Learning in Networked Interactions:
A Replicator Dynamics Approach

Daan Bloembergen[1,2], Ipek Caliskanelli[1], and Karl Tuyls[1]

[1]Department of Computer Science, University of Liverpool, UK
[2]Department of Knowledge Engineering, Maastricht University, NL
{d.bloembergen,ipek.caliskanelli,k.tuyls}@liverpool.ac.uk

Abstract. Many real-world scenarios can be modelled as multi-agent systems, where multiple autonomous decision makers interact in a single environment. The complex and dynamic nature of such interactions prevents hand-crafting solutions for all possible scenarios, hence learning is crucial. Studying the dynamics of multi-agent learning is imperative in selecting and tuning the right learning algorithm for the task at hand. So far, analysis of these dynamics has been mainly limited to normal form games, or unstructured populations. However, many multi-agent systems are highly structured, complex networks, with agents only interacting locally. Here, we study the dynamics of such networked interactions, using the well-known replicator dynamics of evolutionary game theory as a model for learning. Different learning algorithms are modelled by altering the replicator equations slightly. In particular, we investigate lenience as an enabler for cooperation. Moreover, we show how well-connected, stubborn agents can influence the learning outcome. Finally, we investigate the impact of structural network properties on the learning outcome, as well as the influence of mutation driven by exploration.

Keywords: Reinforcement Learning, Social Networks, Replicator Dynamics

1   Introduction

Understanding the dynamics of networked interactions is of vital importance to a wide range of research areas. For example, these dynamics play a central role in biological systems such as the human brain [10] or molecular interaction networks within cells [4]; in large technological systems such as the word wide web [16]; in social networks such as Facebook [2, 18, 37]; and in economic or financial institutions such as the stock market [12, 22]. Recently, researchers have focused on studying the evolution of cooperation in networks of self-interested individuals, aiming to understand how cooperative behaviour can be sustained in the face of individual selfishness [21, 26, 30, 31].

Many studies have targeted the discovery of structural properties of networks that promote cooperation. For instance, Santos and Pecheco show that cooperation has a higher chance of survival in scale-free networks [31]; Ohtsuki et al. find

a relation between the cost-benefit ratio of cooperation and the average node degree of a network that determines whether cooperation can be sustained [27]; and Van Segbroeck et al. investigate heterogeneity and clustering and concludes that these structural properties influence behaviour on the individual rather than the overall network [38]. Others have focused on the role of the particular interaction model between neighbouring nodes in determining the success of cooperation. For example, Hofmann et al. simulate various update rules in different network topologies and find that the evolution of cooperation is highly dependent on the combination of update mechanism and network topology [21].

Cooperation can also be promoted using some incentivising structure in which defection is punishable [9, 32], or in which players can choose beforehand to commit to cooperation for some given cost [19]. Both incentives increase the willingness to cooperate in such scenarios where defection would be individually rational otherwise. Allowing individuals to choose with whom to interact may similarly sustain cooperation, e.g. by giving individuals the possibility to break ties with 'bad' neighbours and replacing them with a random new connection. For example, Zimmermann and Eguíluz show how such a mechanism may promote cooperation, albeit sensitive to perturbations [42]. Similarly, Edmonds et al. use a tag-based system through which agents identify whom to interact with [17]. Allowing agents to choose which tag to adopt gives rise to social structures that can enhance cooperation. Finally, control theory is used by Bloembergen et al. to show how external influence on a subset of nodes can drive the behaviour in social networks [7].

Most of these works share one important limitation, in that they consider only imitation-based learning dynamics. Typically in such models, individual agents update their behaviour by replicating the successful behaviour of their peers. In evolution terms, the update process only incorporates *selection*. However, evolutionary success often stems from the interplay between selection on the one hand, and *mutation* on the other. Closely related is the *exploration/exploitation* dilemma that is well-known in the field of reinforcement learning, where exploration plays the role of mutation, and exploitation yield selection.

Here, we bridge these two interpretations by analysing selection-mutation dynamics as a predictive model for multi-agent reinforcement learning, where interaction between agents is modelled as a structured social network. In particular, we investigate *lenience* [6, 29] as an enabler for cooperation. We report a great difference between pure selection dynamics, and selection-mutation dynamics that include leniency. Moreover, we show how a subset of *stubborn* agents can influence the learning outcome. We find that well connected agents exert a large influence on the overall network behaviour, and as such can drive the learning process towards a desired outcome. Furthermore, we show how structural network properties, such as size and average degree, influence the learning outcome. Finally, we observe that certain network structures give rise to clusters of cooperators and defectors coexisting.

In contrast to the majority of related work, which almost exclusively focuses on Prisoner's Dilemma type interactions, we use the Stag Hunt to describe

the interaction between agents. The Stag Hunt provides an intuitive model of many real-world strategic (economic) interactions, such as the introduction of potentially beneficial new technologies that require a critical mass of adopters in order to pay off. As such, not switching (defecting) is a safe choice, whereas social cooperation (adoption) may yield higher rewards for all.

This paper proceeds as follows. Firstly, we explain required background knowledge on learning, evolutionary game theory, and networks, in Section 2. Secondly, Section 3 outlines the methodology used in this work, in particular the formal link between multi-agent learning and the replicator dynamics. We present our model of networked replicator dynamics in Section 4, accompanied by a range of experiments in Section 5. The paper is closed with main conclusion of this study in Section 6.

2 Background

This section gives an overview of relevant background needed for the remainder of this work. The section is split into three main parts. Section 2.1 briefly introduces reinforcement learning; Section 2.2 describes basic concepts of evolutionary game theory; and Section 2.3 details networks.

