The Dynamics of Q-learning in Population Games: a Physics-Inspired Continuity Equation Model

Shuyue Hu[†], Chin-Wing Leung[‡], Ho-fung Leung[‡], Harold Soh[†]
National University of Singapore[†], The Chinese University of Hong Kong[‡]
Singapore[†], Hong Kong SAR, China[‡]
shuyuehu217@gmail.com,{cwleung,lhf}@cse.cuhk.edu.hk,harold@comp.nus.edu.sg

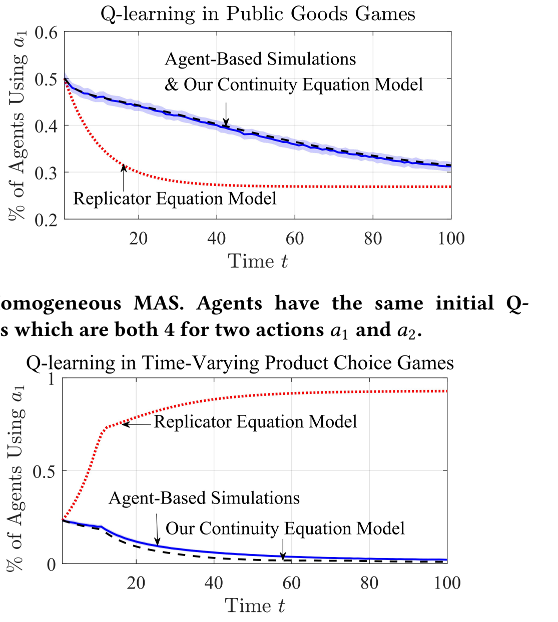
ABSTRACT

Although learning has found wide application in multi-agent systems, its effects on the temporal evolution of a system are far from understood. This paper focuses on the dynamics of Q-learning in large-scale multi-agent systems modeled as population games. We revisit the replicator equation model for Q-learning dynamics and observe that this model is inappropriate for our concerned setting. Motivated by this, we develop a new formal model, which bears a formal connection with the continuity equation in physics. We show that our model always accurately describes the Q-learning dynamics in population games across different initial settings of MASs and game configurations. We also show that our model can be applied to different exploration mechanisms, describe the mean dynamics, and be extended to Q-learning in 2-player and n-player games. Last but not least, we show that our model can provide insights into algorithm parameters and facilitate parameter tuning.

KEYWORDS

Q-Learning; Mathematical Modeling; Population Game

ACM Reference Format:
Shuyue Hu[†], Chin-Wing Leung[‡], Ho-fung Leung[‡], Harold Soh[†]. 2022. The Dynamics of Q-learning in Population Games: a Physics-Inspired Continuity Equation Model. In *Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), Online, May 9–13, 2022*, IFAAMAS, 9 pages.



(a) Homogeneous MAS. Agents have the same initial Q-values which are both 4 for two actions a_1 and a_2.

(b) Heterogeneous MAS. Initial Q-values for actions a_1 and a_2 are distributed according to $\mathrm{Beta}(15, 30)$ and $\mathrm{Beta}(10, 10)$, respectively, with support $[-1.5, 1.5]$.

Figure 1: Comparison among the population dynamics described by the Replicator Equation Model [32, 37] (REM, dotted red line) and our Continuity Equation Model (CEM, dashed black line), and the actual dynamics averaged over 100 runs of agent-based simulations (shaded blue line with the shaded area representing the standard deviation). The game configurations are summarized in Table 1, the Boltzmann temperature is 3, and the learning rate is 0.1. In both homogeneous and heterogeneous MASs, our CEM better captures the qualitative and quantitative dynamics of the systems.

1 INTRODUCTION

Recent years have witnessed a significant gain in the learning capability of intelligent agents. These advances have spurred the usage of learning agents in many *large-scale* multi-agent systems (MASs) that are concerned with a great number of agents, such as autonomous vehicles for transportation [28], online trading/bidding agents in financial markets [39], and cooperative robots for search and rescue [24]. However, despite wide application, learning in large-scale MASs is far from understood and its theoretical underpinnings remain elusive.

