From Chaos to Order: Symmetry and Conservation Laws in Game Dynamics

Sai Ganesh Nagarajan [1] **David Balduzzi** [2] **Georgios Piliouras** [1]

Abstract

Games are an increasingly useful tool for training and testing learning algorithms. Recent examples include GANs, AlphaZero and the AlphaStar league. However, multi-agent learning can be extremely difficult to predict and control. Learning dynamics even in simple games can yield chaotic behavior. In this paper, we present basic *mechanism design* tools for constructing games with predictable and controllable dynamics. We show that arbitrarily large and complex network games, encoding both cooperation (team play) and competition (zero-sum interaction), exhibit conservation laws when agents use the standard regret-minimizing dynamics known as Follow-the-Regularized-Leader. These laws persist when different agents use different dynamics and encode long-range correlations between agents' behavior, even though the agents may not interact directly. Moreover, we provide sufficient conditions under which the dynamics have multiple, linearly independent, conservation laws. Increasing the number of conservation laws results in more predictable dynamics, eventually making chaotic behavior formally impossible in some cases.

1. Introduction

Games have become a powerful training mechanism used in learning how to generate photorealistic images (Goodfellow et al., 2014a), and also how to play Go, Chess and StarCraft (Tesauro, 1995; Bansal et al., 2018; Silver et al., 2017; Jaderberg et al., 2018; Vinyals et al., 2019). The underlying assumption behind this family of architectures is that competition between learning algorithms forces them to continually improve their performance. However, this assumption is not always valid in practice. Indeed, in numerous cases of multi-agent competition (or even cooperation) the resulting dynamics can be unpredictable or, even worse, formally chaotic (Sato et al., 2002; Galla & Farmer, 2013; Piliouras & Shamma, 2014; Palaiopanos et al., 2017; Chotibut et al., 2018). This raises our central problem: *How can we effectively control learning dynamics in games?*

Naturally, this is an important and well studied problem. The classic mechanism design approach to it works in two steps. Step one, design a game with "good" equilibria. Ideally a game where all its equilibria (or at least some prominent set of equilibria) satisfy the desirable properties. Step two, apply/design algorithms that provably converge to these equilibria.

Although such approaches are powerful, they exhibit serious limitations in practice. For example, Generative Adversarial Networks (GANs) (Goodfellow et al., 2014b) have been designed to follow exactly this one-two step approach. Designed as a competitive game between two networks (a generator and a discriminator), this setting has a particularly desirable equilibrium. In this equilibrium, the generator produces realistic images and the discriminator not being able to discern real and generated images classifies the images as fake or real uniformly at random. Step one is thus guaranteed. Unfortunately, step two is far from guaranteed. Standard learning dynamics, such as gradient descent ascent, do not converge to Nash equilibria even in toy zero-sum games (e.g., Matching Pennies) (Mertikopoulos et al., 2018; Bailey & Piliouras, 2018; Cheung & Piliouras, 2019). Even if we design algorithms with provable guarantees in simple zero-sum games (Daskalakis et al., 2018; Balduzzi et al., 2018; Mertikopoulos et al., 2019; Gidel et al., 2019a;b), there is no guarantee that they converge in the high dimensional network settings that we care about and even if they do they may very well end up converging to artificial fixed points (Daskalakis & Panageas, 2018; Adolphs et al., 2018; Flokas et al., 2019).

Once we move from two agents to numerous agents then whatever little structure we had to exploit goes away. For example, in networks of zero-sum games, the notion of a value of an agent no longer exists, e.g., there may exist multiple equilibria where the same agent strictly prefers one of them (Cai et al., 2016). In the case of networks of coordination

[1]Singapore University of Technology and Design, Singapore [2]Google DeepMind, London, UK. Correspondence to: Sai Ganesh Nagarajan <sai_nagarajan@mymail.sutd.edu.sg>, David Balduzzi <dbalduzzi@google.com>, Georgios Piliouras <georgios@sutd.edu.sg>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

(common utility) games multiple equilibria with widely different properties exist and the system performance is heavily dependent on its initialization (Panageas & Piliouras, 2016).

Our approach. In this paper, we present a new approach to solving games. More precisely we design a novel framework for articulating when a multi-agent learning system has succeeded in becoming coordinated. We do this by circumventing step one altogether, i.e., we do not artificially require our system to fixate. Stationarity is not a prerequisite for success. On the other hand, we do not want to go all the way to other end and accept all possible system limit behavior (e.g., unconstrained, "random" chaotic motion). We wish to draw a balance between flexibility on one hand (captured by the anarchic, local evolution of agents) and predictability, globally coherent structure on the other (what a centralized designer requires).

A perfectly balanced multi-agent system does not stand still in rigid deadlock, neither does it proceed erratically. Instead, it looks like a chess board where pieces can move around flexibly but where overall structure clearly exists (i.e., rules and restrictions on allowable moves). But if we are to follow this analogy, what are the rules of the game? Thankfully, as it turns out, some classes of games (such as zero-sum games, coordination games and networks thereof) immediately enforce behavioral rules to their agents as long as these agents apply standard online learning dynamics (e.g., gradient descent). In other words, *the rules are part of the game*. Once we know these hidden rules (and the more such rules we know, or we embed by properly designing the game), the better we can predict which configurations are possible (or even likely) and which are not (e.g., the two white bishops cannot lie on squares of the same color).

Our setting. We consider a class of network (graphical polymatrix (Kearns et al., 2001)) n-player games, that we call **network constant-sum games with charges**. This class of games is characterized by a base network constant-sum game (i.e., each edge between two agents is a constant-sum game) and an individual charge $\lambda_i \in \mathbb{R} \setminus \{0\}$ for each agent. The charge of each agent can either by positive or negative. The payoff of each agent i is equal to the product of her charge and her utility at the original network constant-sum game. If her charge is positive then she experiences a positive payoff, if it is negative then she experiences a negative cost. These games generalize both two agent constant-sum games as well as two-agent coordination games as well as other classes of games, e.g., separable constant-sum multiplayer games or polymatrix games with both coordination and zero-sum games for which equilibrium computation can be hard (Cai & Daskalakis, 2011). We assume that all agents apply (possibly different variants) of *Follow-the-Regularized-Leader (FTRL) dynamics* (Hazan et al., 2016; Mertikopoulos et al., 2018). Special cases of these type of dynamics include replicator dynamics, arguably the most well studied evolutionary dynamic (Sandholm, 2010), as well as the standard online gradient descent.