2.1 Reinforcement learning

The *reinforcement learning* (RL) paradigm is based on the concept of trial-and-error learning, allowing agents to optimise their behaviour without explicitly requiring a model of the environment [34]. The reinforcement learning agent continuously interacts with the environment, perceiving its state, taking actions, and observing the effect of those actions. The agent needs to balance *exploration* and *exploitation* in order to ensure good intermediate rewards while avoiding getting stuck in local optima. RL strategies are powerful techniques for optimising control of large scale control problems [15]. Early RL research focused on single-agent problems where the full state knowledge of the agent is known. Later on, RL has been applied to multi-agent domains as well [11]. The computational complexity of *multi-agent reinforcement learning* (MARL) algorithms is much higher than in single-agent problems, since (near) optimal behaviour of one agent depends on other agents' policies as well.

Despite this challenge, single-agent RL techniques have been applied successfully to multi-agent settings. Arguably the most famous example of an RL algorithm is the model-free temporal difference algorithm *Q-learning* [39]. Q-learning[1] maintains a value function over actions, Q_i, which is updated at every time step t based on the reward r received after taking action a_i:

$$Q_i(t+1) \leftarrow Q_i(t) + \alpha\big(r - Q_i(t)\big) \tag{1}$$

---

[1] We describe *stateless* Q-learning, as this version is suitable for the work presented in this paper.

$$
\begin{pmatrix} a_{11}, b_{11}\ a_{12}, b_{12} \\ a_{21}, b_{21}\ a_{22}, b_{22} \end{pmatrix} \quad
\begin{array}{cc} & C \quad\ D \\ \begin{array}{c} C \\ D \end{array} & \begin{pmatrix} A, A\ C, B \\ B, C\ D, D \end{pmatrix} \end{array} \quad
\begin{array}{cc} & C \ \ D \\ \begin{array}{c} C \\ D \end{array} & \begin{pmatrix} 4, 4\ 1, 3 \\ 3, 1\ 3, 3 \end{pmatrix} \end{array}
$$

Fig. 1. General payoff bi-matrix $(\mathbf{A}, \mathbf{B})$ for two-player two-action games (left) and the Stag Hunt (center), and a typically valued instance of the Stag Hunt (right)

where $\alpha \in [0, 1]$ is the learning rate that determines how quickly Q is updated based on new reward information. Choosing which action to take is crucial for the learning process. The *Boltzmann* exploration scheme is often used as it provides a way to balance exploration and exploitation by selecting an appropriate temperature τ. The policy $\mathbf{x}$ that determines the probability for choosing each action a is computed as

$$
x_i = \frac{e^{Q_i/\tau}}{\sum_j e^{Q_j/\tau}} \tag{2}
$$

A high temperature drives the mechanism towards exploration, whereas a low temperature promotes exploitation.

2.2   Evolutionary Game Theory

The strategic interaction between agents can be modelled in the form of a game, where each player (agent) has a set of actions, and a preference over the joint action space that is captured in the received payoffs. For two-player games, the payoffs can be represented by a bi-matrix $(\mathbf{A}, \mathbf{B})$, that gives the payoff for the row player in $\mathbf{A}$, and the column player in $\mathbf{B}$, see Figure 1 (left). The goal of each player is to decide which action to take, so as to maximise their expected payoff. Classical game theory assumes that full knowledge of the game is available to all players, which together with the assumption of individual rationality does not necessarily reflect the dynamic nature of real world interaction. *Evolutionary game theory* (EGT) relaxes the rationality assumption and replaces it by the concepts of natural selection and mutation from evolutionary biology [24, 41]. Where classical game theory describes strategies in the form of probabilities over pure actions, EGT models them as populations of individuals, each of a pure action type, where the population share of each type reflects its evolutionary success.

Central to EGT are the *replicator dynamics*, that describe how this population of individuals evolves over time under evolutionary pressure. Individuals are randomly paired to interact, and their reproductive success is determined by their fitness which results from these interactions. The replicator dynamics dictate that the population share of a certain type will increase if the individuals of this type have a higher fitness than the population average; otherwise their population share will decrease. The population can be described by the state vector $\mathbf{x} = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}$, with $0 \le x_i \le 1\ \forall i$ and $\sum_i x_i = 1$, representing the fractions of the population belonging to each of n pure types. Now suppose the fitness of type i is given by the fitness function $f_i(\mathbf{x})$, and the average fitness of

the population is given by $\bar{f}(\mathbf{x}) = \sum_j x_j f_j(\mathbf{x})$. The population change over time can then be written as

$$\dot{x}_i = x_i \left[f_i(\mathbf{x}) - \bar{f}(\mathbf{x}) \right] \tag{3}$$

In a two-player game with payoff bi-matrix $(\mathbf{A}, \mathbf{B})$, where the two players use the strategies $\mathbf{x}$ and $\mathbf{y}$ respectively, the fitness of the first player's i^{th} candidate strategy can be calculated as $\sum_j a_{ij} y_j$. Similarly, the average fitness of population $\mathbf{x}$ is defined as $\sum_i x_i \sum_j a_{ij} y_j$. In matrix form, this leads to the following multi-population replicator dynamics:

$$\begin{aligned} \dot{x}_i &= x_i \left[(\mathbf{Ay})_i - \mathbf{x}^{\mathrm{T}} \mathbf{A} \mathbf{y} \right] \\ \dot{y}_i &= y_i \left[(\mathbf{x}^{\mathrm{T}} \mathbf{B})_i - \mathbf{x}^{\mathrm{T}} \mathbf{B} \mathbf{y} \right] \end{aligned} \tag{4}$$

The Stag Hunt is a game that describes a dilemma between safety and social cooperation [33]. The canonical payoff matrix of the Stag Hunt is given in Fig. 1 (center), where $A > B \geq D > C$. Social cooperation between players is rewarded with A, given that both players choose to cooperate (action C). As the players do not foresee each others' strategies, the safe choice of players is to defect (action D), since typically $A + C < B + D$ (see Figure 1, right). Although cooperation pays off more for both players, defection is individually rational when the opponent strategy is unknown. As both players reason like this, they may end up in a state of mutual defection, receiving $D < A$ each, hence the dilemma.