Population games are canonical models of strategic interactions of large-scale MASs [29]. Traditionally, a multi-agent learning (MAL) algorithm is often examined by whether the strategy profile will converge to a (e.g., Nash) equilibrium in games [e.g., 6, 33, 42]. However, emergent theoretical research has shifted its focus to the *dynamics* because static equilibrium notions are fundamentally limiting — they cannot express any temporal evolution of a system nor long-term non-equilibrium phenomena [e.g., 1, 5, 31, 38]. As Tuyls and Parsons [36] voice, the development of theory in this direction is crucial because it will not only yield a better theoretical understanding of existing algorithms, but potentially facilitate the design of new methods, leading to practical algorithmic advancements.

In this work, we focus on the dynamics of Q-learning in population games. Q-learning, as proposed by Watkins & Dayan [40], is one of the most important learning algorithms in AI literature. It forms the basis of numerous learning methods and is a main focus of many theories in MAL [e.g., 19, 27, 41]. In their seminal works, Tuyls et al. [37] and Sato & Crutchfield [32] proposed the *replicator equation model* (REM)[1] to describe the dynamics of agents that

Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), P. Faliszewski, V. Mascardi, C. Pelachaud, M.E. Taylor (eds.), May 9–13, 2022, Online. © 2022 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

[1] In [37], the model is called the selection-mutation model.

apply Q-learning with Boltzmann exploration in 2-player normal-form games. The REM reveals a surprising connection between multi-agent Q-learning and the well-known *replicator dynamics* of evolutionary game theory (EGT). This connection paved the way to the study Q-learning from the EGT perspective and has inspired many works in the MAL literature [e.g., 7, 8, 11]. More recently, Leonardos et al. applied the REM to n-player games [22] and population games with homogeneous populations [23]; using the REM as an example, they provided new mechanisms to induce phase transitions between multiple equilibria in MASs.

Although many studies of Q-learning dynamics are based on the REM [e.g., 12, 19, 26], we observe that the REM is *inappropriate* for Q-learning in general population games. To elaborate, the REM was designed for multi-player games with a discrete number of agents and simplifies the canonical Q-learning dynamics by (i) tracking only the policies of individual agents, and (ii) assuming each agent is performing multiple updates of the Q-values for each update of the policy. While these simplifications are natural in certain settings, they cause the model to neither (i) differentiate between agents that have different Q-values but happen to have the same policy at a given time step, nor (ii) capture the effects of the asynchronous update in Q-learning. As such, the REM can be inexact when applied to Q-learning in population games which feature large and generally heterogeneous populations. As shown by the example in Figure 1, the dynamics prescribed by the REM do not match the actual dynamics in agent-based simulations; sometimes, the REM even suggests a system outcome that is completely different than the ground truth (Figure 1(b)).

Motivated by this observation, we develop a new formal model for Q-learning in population games. Rather than only tracking agent policies, we directly track the Q-values of individual agents. Moreover, we propose to tackle the asynchronous update of Q-learning by modeling its stochastic effects on Q-values. Note that unlike 2-player games, population games involve infinitely many agents that typically have diverse initial Q-values and develop different policies afterwards. This poses a new challenge: how can we characterize the effects of population heterogeneity on Q-learning dynamics? To address this challenge, we focus on the distribution of Q-values in the population; in particular, we investigate the evolution of this distribution function as time progresses, and derive a *differential equation* to model its temporal evolution. Our proposed solution is inspired by statistical physics, where studying the dynamics of a probability distribution rather than the dynamics of individuals is a classic approach (examples range from the heat transfer equation to the Fokker–Planck equation for Brownian motion).

The resultant model (Equation 14) from our approach takes the form of a partial differential equation (PDE), which is fundamentally different from the REM that is based on ordinary differential equations (ODEs). In particular, we find that our model can be viewed as a *continuity equation* that describes the transport phenomena (e.g., of mass or energy) in a physical system. This suggests a connection between MAL and physics — the Q-learning dynamics in population games is analogously the transport of the agent mass in the Q-value space. Moreover, we observe that our continuity equation model (CEM) has some interesting properties (Section 5.2) — CEM can (i) be applied to different exploration mechanisms, (ii) describe the mean dynamics [30] (the temporal evolution of the mean policy) in the system, (iii) be reduced to a system of coupled ODEs for homogeneous populations, and (iv) be extended to model Q-learning dynamics in 2-player games and n-player games.