Our results. Firstly, we show that each network constant-sum game with charges that has an interior Nash equilibrium always exhibits a conservation law, even if agents use different continuous-time variant of FTRL dynamics (Theorem 1). This result generalizes recent results of (Mertikopoulos et al., 2018) which only apply to networks of constant-sum games and the results of (Nagarajan et al., 2018), which characterizes replicator dynamics restricted to the agents playing either coordination or zero-sum games along the edges of a triangle. These conservation laws constrain the dynamics of the game by encoding long-range correlations between agents, even though the agents may not interact directly. Secondly, if the whole network is *bipartite*, then symmetries between the agents yield additional, linearly independent conservation laws (Theorem 2). Thirdly, as a consequence of this phenomenon, we can prove periodicity for specific classes of zero-sum network games with two layers and arbitrarily large number of agents (Theorem 6), whereas without such constraints even two agent systems with a small number of actions can be chaotic (Sato et al., 2002; Palaiopanos et al., 2017; Chotibut et al., 2018). Thus, *cyclic behavior emerges as the ideal balance between stability and flexibility.* Finally, in the inverse direction, given any such conservation law, we provide an efficient procedure to construct a sparse network constant-sum game with charges that implements it (Theorem 7).

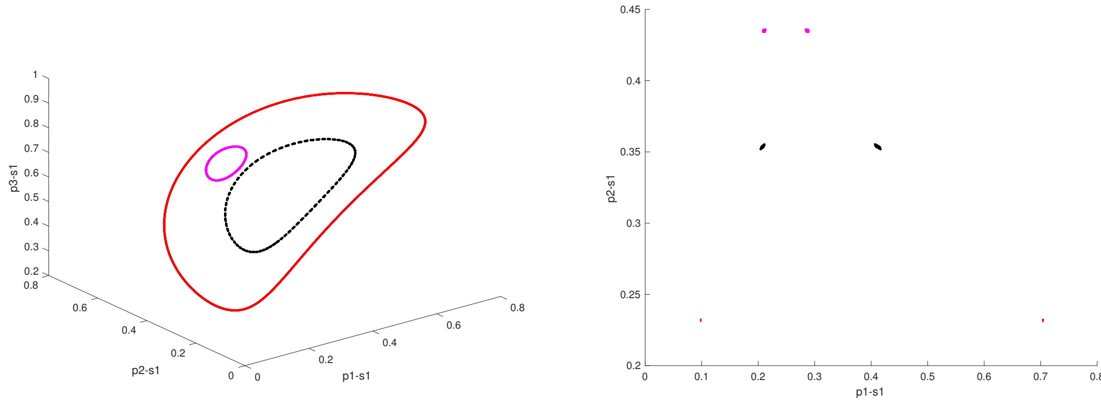
Figures 1 and 2 illustrate the difference between recurrent dynamics (e.g., FTRL in general zero-sum games (Piliouras & Shamma, 2014; Mertikopoulos et al., 2018)) and periodic dynamics. The orbits in recurrent dynamics may produce a complicated pattern of intersections given an arbitrary plane whilst periodic orbits produce much simpler patterns.

2. Preliminaries

This section provides the necessary background concepts including network generalizations of zero-sum games, the replicator dynamic and its connection to game theory and online learning. We conclude with some basic terminology and facts about dynamical systems and information theory.

2.1. Network Zero-Sum Games with Charges

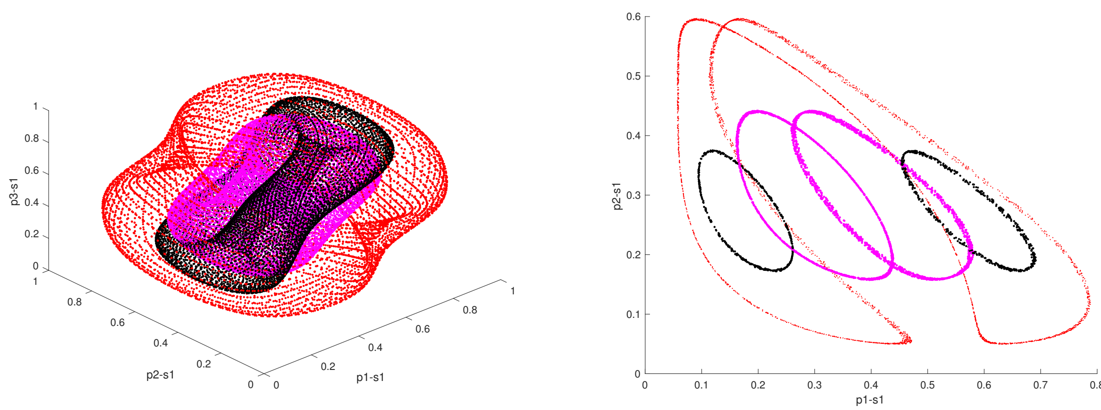
A graphical polymatrix game is defined by an undirected graph $G = (V, E)$, where V corresponds to the set of agents and where edges correspond to bimatrix games between the endpoints/agents. We denote by S_i the set of strategies of agent i. We denote the bimatrix game on edge $(i, k) \in E$ via a pair of payoff matrices: $A^{i,k}$ of dimension $|S_i| \times |S_k|$ and $A^{k,i}$ of dimension $|S_k| \times |S_i|$. Let



(a) The trajectories of agents' behavior exhibit periodic motion.

(b) The intersection points of the trajectories of agents' behavior with a plane form simple patterns.

Figure 1. Plots show the trajectories of agents updating their mixed strategies via replicator dynamics on a 4-agent complete bipartite network ($K_{2,2}$) of Matching Pennies games. Each agent in the first group wants to mismatch each agent in the second group. Each axis represents the individual agent's probability of playing the first strategy (i.e., Heads).



(a) The trajectories of agents' behavior exhibit a more complicated recurrent behavior.

(b) The intersection points of the trajectories of agents' behavior with a plane form complex patterns.

Figure 2. Plots show the trajectories of agents updating their mixed strategies via replicator dynamics on a 4-agent complete bipartite network ($K_{2,2}$) of Matching Pennies games. One agent in the first group wants to mismatch each agent in the second group, whereas the other agent in the first group wants to match each agent in the second group. Each axis represents the individual agent's probability of playing the first strategy (i.e., Heads). In this network game the agents within a group are no longer symmetric to each other.

$s \in \times_i S_i$ be a strategy profile of the game, then we denote by $s_i \in S_i$ the respective strategy of agent i. Similarly, let $s_{-i} \in \times_{j \in V \setminus i} S_j$ denote the strategies of the other agents. The payoff of agent $i \in V$ in strategy profile s is the sum of the payoffs that agent i receives from all the bimatrix games she participates in. Specifically, $u_i(s) = \sum_{(i,k) \in E} A^{i,k}_{s_i,s_k}$. A randomized strategy x_i for agent i lies on the simplex $\Delta(S_i) = \{p \in \mathbb{R}^{|S_i|}_+ : \sum_{R \in S_i} x_{iR} = 1\}$. Payoff functions are extended to randomized strategies in the usual multilinear fashion. A (mixed) Nash equilibrium is a profile of mixed strategies such that no agent can improve her payoff by unilaterally deviating to another strategy.