The Stag Hunt is typically said to model individuals that go out on a hunt, and can only capture big game (e.g. a stag) by joining forces, whereas smaller pray (e.g. a hare) can be captured individually. However, it can also be thought of to describe the introduction of a new technology, which only really pays off when more people are using it. Early adopters risk paying the price for this. As such, despite its simplicity the Stag Hunt is an useful model for many real-world strategic dilemmas.

2.3   Networked Interactions

Networks describe collections of entities (nodes) and the relation between them (edges). Formally, a network can be represented by a graph $\mathbb{G} = (\mathcal{V}, \mathcal{W})$ consisting of a non-empty set of nodes (or vertices) $\mathcal{V} = \{v_1, \ldots, v_N\}$ and an $N \times N$ adjacency matrix $\mathcal{W} = [w_{ij}]$ where non-zero entries w_{ij} indicate the (possibly weighted) connection from v_i to v_j. If $\mathcal{W}$ is symmetrical, such that $w_{ij} = w_{ji}$ for all i, j, the graph is said to be undirected, meaning that the connection from node v_i to v_j is equal to the connection from node v_j to v_i. In social networks, for example, one might argue that friendship is usually mutual and hence undirected. This is the approach followed in this work. In general however this need not be the case, in which case the graph is said to be directed, and $\mathcal{W}$ asymmetrical. The neighbourhood, $\mathbb{N}$, of a node v_i is defined as the set of nodes it is directly connected to, i.e. $\mathbb{N}(v_i) = \cup_j v_j : w_{ij} > 0$. The node's degree $\deg[v_i]$ is given by the cardinality of its neighbourhood.

Several types of networks have been proposed that capture the structural properties found in large social, technological or biological networks, two well-known examples being the small-world and scale-free models. The *small-world* model exhibits short average path lengths between nodes and high clustering, two features often found in real-world networks [40]. Another model is the *scale-free* network, characterised by a heavy-tailed degree distribution following a power law [3]. In such networks the majority of nodes will have a small degree while simultaneously there will be relatively many nodes with very large degree, the latter being the hubs or connectors of the network. For a detailed description of networks and their properties, the interested reader is referred to [22].

3   Evolutionary models of multi-agent learning

Multi-agent learning and evolutionary game theory share a substantial part of their foundation, in that they both deal with the decision making process of bounded rational agents, or players, in uncertain environments. The link between these two fields is not only intuitive, but also formally proven that the continuous time limit of *Cross learning* converges to the replicator dynamics [8].

Cross learning [14] is one of the most basic stateless reinforcement learning algorithms, which updates its policy $\mathbf{x}$ based on the reward r received after taking action j as

$$x_i \leftarrow x_i + \begin{cases} r - x_i r & \text{if } i = j \\ -x_i r & \text{otherwise} \end{cases} \tag{5}$$

A valid policy is ensured by the update rule as long as the rewards are normalised, i.e., $0 \leq r \leq 1$. Cross learning is closely related to learning automata (LA) [25, 35]. In particular, it is equivalent to a learning automaton with a linear reward-inaction (L_{R-I}) update scheme and a learning step size of 1.

We can estimate $E[\Delta x_i]$, the expected change in the policy induced by Equation 5. Note that the probability x_i of action i is affected both if i is selected and if another action j is selected, and let $E_i[r]$ be the expected reward after taking action i. We can now write

$$E[\Delta x_i] = x_i \Big[E_i[r] - x_i E_i[r] \Big] + \sum_{j \neq i} x_j \Big[- E_j[r] x_i \Big]$$
$$= x_i \Big[E_i[r] - \textstyle\sum_j x_j E_j[r] \Big] \tag{6}$$

Assuming the learner takes infinitesimally small update steps, we can take the continuous time limit of Equation 6 and write is as the partial differential equation

$$\dot{x}_i = x_i \Big[\mathrm{E}_i[r] - \textstyle\sum_j x_j \mathrm{E}_j[r] \Big]$$

In a two-player normal form game, with payoff matrix $\mathbf{A}$ and policies $\mathbf{x}$ and $\mathbf{y}$ for the two players, respectively, this yields

$$\dot{x}_i = x_i \Big[(\mathbf{A}\mathbf{y})_i - \mathbf{x}^{\mathrm{T}}\mathbf{A}\mathbf{y} \Big] \tag{7}$$

which are exactly the multi-population replicator dynamics of Equation 4.

