population system $\mathbf{M}$ on $\mathbb{R}^n_{++}$ given by fitness functions $M_i(\mathbf{y}) = \pi(\mathbf{y}) + \frac{1}{y_i} p_i(\mathbf{y})$ for each $i \in [n]$, where p_i are the polynomials constructed above generating $\hat{\Phi}$. Note that $\mathbf{M}$ is given by the ODEs

$$\dot{y_i} = y_i \pi(\mathbf{y}) + p_i(\mathbf{y}) \ .$$

By construction, $\mathrm{relint}(()\Delta^n)$ is forward invariant under $\mathbf{M}$, as $\pi(\mathbf{y}) = 0$ on Δ^n. Furthermore, observe that for $\mathbf{y} = \mathbf{y}(t) \in \mathbb{R}^n_{++}$ the population system $\mathbf{M}$ has

$$\begin{aligned}\frac{d}{dt}\|\mathbf{y}\|_1 &= \sum_{i\in[n]} y_i \pi(\mathbf{y}) + \sum_{i\in[n]} p_i(\mathbf{y}) \\ &= \|\mathbf{y}\|_1 \pi(\mathbf{y}) \\ &= \|\mathbf{y}\|_1 (1 - \|\mathbf{y}\|_1) \ ,\end{aligned}$$

the logistic equation. Thus, for every $\mathbf{y} \in \mathbb{R}^n_{++}$, we know $\|\mathbf{y}\|_1 \to 1$ as $t \to \infty$. It follows that $\mathrm{relint}(()\Delta^n)$ is globally attracting for the dynamical system given by $\mathbf{M}$.

As a final step define Γ to be the flow on $\mathbb{R}^n_{++}$ given by $\mathbf{M}$. By our construction, we know that a subspace of $\mathrm{relint}(\Delta^n)$ is a global attracting set of Γ and that for $\mathbf{y} \in \mathrm{relint}(\Delta^n)$ we have $\Gamma = \hat{\Phi}$. All that remains to show is that Γ is given by a system of GLV equations. Recall that multivariate generalized polynomials on $\mathbf{y} \in \mathbb{R}^n$ are defined as functions of the form

$$\sum_{j\in[m]} a_j \prod_{k\in[n]} y_k^{b_k}$$

where each $a_j \in \mathbb{R}$ and $b_k \in \mathbb{R}$. It is easy to check that the set of generalized polynomials is closed under multiplication and addition. Therefore $\frac{1}{y_i} p_i(\mathbf{y})$ is a generalized polynomial, $\pi(\mathbf{y})$ is a generalized polynomial, and so $M_i(\mathbf{y}) = \pi(\mathbf{y}) + \frac{1}{y_i} p_i(\mathbf{y})$ is a generalized polynomial for each $i \in [n]$. Since the fitness functions M_i for each $i \in [n]$ is given by a generalized polynomial, the flow Γ on $\mathbb{R}^n_{++}$ is given by a system of GLV equations by definition. Furthermore, we showed that part (i) of the Theorem follows since Γ has $\mathrm{relint}(\Delta^n)$ as a global attracting set. In addition, we showed that part (ii) of the Theorem follows, as $\Gamma|_{\mathrm{relint}(\Delta^n)} = \hat{\Phi}|_{\mathrm{relint}(\Delta^n)}$. ■

A.2. Proof of Theorem 4

Theorem 3 *Let $\overline{\lambda} \in \mathbb{R}^n$, $\overline{A} \in \mathbb{R}^{n\times(m-1)}$, and $\overline{B} \in \mathbb{R}^{(m-1)\times n}$ define a system of GLV equations (eq. 7) on $\mathbb{R}^n_{++}$, where $m - 1 \geq n$. Let Γ on $\mathbb{R}^n_{++}$ be the flow generated by this system of GLV equations. There exists a flow Θ on $\mathrm{relint}(\Delta^m)$ and a diffeomorphism $f : \mathbb{R}^n_{++} \to \mathsf{P} \subset \mathrm{relint}(\Delta^m)$ such that:*

(i) The flow Θ on $\mathrm{relint}(\Delta^m)$ is given by RD on a matrix game with payoff matrix $A \in \mathbb{R}^{m\times m}$.