Definition 1. *(Cai & Daskalakis, 2011) A separable constant-sum multiplayer game GG is a graphical polymatrix game in which, for any pure strategy profile, the sum of all agent payoffs is equal to the same constant. Formally, $\forall s \in \times_i S_i, \sum_i u_i(s) = c$.*

Constant-sum games trivially have this property. All graphical games where each edge is a zero-sum game also belong in this class. These games are referred to as pairwise zero-sum polymatrix games (Daskalakis & Papadimitriou, 2009). If the edges were allowed to be arbitrary constant-sum games, the corresponding games are called pairwise constant-sum polymatrix games. There exists (Cai & Daskalakis, 2011) a (polynomial-time computable) payoff preserving transformation from every separable constant-sum multiplayer game to a pairwise constant-sum polymatrix game (i.e., a game played on a graph with agents on the nodes and two-agent games on each edge such for each $i, k \in V : A^{k,i} = c_{\{k,i\}}\mathbf{1} - \left(A^{i,k}\right)^{\mathrm{T}}$ and $\mathbf{1}$ the all-one matrix). We will assume this representation in the rest of the paper. That is, we have a network of agents and each pair of agents participates in a constant-sum game (possibly the trivial all zero game).

Our main class of games will be produced by taking linear transformations (rescalings with possible switch of the direction of axes) of separable zero-sum games of the form $\lambda_i u_i$, where $\lambda_i \in \mathbb{R} \setminus \{0\}$. That is we can think of each agent as a charged particle, where their charge, λ_i can be either positive or negative.

Definition 2. *An n-agent game G is a network constant-sum game with charges if there exists a separable constant-sum multiplayer game GG and constants $\lambda_i \in \mathbb{R} \setminus \{0\}$ for each agent i such that $u^{GG}(s) = \lambda_i u_i^G(s)$ for each outcome $s \in S$. We will also denote such game as $(\vec{\lambda})$-constant-sum multiplayer game.*

Positive rescalings of agent utilities do not affect the structure of Nash equilibrium outcomes. However, they can affect the shape of learning dynamic trajectories and thus have an effect on the properties supported by a specific system trajectory. General (negative) rescalings naturally can lead to different equilibrium sets. Specifically starting from a zero-sum game a rescaling with multiplicative weight vector $(1, -1)$ results in a two-agent coordination (common utility) game, which unlike zero-sum games always has pure Nash equilibria. In fact, it is immediate that the class of network constant-sum games with charges includes all two agent coordination (common utility) games.

Network Topologies We will be particularly interested in a special subclass of network constant-sum games with charges. Inspired by the geometry of deep layered networks, we will consider bipartite, layered polymatrix games where the set of all agents can be partitioned in sets V_i $i \in \{1, \dots, l\}$ such that agents in V_i only play games with agents in their neighboring layers V_{i-1} and V_{i+1}. Moreover, we will assume that all agents on the same level are totally *symmetric*, except possibly to effects captured by their individual charge. Specifically, given any tuple of agents $i, i' \in V_i$ and $j, j' \in V_{i+1}$ $A^{i,j} = A^{i',j} = A^{i,j'} = A^{i',j'}$. We call such games symmetric bipartite network constant-sum games with charges.

We study graphs that are commonly used to embed agent interactions through the framework of network constant-sum games with charges, such as bipartite graphs and star graphs (one center agent connected to leaf agents). When there are $K-1$ leaf agents we call this K-STAR. See Figure 1 in Section A in the supplementary material.

2.2. Follow the Regularized Leader

Follow the Regularized Leader (FTRL) is a class of learning dynamics that tries to optimize the strategies being played by tracking cumulative payoffs over time and maximizing a regularized payoff at every time instant. The regularizer makes sure that the learning is "smooth" and this leads to a variety of learning algorithms with desirable no-regret guarantees. To keep track of the payoffs of the pure strategy, we introduce a new variable $v_{iR}(x) := u_i(R, x_{-i})$ and thus $v_i(x) = (v_{iR}(x))_{R \in S_i}$ The continuous time FTRL dynamics can be specified as follows:

$$y_i(t) = y_i(0) + \int_0^t v_i(x(s))ds, \tag{FTRL}$$

$$x_i(t) = Q_i(y_i(t)), \tag{1}$$

where $Q_i : \mathbb{R}^{S_i} \mapsto \mathcal{X}_i$ is defined as

$$Q_i(y_i) = \underset{x_i \in \mathcal{X}_i}{\arg\max}\{\langle y_i, x_i\rangle - h_i(x_i)\}. \tag{2}$$

In the above cases, $\mathcal{X}_i = \Delta(S_i)$. Furthermore, $y_i(t)$ denotes the evolution of player i's payoff over time, whereas $x_i(t)$ represents the time evolution of the mixed strategy of player i and is obtained by maximizing the function $Q_i(.)$ which contains the regularization term h_i. We may assume that the regularizer $h_i(x)$ for player i satisfies the following standard assumptions:

1. h_i is continuous and strictly convex on $\mathcal{X}_i$.
2. h_i is smooth on the relative interior of every face of $\mathcal{X}_i$ (including $\mathcal{X}_i$ itself)

Another useful notion is that of the convex conjugate of $h_i(x)$, which is defined to be:

$$h_i^*(y_i) = \max_{x_i \in \mathcal{X}_i}\{\langle y_i, x_i\rangle - h_i(x_i)\}. \tag{3}$$

The above definition is useful in measuring divergences from the agents' strategies to the Nash equilibrium in the space of payoffs, and is known as the Fenchel coupling.

$$F(x^*||y) = \sum_i h(x_i^*) + h_i^*(y_i) - \langle y_i, x_i^*\rangle. \tag{4}$$

Here y is the vector of payoffs of each agent i. The FTRL framework is powerful enough to capture highly useful algorithms such as replicator dynamics (when the regularizer is the (negative) Shannon entropy) and online gradient descent, amongst others, by using the appropriate regularizers. For more details refer to (Mertikopoulos et al., 2018).

2.3. Replicator Dynamics

Besides being a special case of FTRL dynamics, the replicator equation (Taylor & Jonker, 1978; Schuster & Sigmund, 1983) is among the basic tools in mathematical ecology, genetics, and mathematical theory of selection and evolution. In its classic continuous form, it is described by the following differential equation:

$$\dot{x}_i \triangleq \frac{dx_i(t)}{dt} = x_i[u_i(x) - \hat{u}(x)], \quad \hat{u}(x) = \sum_{i=1}^{n} x_i u_i(x),$$

where x_i is the proportion of type i in the population, $x = (x_1, \ldots, x_m)$ is the vector of the distribution of types in the population, $u_i(x)$ is the fitness of type i, and $\hat{u}(x)$ is the average population fitness. The state vector x can also be interpreted as a randomized strategy of an adaptive agent that learns to optimize over its m possible actions, given an online stream of payoff vectors. As a result, it can be employed in any distributed optimization setting. An interior point of the state space is a fixed point for the replicator if and only if it is a fully mixed Nash equilibrium of the game. The interior (the boundary) of the state space $\times_i \Delta(S_i)$ are invariants for the replicator. We typically analyze the behavior of the replicator from a generic interior starting point, since points of the boundary can be captured as interior points of smaller dimensional systems. Summing all this up, our model is captured by the following system:

$$\dot{x}_{iR} = x_{iR}\big(u^i(R) - \sum_{R' \in S_i} x_{iR'} u^i(R')\big),$$

for each $i \in N$, $R \in S_i$ where we have that $u^i(R) = \mathrm{E}_{s_{-i} \sim x_{-i}} u_i(R, s_{-i})$.