The dynamical model of Equation 7 only describes the evolutionary process of *selection*, as Cross learning does not incorporate an exploration mechanism. However, in many scenarios *mutation* also plays a role, where individuals not only reproduce, but may change their behaviour while doing so. Given a population $\mathbf{x}$ as defined above, we consider a mutation rate $\mathcal{E}_{ij}$ indicating the propensity of species j to mutate into i (note the order of the indices), such that, $\forall i$:

$$\mathcal{E}_{ij} \geq 0 \quad \text{and} \quad \sum_i \mathcal{E}_{ij} = 1$$

Adding mutation to Equation 7 leads to a dynamical model with separate selection and mutation terms [20], given by

$$\dot{x}_i = x_i \underbrace{\left[(\mathbf{Ay})_i - \mathbf{x}^\mathrm{T}\mathbf{Ay} \right]}_{\text{selection}} + \underbrace{\sum_j \left(\mathcal{E}_{ij}x_j - \mathcal{E}_{ji}x_i \right)}_{\text{mutation}} \tag{8}$$

By slightly altering or extending the model of Equation 7 different RL algorithms can be modelled as well. A selection-mutation model of Boltzmann Q-learning (Eqs. 1 and 2) has been proposed by Tuyls et al. [36]. The dynamical system can again be decomposed into terms for exploitation (selection following the replicator dynamics) and exploration (mutation through randomization based on the Boltzmann mechanism):

$$\dot{x}_i = \frac{\alpha x_i}{\tau} \underbrace{\left[(\mathbf{Ay})_i - \mathbf{x}^\mathrm{T}\mathbf{Ay} \right]}_{\text{selection}} - \alpha x_i \underbrace{\left[\log x_i - \sum_k x_k \log x_k \right]}_{\text{mutation}} \tag{9}$$

Technically, these dynamics model the variant Frequency Adjusted Q-learning (FAQ), which mimics simultaneous action updates [23].

Lenient FAQ-learning (LFAQ) [6] is a variation aimed at overcoming convergence to suboptimal equilibria by mis-coordination in the early phase of the learning process, when mistakes by one agent may lead to penalties for others, irrespective of the quality of their actions. Leniency towards such mistakes can be achieved by collecting κ rewards for each action, and updating the Q-value based on the highest of those rewards. This causes an (optimistic) change in the expected reward for the actions of the learning agent, incorporating the probability of a potential reward for that action being the highest of κ consecutive tries [29]. The expected reward for each action $\mathbf{Ay}$ in Equation 9 is replaced by the utility vector $\mathbf{u}$, with

$$u_i = \sum_j \frac{a_{ij}y_j \left[\left(\sum_{k:a_{ik} \leq a_{ij}} y_k \right)^\kappa - \left(\sum_{k:a_{ik} < a_{ij}} y_k \right)^\kappa \right]}{\sum_{k:a_{ik} = a_{ij}} y_k} \tag{10}$$

Each of these models approximates the learning process of independent reinforcement learners in a multi-agent setting. Specifically, they are presented for the case of two-agent interacting in a normal-form game. Extensions to n-players are straightforward, but fall outside the scope of this work. In the next section we will describe our extension of networked replicator dynamics.

Algorithm 1 Update procedure for the NRD model

1: initialize $\mathcal{X}$
2: $\dot{\mathcal{X}} \leftarrow \mathbf{0}$
3: **for** $j = 1$ **to** N **do**
4:    **for all** $\mathbf{x}^k \in \mathbb{N}(v_j)$ **do**
5:        $\dot{x}_i^j \leftarrow \dot{x}_i^j + x_i^j \left[(\mathbf{A}\mathbf{x}^k)_i - \mathbf{x}^{j\mathrm{T}}\mathbf{A}\mathbf{x}^k \right]$
6:    **end for**
7:    $\dot{\mathbf{x}}^j \leftarrow \frac{\dot{\mathbf{x}}^j}{|\mathbb{N}(v_j)|}$
8: **end for**
9: $\mathcal{X} \leftarrow \mathcal{X} + \dot{\mathcal{X}}$

---

4 Networked Replicator Dynamics

In this work, agents are placed on the nodes of a network, and interact only locally with their direct neighbours. Assume a graph $\mathbb{G}$ with N nodes as defined in Section 2.3, with N agents placed on the nodes $\{v_1, \ldots, v_N\}$. If we define each agent by its current policy $\mathbf{x}$ we can write the current network state $\mathcal{X} = (\mathbf{x}^1, \ldots, \mathbf{x}^N)$. The aim of this work is study how $\mathcal{X}$ evolves over time, given the specific network structure and learning model of the agents. For this purpose, we introduce *networked replicator dynamics* (NRD), where each agent (or node) is modelled by a population of pure strategies, interacting with each if its neighbours following the multi-population replicator dynamics of Equation 4.

The update mechanism of the proposed networked replicator dynamics is given in Algorithm 1. At every time step, each agent (line 3) interacts with each of its neighbours (line 4) by playing a symmetric normal-form game defined by payoff-matrix $\mathbf{A}$. These interactions are modelled by the replicator dynamics (line 5), where each neighbour incurs a potential population change, $\dot{\mathbf{x}}$, in the agent. Those changes are normalised by the degree, $|\mathbb{N}(v_i)|$, of the agent's node (line 7). Finally, all agents update their state (line 9).

This model is flexible in that it is independent of the network structure, it can be used to simulate any symmetric normal form game, and different replicator models can easily be plugged in (line 5 of Algorithm 1). This means that we can use any of the dynamical models presented in Section 3 as update rule, thereby simulating different MARL algorithms.