(ii) The flow $\Theta|_{\mathsf{P}} = f(\Gamma)$ and $\Gamma = f^{-1}(\Theta|_{\mathsf{P}})$, where $\Theta|_{\mathsf{P}}$ is the flow given by Θ restricted to P.

Proof Our proof proceeds by first embedding the GLV equations generating Γ into a system of LV equations, and then constructing a diffeomorphism from the system of LV equations to a replicator system on a matrix game with payoff matrix $A \in \mathbb{R}^{m \times m}$. The embedding from the GLV equations into the LV equations ensures that the original system is easy to recover. Our result follows immediately by composing the transformations from the given GLV equations all the way to the replicator system. The first part of our proof uses an embedding trick introduced by Brenig and Goriely (1989), whose properties (e.g. smoothness) are explored by Hernández-Bermejo and Fairén (1997). The second part of our proof follows Theorem 7.5.1 by Hofbauer and Sigmund (1998).

Consider the system of GLV equations on $\mathbb{R}^n_{++}$ generating Γ

$$\dot{x}_i = x_i \left(\overline{\lambda}_i + \sum_{j \in [m-1]} \overline{A}_{ij} \prod_{k \in [n]} x_k^{\overline{B}_{jk}} \right), \qquad i \in [n] \tag{9}$$

where $\overline{\lambda} \in \mathbb{R}^n$, $\overline{A} \in \mathbb{R}^{n \times (m-1)}$, $\overline{B} \in \mathbb{R}^{(m-1) \times n}$, and $m - 1 \geq n$. Throughout this proof we will assume without loss of generality that $\overline{\lambda} = \mathbf{0}$, as we can simply append a column to $\overline{A}$ and add a row of zeros to $\overline{B}$. In addition, we will also assume without loss of generality that $\overline{B}$ has column rank of n.[1] We will embed the system given by eq. 9 into a higher dimensional system of GLV equations by constructing matrices $\tilde{A}, \tilde{B} \in \mathbb{R}^{(m-1) \times (m-1)}$ as follows:

(i) The matrix $\tilde{A}$ has its first n rows identical to $\overline{A}$ and its last $m - n - 1$ rows as all zeros. That is, the matrix $\tilde{A} = \{\tilde{A}_{ij}\}_{i,j \in [m-1]}$ has $\tilde{A}_{ij} = \overline{A}_{ij}$ for $1 \leq i \leq n, j \in [m-1]$ and has $\tilde{A}_{ij} = 0$ for $n < i \leq m-1, j \in [m-1]$.

(ii) The matrix $\tilde{B}$ has its first n columns identical to $\overline{B}$ and its last $m - n - 1$ columns set to any values which ensure $\tilde{B}$ is non-singular. That is, the matrix $\tilde{B} = \{\tilde{B}_{ij}\}_{i,j \in [m-1]}$ has $\tilde{B}_{ij} = \overline{B}_{ij}$ for $i \in [m-1], 1 \leq j \leq n$ and, for $i \in [m-1]$, $n < j \leq m-1$, has $\tilde{B}_{ij}$ set to any value ensuring the columns are linearly independent.

These matrices define the GLV system on $\mathbb{R}^{m-1}_{++}$ given by

$$\dot{y}_i = y_i \left(\sum_{j \in [m-1]} \tilde{A}_{ij} \prod_{k \in [m-1]} y_k^{\tilde{B}_{jk}} \right), \qquad i \in [m-1] \, . \tag{10}$$