The replicator dynamic enjoys numerous desirable properties such as universal consistency (no-regret)(Fudenberg & Levine, 1998; Hofbauer et al., 2009), as well as connections to several well studied discrete time learning algorithms (e.g. Multiplicative Weights algorithm (Kleinberg et al., 2009; Arora et al., 2005) or Hedge (Freund & Schapire, 1999)).

3. Dimensionality Reduction and Invariant Functions

In our first set of results we will describe an invariant function when the agents in a network constant-sum game with charges update their strategies using arbitrary FTRL dynamics (different agents may use different dynamics/regularizers).

3.1. Constant of Motion

We describe a function that is invariant to the evolution of the agents' strategies over time when playing a network constant-sum game with charges, i.e., a constant of motion. We show that the time derivative of $H(y) := \sum_{i \in V} \lambda_i \left(h_i^*(y_i) - \langle y_i, x_i^* \rangle\right)$ is zero, i.e., $H(y)$ remains invariant with the motion of the FTRL dynamics. In the above definition x^* is an interior Nash equilibrium.

Theorem 1. $H(y) := \sum_{i \in V} \lambda_i \left(h_i^*(y_i) - \langle y_i, x_i^* \rangle\right)$ *is invariant to the evolution of FTRL dynamics when agents play a network constant-sum game with charges that has an interior Nash equilibrium* x^*.

Proof. We begin by expanding $H(y)$ and taking the time derivative.

$$\begin{aligned}
\frac{dH(y)}{dt} &= \sum_{i \in V} \langle v_i(x), \lambda_i \left(\nabla h_i^*(y_i) - x_i^*\right) \rangle \\
&= \sum_{i \in V} \langle \lambda_i v_i(x), x_i - x_i^* \rangle && (5) \\
&= \sum_{i \in V} \sum_{j:(i,j) \in E} \langle \lambda_i A^{ij} x_j, x_i - x_i^* \rangle \\
&= \sum_{i,j \in E} \Big(\lambda_i x_i^T A^{ij} x_j - \lambda_i (x_i^*)^T A^{ij} x_j && (6) \\
&+ \lambda_j x_j^T A^{ji} x_i - \lambda_j (x_j^*)^T A^{ji} x_i \Big) \\
&= \sum_{i,j \in E} \left(c_{i,j} - \lambda_i (x_i^*)^T A^{ij} x_j - \lambda_j (x_j^*)^T A^{ji} x_i \right) && (7) \\
&= \sum_{i,j \in E} \left(-c_{i,j} + \lambda_i x_j^T A^{ji} x_i^* + \lambda_j x_i^T A^{ij} x_j^* \right) && (8) \\
&= \sum_{i,j \in E} \left(-c_{i,j} + \lambda_i (x_j^*)^T A^{ji} x_i^* + \lambda_j (x_i^*)^T A^{ij} x_j^* \right) && (9) \\
&= 0 && (10)
\end{aligned}$$

where 5 follows due to the "maximizing argument" identity $x_i = \nabla h_i^*(y_i)$ (See pg. 149 in (Shalev-Shwartz et al., 2012)). Lines (7),(8) follow from the fact that the λ rescaled edge games are constant sum where the constant on the edge (i, j) is $c_{i,j}$. The last line follows from the fact that the Nash equilibrium x^* is fully mixed. □

Remark 1. *The above conservation law immediately implies the Fenchel coupling (see Equation (4)), is invariant over time. Note that* $\sum_i \lambda_i h_i(x_i^*)$ *can be subsumed in the constant.*

3.2. Dimensionality Reduction & Symmetries

To describe our dimensionality reduction results for bipartite graphs and to state the main theorems, we first describe formally these games. Consider a network constant-sum game with charges with a m-by-m constant-sum base game matrix A, where the network topology is a bipartite graph with L layers and where each layer has K vertices (agents). Let the set of all vertices be V and the set of all edges be E. The agents are indexed by their vertex and the layer. For instance, the agent in vertex i and layer j is indexed as (i,j). The corresponding mixed strategies, payoff vectors and utilities are thus going to be indexed by $x^{(i,j)}$, $y^{(i,j)}$, $u_{(i,j)}$ and the vector of charges by $\vec{\lambda} = \left[\lambda_{(1,1)}, \lambda_{(2,1)}, \dots, \lambda_{(L,K)}\right]$. See Figure 1(b) in Section A for the exact structure.

We use $\mathcal{G} := \left(V, E; \vec{\lambda}, A\right)$ to represent this setting that is parameterized by charges $\vec{\lambda}$ and the base game matrix A. Then the following theorem holds:

Theorem 2. *The dynamical system induced by agents using any FTRL dynamics in any symmetric bipartite network constant-sum game with charges $\mathcal{G} = \left(V, E; \vec{\lambda}, A\right)$, lies on a low dimensional space requiring only $L(m-1)$ variables to completely describe the system.*

Proof Sketch. The idea is to uncover the symmetries in the system through an appropriate transformation. In this case, we use the techniques described in (Mertikopoulos et al., 2018) to first define variables that track the difference in accumulative payoffs w.r.t a reference strategy. Furthermore, the time derivatives of these variables can be written as a difference of two utilities. To obtain the dimensionality reduction for agents in each layer, we need to make the following observation:

For any agent (i,j), i.e., vertex i in layer j, if there is another agent in the same layer (l,j), that plays a scalar multiple of the games played by (i,j) then the time derivative of these variables for the two agents will simply be scalar multiples of each other. This allow us to find new linearly independent invariant functions. We apply this principle for each layer carefully, alternating between row and column agents to obtain the required reduction. For the full proof refer to Section B in the supplementary material. □

Remark 2. *The above theorem implies that, if wlog we were to track the $(m-1)$ strategies of the first agent, i.e., $(1,1)$, we can derive every other agent's values and the system effectively reduces down from having $K(m-1)$ to $m-1$ variables. Overall, this means that for each layer it suffices to track the first agent's $n-1$ strategies and thus making reduction from KLm to $L(m-1)$ variables. In practice, usually for such graphs, the number of layers in general is much smaller than the number of agents K.*

3.3. The Case of Replicator Dynamics

We have seen that the replicator dynamics is a special case of FTRL when the regularizer is the (negative) entropy, i.e., $h_i(x) = \sum_{R \in \mathcal{A}_i} x_{iR} \ln x_{iR}$. Using the previous theorem we get the dimensionality reduction and in addition we can obtain a closed form representation of the mixed strategies, in terms of the initial conditions and the mixed strategies of the agents we are tracking.