5 Experimental Validation

In this section we present experimental results of the networked replicator dynamics model in various setups. In particular, we use Barabási-Albert *scale-free* [3] and Watts-Strogatz *small world* [40] networks. The first set of experiments compares the different learning models, focusing in particular on the role of exploration and lenience in the learning process. We then analyse lenience in more detail, investigating the influence of the degree of lenience on the speed of convergence. Hereafter, we look at the relation between network size and degree with respect to the equilibrium outcome. The last set of experiments investigates
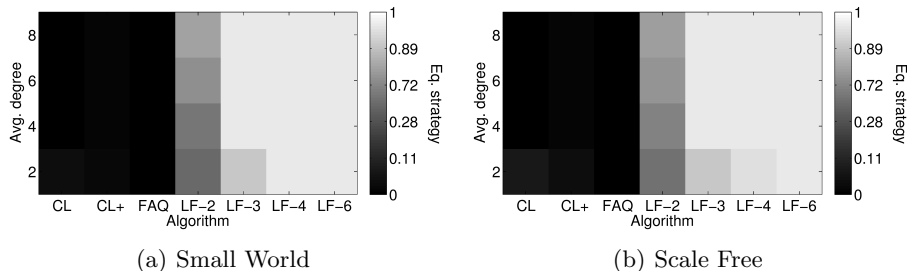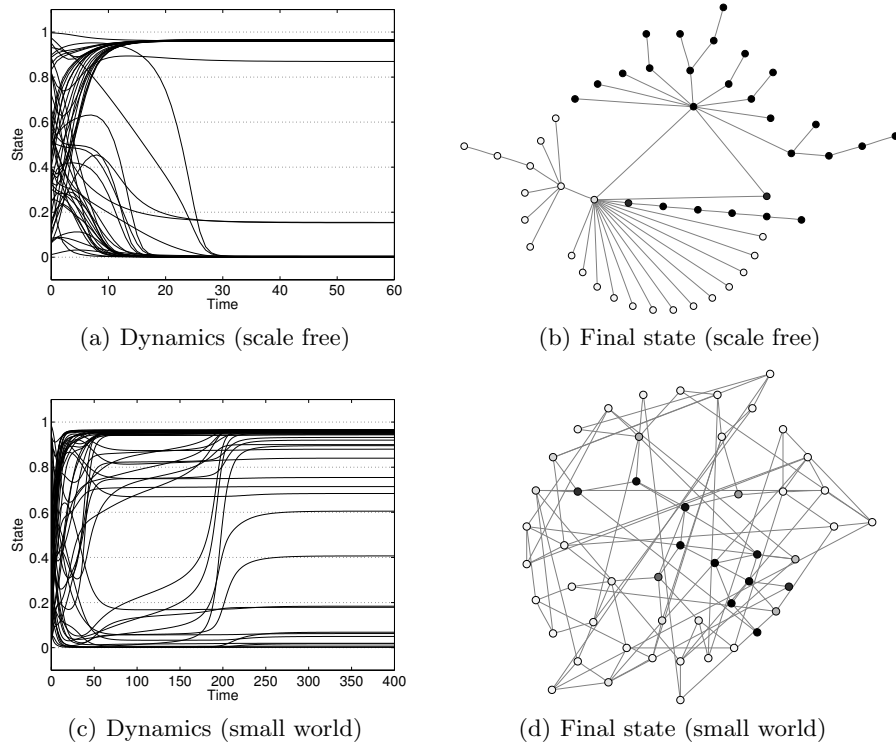
Fig. 2. Dynamics of a networked Stag Hunt game in small world and scale free networks. The figure shows the mean network state in equilibrium (gray scale) for different algorithms (x-axis) and average network degree (y-axis).

the role of stubborn nodes, which do not update their strategy, on the resulting network dynamics. All experiments use the Stag Hunt (page 4, Figure 1, right) as the model of interaction.

5.1 Comparing different learning models

We compare the different dynamical models of multi-agent learning presented before in Section 3. We use the following abbreviations: CL is Cross learning (Eq. 7); CL+ is CL with mutation (Eq. 8); FAQ is frequency adjusted Q-learning (Eq. 9); LF-κ is lenient FAQ with degree of lenience κ (Eq. 10). In order to ensure smooth dynamics we multiply the update $\dot{\mathbf{x}}$ of each model by a step size α. CL and CL+ use $\alpha = 0.5$, FAQ uses $\alpha = 0.1$, and LF uses $\alpha = 0.2$. Moreover, the exploration (mutation) rates are set as follows: CL+ uses $\mathcal{E}_{ij} = 0.01$ for all $i \neq j$, and $\mathcal{E}_{ii} = 1 - \sum_{j \neq i} \mathcal{E}_{ij}$; and FAQ and LF use $\tau = 0.1$. We simulate the model on 100 randomly generated networks of $N = 50$ nodes (both scale free and small world, the latter with rewiring probability $p = 0.5$), starting from 50 random initial states $\mathcal{X} \in \mathbb{R}^N$, and report the average network state $\bar{\mathcal{X}} = \frac{1}{N} \sum_i \mathbf{x}^i$ after convergence. Since the Stag Hunt only has two actions, the full state can be defined by x_1, the probability of the first action (cooperate).

Figure 2 shows the results of this comparison. The gray scale indicates the final network state $\bar{\mathcal{X}}$ after convergence, where black means defection, and white means cooperation. Note the non-linear scale, this is chosen to highlight the details in the low and high ranges of $\bar{\mathcal{X}}$. Several observations can be made based on these results. First of all, there is a clear distinction between non-lenient algorithms, which converge mostly to defection, and lenient algorithms that converge toward cooperation. As expected, lenience indeed promotes cooperation also in a networked interactions. Equally striking is the lack of distinction between pure selection (CL) and selection-mutation (CL+, FAQ) models. Adding mutation (or exploration) in this setting has no effect on the resulting convergence. Increasing the mutation rate does lead to a change at some point, however, this

(a) Dynamics (scale free)

(b) Final state (scale free)

(c) Dynamics (small world)

(d) Final state (small world)

Fig. 3. Example of the convergence of LF-2 on a Scale Free (top) and Small World (bottom) network with average degree 2 and 4, respectively. The network is split between cooperators (white) and defectors (black) in the final equilibrium state.

is to the exten that the added randomness automatically drives the equilibrium away from a state of pure defection.