Observe that by construction $\dot{y}_i = 0$ for $i > n$ since $\tilde{A}_{ij} = 0$ for $j \in [m-1]$, therefore we know that the dynamics of the $m - n - 1$ newly introduced species are stationary. Furthermore, since the newly introduced species have stationary dynamics, this construction ensures that the ODEs associated with the the original n species only change by multiplying certain monomials with a constant–where for every species the multiplicative constant being introduced to the j^{th} monomial is fully defined by the initial conditions of the newly introduced species and is given by the term $\prod_{k>n} y_k^{\tilde{B}_{jk}}$. It follows that if we assign the initial condition $y_i = 1$ for $n < i \leq m-1$, then the ODEs $\dot{y}_i \equiv \dot{x}_i$ for $i \in [n]$. As a consequence, this construction gives a natural embedding

1. This assumption is without loss of generality because we can always add rows to $\overline{B}$ (i.e. "increase m") to ensure it has rank n. It is important that we add a column of 0's to A for every new row added to $\overline{B}$ and that any newly introduced species have unit valued initial conditions (i.e. population of one at $t = 0$). We use the same trick when embedding eq. 9 into the system given by eq. 10. Details about why this works are discussed below.

of the system given by eq. 9 into the system given by eq. 10 while ensuring the dynamics of the first n species remain identical. We can formally write this embedding as the injective smooth map $\hat{f}_1 : \mathbb{R}^n_{++} \to \mathbb{R}^{m-1}_{++}$, where $\hat{f}_1(\mathbf{x}) = (x_1, \cdots, x_n, 1, \cdots, 1)$ ensures $\dot{y}_i \equiv \dot{x}_i$ for $i \in [n]$ and $\mathbf{x} \in \mathbb{R}^n_{++}$. It is worth noting that $\hat{f}_1$ is a diffeomorphism onto its image,[2] i.e. it defines a diffeomorphism $f_1 : \mathbb{R}^n_{++} \to \hat{f}_1(\mathbb{R}^{m-1}_{++})$.

Now, for $i \in [m-1]$, transform each y_i in eq. 10 by

$$y_i = \prod_{k \in [m-1]} z_k^{C_{ik}}, \qquad i \in [m-1] \tag{11}$$

where $C \in \mathbb{R}^{(m-1)\times(m-1)}$ is some non-singular matrix. It was shown by Brenig and Goriely (1989) that transformations given by eq. 11 define diffeomorphisms from $\mathbb{R}^{m-1}_{++}$ to itself and that GLV equations are closed under these transformations. In fact, this transformation maps eq. 10 to another system of GLV equations on $\mathbb{R}^{m-1}_{++}$ given by

$$\dot{z}_i = z_i \left(\sum_{j \in [m-1]} \hat{A}_{ij} \prod_{k \in [m-1]} z_k^{\hat{B}_{jk}} \right), \qquad i \in [m-1] \tag{12}$$

where $\hat{A} = C^{-1} \cdot \tilde{A}$ and $\hat{B} = \tilde{B} \cdot C$. In particular, by using $C = \tilde{B}^{-1}$, the transformation given by eq. 11 makes $\hat{B} = I$ (the identity matrix). Therefore, by using $C = \tilde{B}^{-1}$, each generalized monomial in eq. 12 reduces to a single variable and we have the system of LV equations

$$\dot{z}_i = z_i \left(\sum_{j \in [m-1]} \hat{A}_{ij} z_j \right), \qquad i \in [m-1] \,. \tag{13}$$

Furthermore, by eq. 11, we have a diffeomorphism from $\mathbf{y} \in \mathbb{R}^{m-1}_{++}$ to $\mathbf{z} \in \mathbb{R}^{m-1}_{++}$ given by the transformations $z_i = \prod_{k \in [m-1]} y_k^{\tilde{B}_{ik}}$ for each $i \in [m-1]$. Let $f_2 : \mathbb{R}^{m-1}_{++} \to \mathbb{R}^{m-1}_{++}$ be this diffeomorphism from the GLV system given by eq. 10 to the LV system given by eq. 13. By composing $\hat{f}_1$ with f_2 we have defined an embedding of our original GLV system to the LV system given by eq. 13. In addition, since $\hat{f}_1$ ensures $y_i = 1$ for every $n < i \leq m-1$ in eq. 10, we find that the embedding into the LV system can be written as $z_i = \prod_{k \in [n]} y_k^{\tilde{B}_{ik}} = \prod_{k \in [n]} x_k^{\overline{B}_{ik}}$ for each $i \in [m-1]$.