Lemma 3. *In the dynamical system induced by agents using replicator dynamics in any symmetric bipartite network constant-sum game with charges $\mathcal{G} = \left(V, E; \vec{\lambda}, A\right)$, the mixed strategies of agent $(i+1,j)$ has a closed form representation in terms of the mixed strategies of agent (i,j) and the initial conditions.*

$$x_k^{(i,j)} = \frac{\exp\left(\frac{-C_k^0}{\lambda_{(i,j)}}\right)\left(\frac{x_k^{(i,j)}(t)}{x_{m-1}^{(i,j)}(t)}\right)^{\frac{\lambda_{(i+1,j)}}{\lambda_{(i,j)}}}}{1 + \sum_{k=0}^{m-2} \exp\left(\frac{-C_k^0}{\lambda_{(i,j)}}\right)\left(\frac{x_k^{(i,j)}(t)}{x_{m-1}^{(i,j)}(t)}\right)^{\frac{\lambda_{(i+1,j)}}{\lambda_{(i,j)}}}}, \tag{11}$$

where C_k^0 is a constant term that depends only on the initial strategies (at time 0) and the charges.

Proof. For a fixed agent indexed by (i,j) (represents the vertex i in layer j) we can rewrite their replicator equations after a diffeomorphic change of all variables using the following variables w instead:

$$w_k^{(i,j)} = \frac{d\ln(x_k^{(i,j)})}{dt} - \frac{d\ln(x_{m-1}^{(i,j)})}{dt}, \tag{12}$$

where k denotes the action index (goes from 0 to $m-1$). Consider tracking the agent $(1,j)$, i.e., we are given $x_k^{(1,j)}(t)$ $\forall k \in 0, 1, 2, \dots, m-2$ and $\forall t \geq 0$.

Using the invariant equations derived in Theorem 2 we then have the following relationship for each k, it holds that:

$$\lambda_{(i+1,j)} w_k^{(i,j)} - \lambda_{(i,j)} w_k^{(i+1,j)} = 0 \tag{13}$$

Substituting for w, and integrating with respect to time, we get the following relations:

$$\ln\left(\frac{x_k^{(i+1,j)}(t)}{x_{m-1}^{(i+1,j)}(t)}\right) = -\frac{C_k^0}{\lambda_{(i,j)}} + \frac{\lambda_{(i+1,j)}}{\lambda_{(i,j)}} \ln\left(\frac{x_k^{(i,j)}(t)}{x_{m-1}^{(i,j)}(t)}\right)$$
$$x_k^{(i+1,j)}(t) = x_{m-1}^{(i+1,j)}(t) \exp\left(\frac{-C_k^0}{\lambda_{(i,j)}}\right) \times \left(\frac{x_k^{(i,j)}(t)}{x_{m-1}^{(i,j)}(t)}\right)^{\frac{\lambda_{(i+1,j)}}{\lambda_{(i,j)}}} \tag{14}$$

Where,

$$C_k^0 = \lambda_{(i+1,j)} \ln\left(\frac{x_k^{(i,j)}(0)}{x_{m-1}^{(i,j)}(0)}\right) - \lambda_{(i,j)} \ln\left(\frac{x_k^{(i+1,j)}(0)}{x_{m-1}^{(i+1,j)}(0)}\right).$$

Now sum of the above equations over all k should be 1, as these are mixed strategies. Using this and solving for $x_k^{(i,j)}$, we get the following relation:

$$x_k^{(i,j)} = \frac{\exp\left(\frac{-C_k^0}{\lambda_{(i,j)}}\right)\left(\frac{x_k^{(i,j)}(t)}{x_{m-1}^{(i,j)}(t)}\right)^{\frac{\lambda_{(i+1,j)}}{\lambda_{(i,j)}}}}{1+\sum_{k=0}^{m-2}\exp\left(\frac{-C_k^0}{\lambda_{(i,j)}}\right)\left(\frac{x_k^{(i,j)}(t)}{x_{m-1}^{(i,j)}(t)}\right)^{\frac{\lambda_{(i+1,j)}}{\lambda_{(i,j)}}}} \tag{15}$$

□

As the above relation depends only on all the initial conditions and the variable being tracked. Using a series of substitutions for i, starting from 1, we can obtain the full mixed strategies of all players.

4. Periodic Orbits

Here we show how the dimensionality reduction proven in the previous section can be applied to establish the emergence of periodic orbits and other useful properties about the system dynamics (such as the lack of chaos). It is important here to remind ourselves that chaotic behavior can actually emerge even in very simple settings such as replicator dynamics in Rock-Paper-Scissors and variants (Sato et al., 2002). To counter this possibility we will leveraging the following two theorems from dynamical systems literature:

Theorem 4. ***Poincaré Recurrence*** *(Poincaré, 1890; Barreira, 2006) If a flow preserves volume and has only bounded orbits then for each open set there exist orbits that intersect the set infinitely often.*

Next, we mention one of the key results in planar (2-dimensional) dynamical systems.

Theorem 5. ***Poincaré-Bendixson theorem*** *(Bendixson, 1901; Teschl, 2012) Given a differentiable real dynamical system defined on an open subset of the plane, then every non-empty compact ω-limit set of an orbit, which contains only finitely many fixed points, is either a fixed point, a periodic orbit, or a connected set composed of a finite number of fixed points together with homoclinic and heteroclinic orbits connecting these.*

Remark 3. *Theorem 5 restricts the possible limit behaviors of a planar dynamical system. Only simple limit behavior is possible (no chaos). Theorem 4 states that a neighborhood of the initial condition is visited infinitely often. When used in conjunction with Theorem 5, this will allow us to show that the limit behavior is periodic.*

Next, we will show by applying a strong enough version of our dimensionality reduction arguments we can actually understand to a large extent the topology of these multi-agent systems, despite the fact that they correspond to games with possibly arbitrarily large number of agents. When contrasting this with the possibility of chaos even in two player games (Sato et al., 2002; Palaiopanos et al., 2017; Chotibut et al., 2018), we see the power of these techniques.