The most interesting results of Figure 2 are those of LF-2. Here, we can observe a range of outcomes, depending on the average network degree. A more strongly connected network yields a higher probability of cooperation in equilibrium. Moreover, LF-2 is the only algorithm that yield an "indecisive" final state, that is significantly removed from pure cooperation or defection. In order to investigate this situation further, we look in detail at the dynamics of a single network. Figure 3(a) shows the network state $\mathcal{X}$ over time for one specific (randomly drawn) initial state of a scale free network with average degree 2. Clearly, the network is split into clusters of cooperators and defectors, no unanimous outcome is reached. The final state is highlighted in Figure 3(b), depicting the network structure and state of each node, and clearly showing two clusters. Depending on initial conditions, different splits can be observed.

Table 1. Time to convergence (mean and std. dev.) of lenient FAQ, for Small World and Scale Free networks of various degree d.

| Algorithm	Small World				Scale Free			
	$d = 2$	$d = 4$	$d = 6$	$d = 8$	$d = 2$	$d = 4$	$d = 6$	$d = 8$
LF-2	148 (71)	72 (50)	47 (21)	43 (12)	81 (53)	50 (28)	41 (7)	40 (6)
LF-3	72 (58)	36 (3)	35 (1)	35 (1)	44 (21)	36 (2)	35 (2)	35 (1)
LF-4	43 (24)	34 (1)	34 (1)	34 (1)	38 (13)	34 (1)	34 (1)	34 (1)
LF-6	35 (12)	33 (1)	33 (1)	33 (1)	35 (8)	33 (1)	33 (1)	33 (1)

Similar results can be observed in small world networks. Figures 3(c)-(d) show the dynamics in an example network with average degree 4. Again, a cluster of defectors is maintained in equilibrium amongst a majority of cooperators. Identifying specific structural network properties that lead to clustering is a main question for future work.

5.2   The effect of lenience on convergence

In this set of experiments, we take a closer look at the influence of leniency on the dynamics and convergence of the network. Using the same set of networks as in the previous section, we zoom in only on the lenient algorithms and compare their convergence speed for the different networks. Table 1 lists the number of time steps to convergence, again averaged over 100 networks with 50 random initial states. Two trends are clearly visible: increasing the degree of lenience decreases the convergence time (most notably for degree 2 networks); and increasing the network degree similarly decreases the convergence time (most notably for LF-2). These results can be explained intuitively, as lenience pushes the learning process in the direction of cooperation, whereas a higher network degree yield more interactions per time step, and hence faster convergence. The fact that no convergence below 33 time steps is observed, independent of the network type, can be explained by the limit that the step size α and the inherent dynamics of the model pose.

5.3   The relation between network size and degree

Here we investigate the role that both network size and average degree play in determining the equilibrium outcome of the learning process. Specifically, we compare networks of different sizes with a fixed degree, with networks which have a degree proportional to their size. Figure 4 shows the results for both small world and scale free networks. For each combination we simulate 100 randomly generated networks, each using 10 randomly drawn initial states, following the LF-2 dynamics. The figure shows that the equilibrium state is independent of the network size if the degree is kept fixed, whereas the probability of cooperation increases when the degree grows with the network. This result shows that a more strongly connected network tends to cooperate more than one with sparse interactions. Intuitively, this can be explained by the inherent dynamics of the

(a) Small World   (b) Scale Free
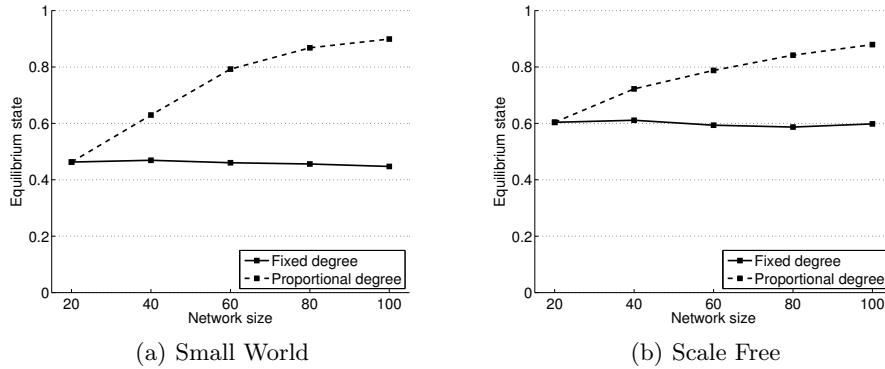
Fig. 4. The equilibrium state for different network sizes, for Small World and Scale Free networks using LF-2. Fixed degree is 2, proportional degree is 10% of the network size.

Stag Hunt: a critical mass of cooperators is required for cooperation to be a beneficial strategy. In more densely connected networks, this critical mass is reached more easily.

5.4   The influence of stubborn agents

Finally, we look at the influence of *stubborn agents* on the final state. Stubborn agents are ones that do not update their state, regardless of the actions of their neighbours or the rewards they receive. These agents could be perceived as regulating bodies in financial networks, or politicians in social networks trying to spread their views.