To conclude our construction, let $\mathbf{p} \in \mathrm{relint}(\Delta^m)$ be the mixed strategy of an agent playing an m-dimensional matrix game. Furthermore, to make the notation of our argument easier to follow, add a homogenous compactifying dimension $z_m \equiv 1$ to the LV system from eq. 13. That is, let $\mathbf{z} \in \mathbb{R}^m_{++}$ be a population of species where z_i is given by eq. 13 for $i \in [m-1]$ and $z_m \equiv 1$. In particular, we consider the system of LV equations given by the coefficient matrix $A \in \mathbb{R}^{m \times m}$ where A is simply the matrix $\hat{A}$ with an additional row and column of zeros (i.e. $\{A_{l,h} = \hat{A}_{l,h}\}_{l,h \in [m-1]}$, $\{A_{l,m} = 0\}_{l \in [m]}$, and $\{A_{m,h} = 0\}_{h \in [m]}$). Observe that, aside from the compactifying dimension z_m, this LV system is equivalent to the LV system given by eq. 13. Now define a map $\mathbf{z} \to \mathbf{p}$, from populations in the LV system to mixed strategies in a game, by

$$p_i = \frac{z_i}{\sum_{j \in [m]} z_j}, \qquad i \in [m] \,. \tag{14}$$

2. Readers familiar with differential topology might notice that $\hat{f}_1$ is an immersion, which implies $\hat{f}_1$ is a smooth embedding.

Similarly, define the inverse map by

$$z_i = \frac{z_i}{z_m} = \frac{p_i}{p_m}, \qquad i \in [m]\,. \tag{15}$$

By the product rule, eq. 14, and eq. 15 we have

$$\begin{aligned}
\dot{p}_i &= \frac{\dot{z}_i}{\sum_{j\in[m]} z_j} - \frac{z_i \sum_{j\in[m]} \dot{z}_j}{\left(\sum_{j\in[m]} z_j\right)^2} \\
&= p_m \left(z_i \sum_{j\in[m]} A_{ij} z_j \right) - p_m^2 z_i \left(\sum_{j\in[m]} z_j \sum_{k\in[m]} A_{jk} z_k \right) \\
&= \frac{p_i}{p_m} \sum_{j\in[m]} A_{ij} p_j - \frac{p_i}{p_m} \left(\sum_{j\in[m]} p_j \sum_{k\in[m]} A_{jk} p_k \right) \\
&= \frac{p_i}{p_m} \left(\sum_{j\in[m]} A_{ij} p_j - \sum_{j\in[m]} p_j \sum_{k\in[m]} A_{jk} p_k \right)
\end{aligned}$$

for each $i \in [m]$. By a change in velocity we can remove the term $\frac{1}{p_m}$. This yields

$$\dot{p}_i = p_i \left(\left(\sum_{j\in[m]} A_{ij} p_j \right) - \left(\sum_{j\in[m]} p_j \sum_{k\in[m]} A_{jk} p_k \right) \right), \qquad i \in [m]\,.$$

Noting that $\sum_{j\in[m]} A_{ij} p_j = (A\mathbf{p})_i$ and $\sum_{j\in[m]} p_j \sum_{k\in[m]} A_{jk} p_k = \mathbf{p}^\intercal A\mathbf{p}$, we have derived the dynamical system

$$\dot{p}_i = p_i \left((A\mathbf{p})_i - \mathbf{p}^\intercal A\mathbf{p} \right), \qquad i \in [m] \tag{16}$$

where eq. 16 is a replicator system (eq. 3) on the matrix game with payoff matrix $A \in \mathbb{R}^{m\times m}$. The converse direction, from eq. 16 to eq. 13, is derived in a similar way.[3] We conclude that eq. 14 is a diffeomorphism mapping trajectories of our LV system given by eq. 13 onto trajectories of RD on A. Since $z_m \equiv 1$, we can define the diffeomorphism $f_3 : \mathbb{R}^{m-1}_{++} \to \operatorname{relint}(\Delta^m)$ where $p_i = z_i / \left(1 + \sum_{j\in[m-1]} z_j\right)$ for $i \in [m]$ and $p_m = 1/\left(1 + \sum_{j\in[m-1]} z_j\right)$.