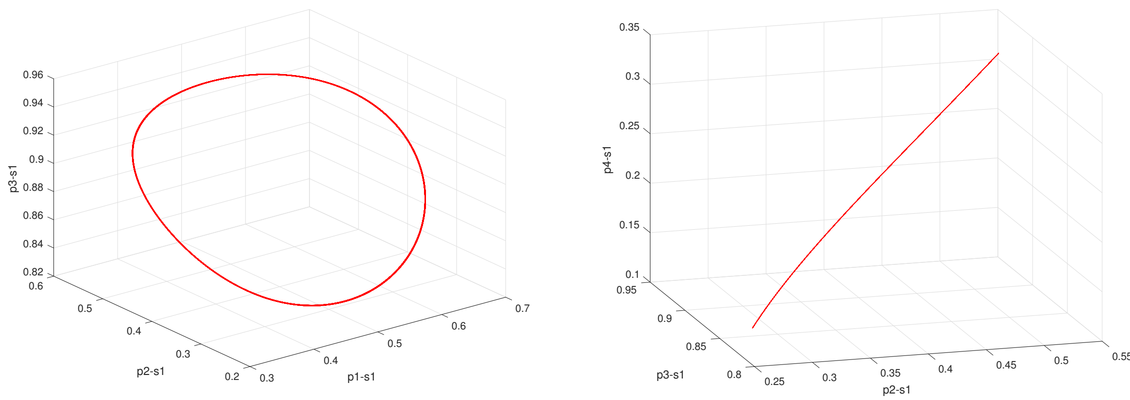
Theorem 6. *If the setting of symmetric bipartite network constant-sum games with charges $\mathcal{G} = \left(V, E; \vec{\lambda}, A\right)$ consists of two layers and each agent has two actions and the charges of all agents have the same sign then almost all orbits are periodic.*

Proof. When all agents have the same sign, then this network game belongs to the class of affine variants of network constant-sum games, for which it is known that almost all trajectories are (Poincaré) recurrent (Mertikopoulos et al., 2018). This is effectively due to an application of the Poincaré recurrence theorem. Moreover, by the invariance of the Fenchel coupling almost all trajectories stay bounded away from the set of equilibria. When we have a two layer bipartite graph with agents playing two-by-two network constant-sum game with charges, Theorem 2 guarantees that the system reduces to an autonomous two dimensional dynamical system. This allows us to apply Poincaré-Bendixon. However, the only type of recurrence behavior allowable by Poincaré-Bendixon is periodicity. Combining these two arguments the theorem follows. □

Figure 3 shows that the behavior of the agents matches with what we expect from the dimensionality reduction. The original system is 4-dimensional, but using reduction arguments we can show that system is effectively periodic. This is seen in Figure 3(a) and this is observed for any 3-dimensional projection that involves the center agent. Interestingly, the leaf agents (2,3 and 4) exhibit the behavior as shown in Figure 3(b), which indicates a form of coordination among the leaf agents as the probability of playing the same strategy (strategy 1) increases simultaneously for all the leaf agents. We observe this pattern in 3-STAR graphs as well.

5. Reverse-Engineering the Game

Our stated goal is to design systems that exhibit these conservation laws, i.e., long-range correlations. By theorem 1 we have that specific classes of games enforce a parametric family of constant of motions of the form $H(y) := \sum_{i \in V} \lambda_i \left(h_i^*(y_i) - \langle y_i, x_i^* \rangle\right)$. We will show how



(a) The trajectory of (any) two leaf agents and the center agent is shown.

(b) The trajectory of 3 leaf agents is shown.

Figure 3. Plots show the trajectories of agents updating via replicator dynamics on a 4-STAR (see Figure 1(a) in Section A of the supplementary material) network of Matching Pennies. Each axis represents the individual agent's probability of playing strategy 1. We show one projection that includes the center agent (left) and one with only the leaf agents (right).

to compute these games efficiently. For simplicity, we will assume that the target equilibrium x^* is given explicitly.

Theorem 7. *Given any conservation law of the form $H(y) := \sum_{i\in V} \lambda_i \left(h_i^*(y_i) - \langle y_i, x_i^*\rangle\right)$ we can compute in linear time a network constant-sum game with charges that implements it when each agent i uses FTRL dynamics with regularizer h_i. Moreover, the payoff matrices of the network constant-sum game with charges are sparse.*[1]

Proof. It suffices to identify a network constant-sum game with charges such that it has the desired fully mixed Nash equilibrium x^* and then have each agent i apply FTRL dynamic with regularizer $h_i(x)$.

We will do so by utilizing sparse payoff matrices A^{ik} in the construction of network constant-sum games with charges. In fact, since any constant shift to the payoff matrix of any agent does not affect the trajectories of FTRL dynamics, it suffices to consider (network) zero-sum games. We will show our analysis for the two-agent case and the multi-agent case is in the supplementary material.

Zero-sum game: The geometry of the Nash equilibrium set of zero-sum games is a classic problem (Bohnenblust et al., 1950). Let N_1, N_2 denote the set of equilibrium (*i.e.* maxmin) strategies for the row and column agent respectively. Let F_1 and F_2 be the smallest faces of simplices containing N_1 and N_2. In all zero-sum games (with finite set of strategies) we have that $\dim F_1 - \dim N_1 = \dim F_2 - \dim N_2$ (Bohnenblust et al., 1950). Since we are interested in zero-sum games of full support, we have that $|S_1| - \dim N_1 = |S_2| - \dim N_2$. To avoid trivial games (such that all zero-one), and allow for more flexibility of motion for our agents, we aim to minimize the dimension of the equilibrium strategies. We start by considering games with a unique fully mixed Nash equilibrium. Since $\dim N_1 = \dim N_2 = 0$, we derive the necessary condition that the payoff matrix is square (*i.e.* $|S_1| = |S_2|$).

Case: $|S_1| = |S_2|$: Let $t = |S_1| - 1 = |S_2| - 1$ and let's denote for notational convenience this unique equilibrium profile as $(x_0, x_1 \ldots, x_t)$, $(y_0, y_1 \ldots, y_t)$ instead of $(x_{10}, x_{11} \ldots, x_{1t})$, $(x_{20}, x_{21} \ldots, x_{2t})$. Given a value[2] $c \neq \frac{1}{t}$, the zero-sum game defined by the matrix $A(c, \vec{x}, \vec{y})$ exhibits the desired equilibrium and value (*i.e.* c) and furthermore its equilibrium is unique (Bohnenblust et al., 1950). The matrix $A(c, \vec{x}, \vec{y})$ (of dimension $t+1$=$|S_1|$=$|S_2|$) has as follows:

$$\begin{bmatrix} a(c) & \frac{c-x_1}{x_0} & \frac{c-x_2}{x_0} \cdots & \frac{c-x_t}{x_0} \\ \frac{c-y_1}{y_0} & & & \\ \frac{c-y_2}{y_0} & & I_{t\times t} & \\ \vdots & & & \\ \frac{c-y_t}{y_0} & & & \end{bmatrix}$$

[1]Naturally, all such properties can be trivially satisfied by the all-zero separable multi-agent game. This game gives rise to trivial learning dynamics, since all points of the state space are fixed points of the dynamic. So, the constant of the motions are satisfied by the fact that the system is not in motion. However, this clearly violates the spirit of what we wish to achieve, which is to give rise to dynamically evolving systems that allow both for agent coordination and flexibility. One way of excluding such trivial system is to aim for system with equilibrium sets of small dimension (e.g. unique fixed point).