Here, we select the highest degree nodes in the network to be stubborn - future work will investigate this issue further. Figure 5 shows the results of an extensive set of experiments, simulating networks of different sizes $N \in \{20, 40, 60, 80, 100\}$ with average degree 2, and varying the percentage of stubborn agents. The stubborn agents keep their state fixed at $x_1 = 0.95$.[2] Interestingly, the results are independent of the network size when the degree is fixed, and hence the results in Figure 5 are averaged. We can observe that stubborn agents pull the whole network toward cooperation. Moreover, we see that this effect diminishes as the percentage goes up. Scale free networks in particular show this effect, which can be explained by the fact the in such networks a small number of "hubs" take part in a majority of the connections. Once these hubs are cooperative, the rest follows quickly.

---

[2] Note that we exclude these fixed nodes from the results presented here, however a similar trend can be observed it they are included.
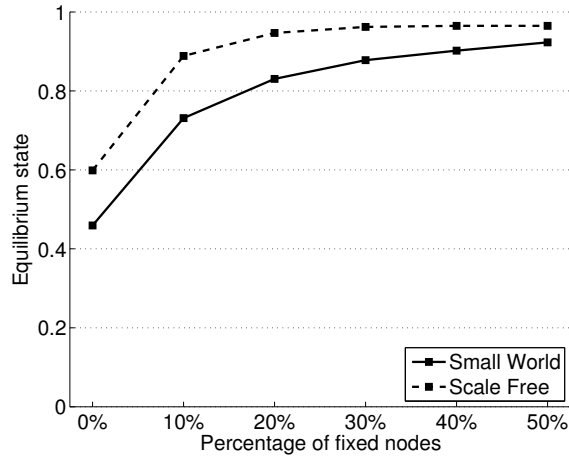
Fig. 5. The influence of the number of stubborn agents on final network state, for Small World and Scale Free networks of degree 2.

6   Conclusions

We have proposed networked replicator dynamics (NRD) that can be used to model learning in (social) networks. The model leverages the link between evolutionary game theory and multi-agent learning, that exists for unstructured populations, and extends it to settings in which agents only interact locally with their direct network neighbours. We evaluated this model in a range of experiments, showing the effect of various properties of both network and learning mechanism on the resulting equilibrium state. We found that lenience is an enabler for cooperation in a networked Stag Hunt game. A higher degree of lenience yields a higher probability of cooperation in this game; moreover this equilibrium is reached faster. More densely connected networks promote cooperation in a similar way, and stubborn agents can pull the network towards their desired state, in particular when they are well connected within the network. The latter finding is of particular interest to the scenario of adoption of new technologies, as it shows that getting few key players to opt-in may pull the whole network to adopt as well.

There are many interesting avenues for future work stemming from these initial findings. The networked replicator dynamics can be further validated by comparing these findings with the dynamics that would result from placing actual learning agents, rather than their dynamical model, on the network. Moreover, one can look at networks in which different types of learning mechanisms interact. E.g., each agent is modelled by a different dynamical model. This can be easily integrated in the NRD. Furthermore, different games can be studied as the model for various real-world scenarios, such as the N-player Stag Hunt which yields richer dynamics than its two-player counterpart [5, 28]. Finally, an

interesting direction for further research would be to extend the NRD model for more complex learning algorithms. For example, it has been shown that adding memory can help sustain cooperation by taking past encounters into account, e.g. by recording the opponent's intention [1] or by the inclination to stick to previous actions [13].

Acknowledgements

References

1. Ahn, H.T., Pereira, L.M., Santos, F.C.: Intention recognition promotes the emergence of cooperation. Adaptive Behavior 19(4), 264–279 (2011)
2. Backstrom, L., Boldi, P., Rosa, M.: Four degrees of separation. Arxiv preprint arXiv:1111.4570 (2011)
3. Barabási, A.L., Albert, R.: Emergence of scaling in random networks. Science 286(5439), 509–12 (Oct 1999)
4. Barabási, A.L., Oltvai, Z.N.: Network biology: understanding the cell's functional organization. Nature reviews. Genetics 5(2), 101–13 (Feb 2004)
5. Bloembergen, D., De Jong, S., Tuyls, K.: Lenient learning in a multiplayer stag hunt. In: Proc. of 23rd Benelux Conf. on Artificial Intelligence (BNAIC 2011). pp. 44–50 (2011)
6. Bloembergen, D., Kaisers, M., Tuyls, K.: Empirical and theoretical support for lenient learning. In: Tumer, Yolum, Sonenberg, Stone (eds.) Proc. of 10th Intl. Conf. on AAMAS 2011. pp. 1105–1106. International Foundation for AAMAS (2011)
7. Bloembergen, D., Ranjbar-Sahraei, B., Ammar, H.B., Tuyls, K., Weiss, G.: Influencing social networks: An optimal control study. In: Proc. of the 21st ECAI 2014. pp. 105–110 (2014)
8. Börgers, T., Sarin, R.: Learning through reinforcement and replicator dynamics. Journal of Economic Theory 77(1) (November 1997)
9. Boyd, R., Gintis, H., Bowles, S.: Coordinated punishment of defectors sustains cooperation and can proliferate when rare. Science 328(5978), 617–620 (2010)
10. Bullmore, E., Sporns, O.: Complex brain networks: graph theoretical analysis of structural and functional systems. Nature reviews. Neuroscience 10(3), 186–98 (Mar 2009)
11. Busoniu, L., Babuska, R., De Schutter, B.: A comprehensive survey of multiagent reinforcement learning. IEEE Trans. on Systems, Man, and Cybernetics, Part C: Applications and Reviews 38(2), 156–172 (2008)
12. Chapman, M., Tyson, G., Atkinson, K., Luck, M., McBurney, P.: Social networking and information diffusion in automated markets. In: Joint Inter. Workshop on TADA and AMEC. pp. 1–14 (2012)
13. Cimini, G., Sánchez, A.: Learning dynamics explains human behaviour in prisoner's dilemma on networks. Journal of The Royal Society Interface 11(94), 20131186 (2014)
14. Cross, J.G.: A stochastic learning model of economic behavior. The Quarterly Journal of Economics 87(2), 239–266 (1973)