Taken as a whole, we have constructed an embedding from the original GLV system given by eq. 9 to the replicator system on a matrix game given by eq. 16. The embedding itself can be written as the injective smooth map $\hat{f} : \mathbb{R}^n_{++} \to \operatorname{relint}(\Delta^m)$ where $\hat{f} = \hat{f}_1 \circ f_2 \circ f_3$. Furthermore, since we know $\hat{f}_1$ is diffeomorphic onto its image, we know there exists a diffeomorphism $f : \mathbb{R}^n_{++} \to \mathsf{P} \subseteq \operatorname{relint}(\Delta^m)$ where $f = f_3 \circ f_2 \circ f_1$ and $\mathsf{P} = \hat{f}(\mathbb{R}^n_{++})$.

Let Θ be the flow generated by eq. 16 and $\Theta|_{\mathsf{P}}$ be the flow given by Θ restricted to P. Also, recall that Γ was the flow generated by eq. 9. From our derivations of f_1,f_2,and f_3, we know that $f(\Gamma) = \Theta|_{\mathsf{P}}$. Furthermore, as diffeomorphisms as invertible and have C^1 inverses, we know $f^{-1} : \mathsf{P} \to \mathbb{R}^n_{++}$ exists and that $f^{-1}(\Theta|_{\mathsf{P}}) = \Gamma$. From our derivation of eq. 16 we know Θ is a flow

3. A full derivation of the inverse direction can be found in Hofbauer and Sigmund (1998) for Theorem 7.5.1.

on $\mathrm{relint}(\Delta^m)$ that is given by RD on a matrix game with the payoff matrix $A \in \mathbb{R}^{m \times m}$ defined above. Thus we have constructed a flow Θ and diffeomorphism f satisfying properties (i) and (ii) in the Theorem, which concludes the proof.

Though not necessary for proving the Theorem, it is interesting to observe that the diffeomorphism f can be written as

$$p_i = \frac{z_i}{1 + \sum_{j \in [m-1]} z_j} = \frac{\prod_{k \in [n]} x_k^{\overline{B}_{ik}}}{1 + \sum_{j \in [m-1]} \prod_{k \in [n]} x_k^{\overline{B}_{jk}}}, \qquad i \in [m-1], \tag{17}$$

and $p_m = 1/\left(1 + \sum_{j \in [m-1]} z_j\right) = 1/\left(1 + \sum_{j \in [m-1]} \prod_{k \in [n]} x_k^{\overline{B}_{jk}}\right)$. Similarly, by composing the inverse directions of our construction, the inverse diffeomorphism f^{-1} can be written as

$$x_i = y_i = \prod_{k \in [m-1]} z_k^{\overline{B}_{ik}^{-1}} = \prod_{k \in [m-1]} \left(\frac{p_k}{p_m}\right)^{\overline{B}_{ik}^{-1}}, \qquad i \in [n]. \tag{18}$$

■

A.3. The Lorenz Game: End-to-End Construction

In §4 we highlighted a construction of a matrix game that embeds the iconic Lorenz system under RD, but many of the details were omitted to keep the ideas concise and understandable. In this Appendix we will go through the construction of this game in its entirety. To start, note that the Lorenz system's strange attractor is given by the following three dimensional the system of ODEs in $\mathbb{R}^3$

$$\begin{aligned} \dot{x}_1 &= \sigma(x_2 - x_1) \\ \dot{x}_3 &= x_1(\rho - x_3) - x_2 \\ \dot{x}_3 &= x_1 x_2 - \beta x_3 , \end{aligned}$$