[2]For $c = 1/t$ use matrix $A(2/t, \vec{x}, \vec{y})/2$.

where $a(c) = \frac{c(x_0+y_0-1)+\sum_{i-1}^{t} x_i y_i}{x_0 y_0}$ and where the submatrix $I_{t\times t}$ corresponds to the identity matrix of size t. The sparsity of the matrix is immediate. See Section C in the supplementary material for the case when $|S_1| \neq |S_2|$ and the multi-agent case. □

6. Discussion

In this section, we look at how our results can be used to understand learning in games with an algorithmic approach. Firstly, we note that conservations laws do not require the constant-sum property. For example, all coordination games, which are potential games, lie in our class of network zero-sum games with charges. Moreover, our family of games allows for potential games with arbitrarily many agents and strategies. So, even games/dynamics that have very different behavior (potential games-convergence, network zero-sum games-cycles) can both be analyzed under the same lens of conservations laws the same way a ball rolling down a hill (decreasing its potential on its way to a local minimum) and an oscillating pendulum can both be studied by the same set of conservation laws. A couple of ways to utilize these conservation laws, which might serve as follow-ups of our work, are described below:

Equilibrium selection in potential games/non-convex optimization: In the case of potential games, the invariant functions can help us compute boundaries between regions of attraction of different equilibria. For example, given an unstable fixed point, all the points both on its stable and unstable manifold (i.e., points that converge to it at +/- ∞) have to agree with the value of the invariant function. The equation $H(x) = H(\text{unstable Nash})$ is satisfied exactly by these points and these points alone. So, now we have an algebraic handle to try to compute separatrices and thus the regions of attraction of stable fixed points. Zhang & Hofbauer (2015) apply a similar idea for replicator in 2x2 coordination games. Our ideas can be applied to potential games of arbitrary size even if the agents mix-and-match using different optimization algorithms. Additionally, one might be able to identify the "optimal" regularizer in terms of equilibrium selection.

Invariant measure as a solution concept in zero-sum games: In the case of network zero-sum games, where the dynamics cycle on a closed, bounded level set of the invariant function, we now have a new way of understanding them. Instead of having the Dirac distribution at the Nash equilibrium as an invariant of the dynamics, we can define invariant measures on any level set. Furthermore, if the system is periodic then it is trivial to compute them on every trajectory. This idea is related to the notion of mixed Nash in WGANs (Hsieh et al., 2019). WGANs are infinite dimensional bi-affine games and (Hsieh et al., 2019) studies entropic mirror-descent (discretization of replicator) from the perspective of mixed strategies (i.e. measures over the parameter space to argue convergence to invariant measures). However, this approach cannot provide any characterization about what are the properties of the invariant measures, or which of the uncountably infinitely many such measures (at least one for each level set and their convex combinations) do they approximately converge to. Experimentally, however, they seem to work well. Our invariant functions may help towards a better understanding of these algorithmic techniques.

Discrete time FTRL: Although we analyze continuous time FTRL dynamics, with appropriate discretization (based on ideas coming from geometric and symplectic integration) it may be possible to create discrete versions of FTRL dynamics that preserve new invariants functions. These invariant functions would be perturbations of the invariant functions for the continuous-time dynamics. See (Bailey et al., 2019) for some early results in this direction that focus on gradient dynamics in two player zero-sum games. These results suggest that our framework may be fully extendable to discrete-time algorithms as well.

7. Conclusion

What is self-organization? We know it when we see it in familiar games like soccer, where forcing teams to compete encourages players to learn coordinated behaviors such as passing (Liu et al., 2019). In this paper, we examine a special instance of this immensely complex question. We precisely characterize how self-organization arises in simple network games. Our strategy is twofold. Firstly, we show that large classes of network games satisfy conservation laws. That is, the dynamics of the game are contained in level sets of certain invariant functions. It follows that the dynamics of the game live on a (sometimes much) lower dimensional subspace of the space of possible joint actions. Secondly, we apply the dimensionality reduction argument to show that, for symmetric games on bipartite networks, the limit behaviors of the dynamics are simple, chaotic dynamics are excluded. Understanding how far these ideas can be applied is a fascinating question that lies on the intersection of machine learning, dynamical systems and information theory and could help expand our vocabulary when it comes to dealing with complex non-equilibrating systems.

Acknowledgements

Sai Ganesh Nagarajan would like to acknowledge the SUTD President's Graduate fellowship. Georgios Piliouras acknowledges AcRF Tier 2 grant 2016-T2-1-170, grant PIE-SGP-AI-2018-01, NRF2019-NRF-ANR095 ALIAS grant and NRF 2018 Fellowship NRF-NRFF2018-07.

References

Adolphs, L., Daneshmand, H., Lucchi, A., and Hofmann, T. Local saddle point optimization: A curvature exploitation approach. *arXiv preprint arXiv:1805.05751*, 2018.

Arora, S., Hazan, E., and Kale, S. The multiplicative weights update method: a meta algorithm and applications. Technical report, 2005.

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

Bailey, J. P., Gidel, G., and Piliouras, G. Finite regret and cycles with fixed step-size via alternating gradient descent-ascent. *arXiv preprint arXiv:1907.04392*, 2019.

Balduzzi, D., Racaniere, S., Martens, J., Foerster, J., Tuyls, K., and Graepel, T. The Mechanics of n-Player Differentiable Games. In *ICML*, 2018.

Bansal, T., Pachocki, J., Sidor, S., Sutskever, I., and Mordatch, I. Emergent complexity via multi-agent competition. *ICLR*, 2018.

Barreira, L. Poincaré recurrence: old and new. In *XIVth International Congress on Mathematical Physics. World Scientific*, pp. 415–422, 2006.

Bendixson, I. Sur les courbes définies par des équations différentielles. *Acta Mathematica*, 24(1):1–88, 1901.

Bohnenblust, H. F., Karlin, S., and Shapley, L. S. Solutions of discrete, two-person games. *Contributions to the Theory of Games, Annals of Mathematics Studies*, 1:51–72, 1950.

Cai, Y. and Daskalakis, C. On minmax theorems for multiplayer games. In *ACM-SIAM Symposium on Discrete Algorithms*, SODA, pp. 217–234, 2011.

Cai, Y., Candogan, O., Daskalakis, C., and Papadimitriou, C. Zero-sum polymatrix games: A generalization of minmax. *Mathematics of Operations Research*, 41(2): 648–655, 2016.

Cheung, Y. K. and Piliouras, G. Vortices instead of equilibria in minmax optimization: Chaos and butterfly effects of online learning in zero-sum games. *COLT*, 2019.