15. Dickens, L., Broda, K., Russo, A.: The dynamics of multi-agent reinforcement learning. In: ECAI. pp. 367–372 (2010)
16. Easley, D., Kleinberg, J.: Networks, Crowds, and Markets: Reasoning about a Highly Connected World. Cambridge University Press (2010)
17. Edmonds, B., Norling, E., Hales, D.: Towards the evolution of social structure. Computational and Mathematical Organization Theory 15(2), 78–94 (2009)
18. Ghanem, A.G., Vedanarayanan, S., Minai, A.A.: Agents of influence in social networks. In: Proc. of 11th Int. Conf. on AAMAS 2012 (2012)
19. Han, T.A., Pereira, L.M., Santos, F.C., Lenaerts, T.: Good agreements make good friends. Scientific reports 3 (2013)
20. Hofbauer, J., Sigmund, K.: Evolutionary games and population dynamics. Cambridge University Press (1998)
21. Hofmann, L.M., Chakraborty, N., Sycara, K.: The Evolution of Cooperation in Self-Interested Agent Societies: A Critical Study. Proc. of 10th Int. Conf. on AAMAS 2011 pp. 685–692 (2011)
22. Jackson, M.O.: Social and Economic Networks. Princeton University Press (2008)
23. Kaisers, M., Tuyls, K.: Frequency adjusted multi-agent Q-learning. In: Proc. of 9th Intl. Conf. on AAMAS 2010. pp. 309–315 (May, 10-14, 2010)
24. Maynard Smith, J., Price, G.R.: The logic of animal conflict. Nature 246(2), 15–18 (1973)
25. Narendra, K.S., Thathachar, M.A.L.: Learning automata - a survey. IEEE Transactions on Systems, Man, and Cybernetics 4(4), 323–334 (1974)
26. Nowak, M.A., May, R.M.: Evolutionary games and spatial chaos. Nature 359(6398), 826–829 (1992)
27. Ohtsuki, H., Hauert, C., Lieberman, E., Nowak, M.A.: A simple rule for the evolution of cooperation on graphs and social networks. Nature 441(7092), 502–505 (2006)
28. Pacheco, J.M., Santos, F.C., Souza, M.O., Skyrms, B.: Evolutionary dynamics of collective action in n-person stag hunt dilemmas. In: Proceedings of the Royal Society B: Biological Sciences. vol. 276, pp. 315–321 (2009)
29. Panait, L., Tuyls, K., Luke, S.: Theoretical advantages of lenient learners: An evolutionary game theoretic perspective. Journal of Machine Learning Research 9, 423–457 (2008)
30. Ranjbar-Sahraei, B., Bou Ammar, H., Bloembergen, D., Tuyls, K., Weiss, G.: Evolution of cooperation in arbitrary complex networks. In: Proc. of the 2014 Int. Conf. on AAMAS 2014. pp. 677–684. International Foundation for AAMAS (2014)
31. Santos, F., Pacheco, J.: Scale-free networks provide a unifying framework for the emergence of cooperation. Physical Review Letters 95(9), 1–4 (Aug 2005)
32. Sigmund, K., Hauert, C., Nowak, M.A.: Reward and punishment. Proc. of the National Academy of Sciences 98(19), 10757–10762 (2001)
33. Skyrms, B.: The stag hunt and the evolution of social structure. Cambridge University Press (2004)
34. Sutton, R., Barto, A.: Reinforcement Learning: An introduction. MA: MIT Press, Cambridge (1998)
35. Thathachar, M., Sastry, P.S.: Varieties of learning automata: an overview. IEEE Trans. on Systems, Man, and Cybernetics, Part B: Cybernetics 32(6), 711–722 (2002)
36. Tuyls, K., Verbeeck, K., Lenaerts, T.: A selection-mutation model for q-learning in multi-agent systems. In: Proc. of 2nd Intl. Conf. on AAMAS 2003. pp. 693–700. ACM, New York, NY, USA (2003)

37. Ugander, J., Karrer, B., Backstrom, L., Marlow, C.: The anatomy of the facebook social graph. arXiv preprint arXiv:1111.4503 pp. 1–17 (2011)
38. Van Segbroeck, S., de Jong, S., Nowe, A., Santos, F.C., Lenaerts, T.: Learning to coordinate in complex networks. Adaptive Behavior 18(5), 416–427 (Nov 2010)
39. Watkins, C.J.C.H., Dayan, P.: Q-learning. Machine Learning 8(3), 279–292 (1992)
40. Watts, D.J., Strogatz, S.H.: Collective dynamics of 'small-world' networks. nature 393(6684), 440–442 (1998)
41. Weibull, J.W.: Evolutionary game theory. MIT press (1997)
42. Zimmermann, M.G., Eguíluz, V.M.: Cooperation, social networks, and the emergence of leadership in a prisoner's dilemma with adaptive local interactions. Physical Review E 72(5) (Nov 2005)