where $\sigma, \rho, \beta > 0$ are constants. We will focus on the setting first studied by Lorenz himself when $\rho = 28$, $\sigma = 10$, and $\beta = 8/3$, but it is worth noting that this construction applied for any setting of these parameters. Given these parameters, it is straightforward to show that there exists a spherical region with sufficiently large (constant) radius that is forward invariant under the Lorenz equations and is globally attracting. To find such a region, define the ellipsoid region $E = \{(x_1, x_2, x_3) : \rho x_1^2 + \sigma x_2^2 + \sigma(x_3 - 2\rho)^2 \leq c, c > 0\}$ and choose $r > 0$ such that E is contained inside a region bounded by the sphere $R = \{(x_1, x_2, x_3) : x_1^2 + x_2^2 + (x_3 - \rho - \sigma)^2 = r\}$; the region R is a globally attracting and forward invariant spherical region under the Lorenz system.

By shifting the solutions of the Lorenz equation by r in the positive direction for all three dimensions, and then rearranging terms, we get a GLV system that is well defined on $\mathbb{R}^3_{++}$ and can be written as

$$\begin{aligned} \dot{x}_1 &= \sigma\left((x_2 - r) - (x_1 - r)\right) \\ \dot{x}_3 &= (x_1 - r)\left(\rho - (x_3 - r)\right) - (x_2 - r) \\ \dot{x}_3 &= (x_1 - r)(x_2 - r) - \beta(x_1 - r) \end{aligned} \quad \Longrightarrow \quad \begin{aligned} \dot{x}_1 &= x_1\left(\sigma x_2 x_1^{-1} - \sigma\right) \\ \dot{x}_2 &= x_2\left(\eta x_1 x_2^{-1} - x_1 x_3 x_2^{-1} + r x_3 x_2^{-1} + \alpha x_2^{-1} - 1\right) \\ \dot{x}_3 &= x_3\left(x_1 x_2 x_3^{-1} - r x_1 x_3^{-1} - r x_2 x_3^{-1} + \mu x_3^{-1} - \beta\right) \end{aligned}$$

where $\eta = \rho + r$, $\alpha = r \quad \rho r \quad r^2$, and $\mu = r^2 + \beta r$. Furthermore, observe that this system of GLV equations is given by the matrices $\overline{A} \in \mathbb{R}^{3 \quad 10}$ and $\overline{B} \in \mathbb{R}^{10 \quad 3}$ which look as follows

$$\overline{A} = \begin{bmatrix} \sigma & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sigma \\ 0 & \eta & 1 & r & \alpha & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & r & r & \mu & \beta \end{bmatrix}, \qquad \overline{B} = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}.$$

Since we can rewrite the shifted Lorenz system in this GLV form, there is no need do derive the approximation highlighted in Theorem 3 and we can directly apply Theorem 4.

Using the embedding trick by Brenig and Goriely (1989) that is explained in Appendix A.2, we first embed this GLV into a higher dimensional GLV system on $\mathbb{R}^{10}_{++}$ given by the matrices

$$\tilde{A} = \begin{bmatrix} \sigma & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sigma \\ 0 & \eta & 1 & r & \alpha & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & r & r & \mu & \beta \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \qquad \tilde{B} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Next we use the diffeomorphism given by eq. 11 to transform this higher dimensional GLV system into the LV system on $\mathbb{R}^{10}_{++}$ given by the matrix

$$\hat{A} = \begin{bmatrix} \sigma & \eta & 1 & r & \alpha & 0 & 0 & 0 & 0 & (\sigma \quad 1) \\ \sigma & \eta & 1 & r & \alpha & 0 & 0 & 0 & 0 & (1 \quad \sigma) \\ \sigma & \eta & 1 & r & \alpha & 1 & r & r & \mu & (1 \quad \sigma \quad \beta) \\ 0 & \eta & 1 & r & \alpha & 1 & r & r & \mu & (1 \quad \beta) \\ 0 & \eta & 1 & r & \alpha & 0 & 0 & 0 & 0 & 1 \\ \sigma & \eta & 1 & r & \alpha & 1 & r & r & \mu & (\beta \quad \sigma \quad 1) \\ \sigma & 0 & 0 & 0 & 0 & 1 & r & r & \mu & (\beta \quad \sigma) \\ 0 & \eta & 1 & r & \alpha & 1 & r & r & \mu & (\beta \quad 1) \\ 0 & 0 & 0 & 0 & 0 & 1 & r & r & \mu & \beta \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