Chotibut, T., Falniowski, F., Misiurewicz, M., and Piliouras, G. Family of chaotic maps from game theory, 2018.

Daskalakis, C. and Panageas, I. The limit points of (optimistic) gradient descent in min-max optimization. In *Advances in Neural Information Processing Systems*, pp. 9236–9246, 2018.

Daskalakis, C. and Papadimitriou, C. On a network generalization of the minmax theorem. In *ICALP*, pp. 423–434, 2009.

Daskalakis, C., Ilyas, A., Syrgkanis, V., and Zeng, H. Training GANs with optimism. In *ICLR*, 2018.

Flokas, L., Vlatakis-Gkaragkounis, E.-V., and Piliouras, G. Poincaré recurrence, cycles and spurious equilibria in gradient-descent-ascent for non-convex non-concave zero-sum games. *arXiv preprint arXiv:1910.13010*, 2019.

Freund, Y. and Schapire, R. E. Schapire: Adaptive game playing using multiplicative weights. *Games and Economic Behavior*, pp. 133, 1999.

Fudenberg, D. and Levine, D. K. *The Theory of Learning in Games*. MIT Press Books. The MIT Press, 1998.

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

Gidel, G., Berard, H., Vincent, P., and Lacoste-Julien, S. A variational inequality perspective on generative adversarial nets. *ICLR*, 2019a.

Gidel, G., Hemmat, R. A., Pezeshki, M., Huang, G., Priol, R. L., Lacoste-Julien, S., and Mitliagkas, I. Negative momentum for improved game dynamics. *AISTAS*, 2019b.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014a.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative Adversarial Nets. In *NeurIPS*, 2014b.

Hazan, E. et al. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.

Hofbauer, J., Sorin, S., and Viossat, Y. Time average replicator and best-reply dynamics. *Math. Oper. Res.*, 34 (2):263–269, May 2009. ISSN 0364-765X. doi: 10.1287/moor.1080.0359. URL http://dx.doi.org/10.1287/moor.1080.0359.

Hsieh, Y.-P., Liu, C., and Cevher, V. Finding mixed nash equilibria of generative adversarial networks. In *International Conference on Machine Learning*, pp. 2810–2819, 2019.

Jaderberg, M., Czarnecki, W. M., Dunning, I., Marris, L., Lever, G., Castaneda, A. G., Beattie, C., Rabinowitz, N. C., Morcos, A. S., Ruderman, A., Sonnerat, N., Green, T., Deason, L., Leibo, J. Z., Silver, D., Hassabis, D., Kavukcuoglu, K., and Graepel, T. Human-level performance in first-person multiplayer games with population-based deep reinforcement learning. *arXiv:1807.01281*, 2018.

Kearns, M. J., Littman, M. L., and Singh, S. P. Graphical models for game theory. In *UAI*, 2001.

Kleinberg, R., Piliouras, G., and Tardos, É. Multiplicative updates outperform generic no-regret learning in congestion games. In *ACM Symposium on Theory of Computing (STOC)*, 2009.

Liu, S., Lever, G., Merel, J., Tunyasuvunakool, S., Heess, N., and Graepel, T. Emergent Coordination Through Competition. In *ICLR*, 2019.

Mertikopoulos, P., Papadimitriou, C., and Piliouras, G. Cycles in adversarial regularized learning. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 2703–2717. SIAM, 2018.

Mertikopoulos, P., Zenati, H., Lecouat, B., Foo, C.-S., Chandrasekhar, V., and Piliouras, G. Optimistic mirror descent in saddle-point problems: Going the extra (gradient) mile. In *ICLR*, 2019.

Nagarajan, S. G., Mohamed, S., and Piliouras, G. Three body problems in evolutionary game dynamics: Convergence, periodicity and limit cycles. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 685–693. International Foundation for Autonomous Agents and Multiagent Systems, 2018.

Palaiopanos, G., Panageas, I., and Piliouras, G. Multiplicative weights update with constant step-size in congestion games: Convergence, limit cycles and chaos. In *Advances in Neural Information Processing Systems*, pp. 5872–5882, 2017.

Panageas, I. and Piliouras, G. Average case performance of replicator dynamics in potential games via computing regions of attraction. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, pp. 703–720. ACM, 2016.

Piliouras, G. and Shamma, J. S. Optimization despite chaos: Convex relaxations to complex limit sets via Poincaré recurrence. In *SODA*, 2014.

Poincaré, H. Sur le problème des trois corps et les équations de la dynamique. *Acta mathematica*, 13(1):A3–A270, 1890.

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

Sato, Y., Akiyama, E., and Farmer, J. D. Chaos in learning a simple two-person game. *Proceedings of the National Academy of Sciences*, 99(7):4748–4751, 2002. doi: 10.1073/pnas.032086299. URL `http://www.pnas.org/content/99/7/4748.abstract`.

Schuster, P. and Sigmund, K. 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`.

Shalev-Shwartz, S. et al. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2012.

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., Chen, Y., Lillicrap, T., Hui, F., Sifre, L., van den Driessche, G., Graepel, T., and Hassabis, D. Mastering the game of Go without human knowledge. *Nature*, 550:354–359, 2017.

Taylor, P. D. and Jonker, L. B. Evolutionary stable strategies and game dynamics. *Mathematical Biosciences*, 40(12):145–156, 1978. ISSN 0025-5564. doi: http://dx.doi.org/10.1016/0025-5564(78)90077-9. URL `http://www.sciencedirect.com/science/article/pii/0025556478900779`.

Tesauro, G. Temporal difference learning and TD-Gammon. *Communications of the ACM*, 38(3):58–68, 1995.

Teschl, G. *Ordinary differential equations and dynamical systems*, volume 140. American Mathematical Soc., 2012.

Vinyals, O., Babuschkin, I., Chung, J., Mathieu, M., Jaderberg, M., Czarnecki, W. M., Dudzik, A., Huang, A., Georgiev, P., Powell, R., Ewalds, T., Horgan, D., Kroiss, M., Danihelka, I., Agapiou, J., Oh, J., Dalibard, V., Choi, D., Sifre, L., Sulsky, Y., Vezhnevets, S., Molloy, J., Cai, T., Budden, D., Paine, T., Gulcehre, C., Wang, Z., Pfaff, T., Pohlen, T., Wu, Y., Yogatama, D., Cohen, J., McKinney, K., Smith, O., Schaul, T., Lillicrap, T., Apps, C., Kavukcuoglu, K., Hassabis, D., and Silver, D. AlphaStar: Mastering the Real-Time Strategy Game StarCraft II, 2019. URL `https://deepmind.com/blog/alphastar-mastering-real-time-strategy-game-starcraft-ii/`.

Zhang, B. and Hofbauer, J. Equilibrium selection via replicator dynamics in 2 x 2 coordination games. *International Journal of Game Theory*, 44(2):433–448, 2015.