From the embedding trick we know that each the states $\mathbf{z} \in \mathbb{R}^{10}_{++}$ of this LV system are given by $z_i = \prod_{j \in [3]} x_j^{\overline{B}_{ij}}$, where $\mathbf{x} \in \mathbb{R}^{3}_{++}$ is from the shifted Lorenz system. As such, notice that each row of $\hat{A}$ is associated with a monomial in the shifted Lorenz system.

To conclude the construction, we apply the diffeomorphism by Hofbauer and Sigmund (1998) to this LV system and get game matrix $A \in \mathbb{R}_{++}^{11 \ 11}$ that can be written as

$$A = \begin{bmatrix} \sigma & \eta & 1 & r & \alpha & 0 & 0 & 0 & 0 & (\sigma \ \ 1) & 0 \\ \sigma & \eta & 1 & r & \alpha & 0 & 0 & 0 & 0 & (1 \ \ \sigma) & 0 \\ \sigma & \eta & 1 & r & \alpha & 1 & r & r & \mu & (1 \ \ \sigma \ \ \beta) & 0 \\ 0 & \eta & 1 & r & \alpha & 1 & r & r & \mu & (1 \ \ \beta) & 0 \\ 0 & \eta & 1 & r & \alpha & 0 & 0 & 0 & 0 & 1 & 0 \\ \sigma & \eta & 1 & r & \alpha & 1 & r & r & \mu & (\beta \ \ \sigma \ \ 1) & 0 \\ \sigma & 0 & 0 & 0 & 0 & 1 & r & r & \mu & (\beta \ \ \sigma) & 0 \\ 0 & \eta & 1 & r & \alpha & 1 & r & r & \mu & (\beta \ \ 1) & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & r & r & \mu & \beta & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

The solution of RD on A is plotted in Figure 1. It is worth noting that the last row and column are all zeros since they correspond to the compactifying dimension added during the diffeomorphism by Hofbauer and Sigmund (1998). Similarly, the second to last row of zeros and the corresponding column are from the matrix $\hat{A}$ and serve the role of keeping track of the constants in the shifted Lorenz system. In addition, we have a diffeomorphism $f : \mathbb{R}_{++}^{3} \to \mathrm{P} \quad \mathrm{relint}(\Delta^{11})$ from $\mathbf{x} \in \mathbb{R}_{++}^{3}$ to $\mathbf{p} \in \mathrm{relint}(\Delta^{11})$ that is given by

$$p_i = \frac{z_i}{1 + \sum_{j \in [m \ \ 1]} z_j} = \frac{\prod_{k \in [n]} x_k^{\overline{B}_{ik}}}{1 + \sum_{j \in [m \ \ 1]} \prod_{k \in [n]} x_k^{\overline{B}_{jk}}}, \qquad i \in [10] \,, \tag{19}$$

and $p_{11} = 1/\left(1 + \sum_{j \in [m \ \ 1]} z_j\right) = 1/\left(1 + \sum_{j \in [m \ \ 1]} \prod_{k \in [n]} x_k^{\overline{B}_{jk}}\right)$. By finding that

$$\tilde{B}^{\ 1} = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 2 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

we find that the inverse diffeomorphism $f^{\ 1}$ can be written as

$$x_i = y_i = \prod_{k \in [m \ \ 1]} z_k^{\tilde{B}_{ik}^{\ 1}} = \prod_{k \in [m \ \ 1]} \left(\frac{p_k}{p_m}\right)^{\tilde{B}_{ik}^{\ 1}}, \qquad i \in [3] \,. \tag{20}$$