In our experiments, we validate that given different population games and initial settings of MASs, our CEM always provides an accurate description of Q-learning dynamics with respect to the actual dynamics in agent-based simulations (Section 6.1). In addition, we illustrate two potential use cases of our model. Through a concrete example, we show that our CEM can provide non-trivial insights into the effects of algorithm parameters (particularly, the temperature of Boltzmann exploration); these insights lead to practical guidelines for notoriously cumbersome parameter tuning (Section 6.2). We also show that our CEM can contrast the dynamics that arise from different exploration mechanisms, which potentially can facilitate the choice of exploration mechanisms (Section 4 in the supplementary). The supplementary of this paper can be found online [14, 15].

To summarize, our key contributions are:

- An analysis of the limitations of the well-known replication equation model in settings with heterogeneous agents that perform asynchronous updates;
- The development of a new theoretical model for Q-learning dynamics in population games, which bears a formal connection with the continuity equation in physics;
- Experimental results validating the descriptive power of our continuity equation model, and two examples illustrating its use to gain insights into the effects of algorithm parameters and to contrast the dynamics that arise from different exploration mechanisms.

2 RELATED WORK

Previous works that examine Q-learning dynamics are largely based on the REM proposed by Tuyls et al. [37] and Sato & Crutchfield [32]. Panozzo et al. [26] introduced an extension of the REM for Q-learning that operates on sequence forms. Based on the REM, Kianercy & Galstyan [19] provided a comprehensive characterization of the fixed point structure for Q-learning in different 2-player games. Kaisers & Tuyls [17] noticed that the prediction of the REM in 2-player games may deviate from the actual Q-learning dynamics; but rather than developing a more accurate model for Q-learning, they proposed a new algorithm that is more consistent with what the REM predicts. More recently, the REM has been applied to study phase transitions in *n*-player games [22] and population games where agents have the same initial policy [23]; Leonardo et al. [22, 23] showed that by tuning the exploration parameter, there are phase transitions between multiple equilibria in MASs. The REM has inspired many works to study MAL (not limited to Q-learning) using EGT approaches; we refer interested readers to a recent survey [2] and references therein. However, the REM is unable to provide an appropriate model of Q-learning in general population games due to the simplifications it makes.

There are few exceptions that examine Q-learning dynamics without the use of REM and its variants [16, 27, 41]. Gomes & Kowalczyk [27] and Wunder et al. [41] focused on Q-learning with ϵ-greedy exploration in 2-player games; however, their approaches are tailored to address the discontinuity caused by ϵ-greedy exploration and are not applicable to large agent populations. Hu et al. [16] considered an n-agent setting where Q-learning agents with Boltzmann exploration are paired up to play 2-player games; using mean field theory, they reduced the setting to an 2-agent setting and developed a Fokker-Planck equation for the learning dynamics. As we shall discuss in Section 5.2, our model can be generalized to their setting, even though 2-player games and population games are different in nature. In this sense, our model can be viewed as a generalization of [16], which goes beyond 2-player games and Boltzmann exploration.

Lahkar & Seymour [20] studied Cross learning in population games and also derived a continuity equation for the learning dynamics. In addition to the difference in algorithms (Cross learning is policy-based whereas Q-learning is value-based), their approach is *incompatible* with Q-learning. Specifically, their approach is to work with the distribution of the policy in the population. However, as we shall show in Section 4.2, tracking the policies of agents can be misleading for Q-learning (especially in population games); the policy dynamics are not equivalent to the Q-values dynamics.

Mean field games are also concerned with infinitely many agents [9, 33]. Lasry & Lions [21] developed a system of two coupled PDEs — a Fokker-Planck equation and a Hamilton–Jacobi–Bellman equation — for the mean-field game theory setting. However, rather than Q-learning, agents in [21] apply optimal control to a well understood system with complete observation of the system state.

3 PRELIMINARIES

In this paper, we consider Q-learning in population games. Specifically, at each time step t, a population of Q-learning agents each takes an action independently. Based on the action applied and the population state, each agent receives an immediate reward in the game and adapts its Q-value and policy accordingly. At the next time step $t + 1$, agents start over for another play of the game. In this section, we define population games and Q-learning.

3.1 Population games

The population game is a widely adopted framework for modeling strategic interactions that are commonly observed in large-scale MASs [30], such as network congestion, task allocation, and social norm emergence. Specifically, population games model scenarios that simultaneously exhibit three properties: (i) the number of agents is large, (ii) each agent is small, such that any particular one agent's behavior has little or negligible effect on other individual agents, and (iii) each agent is anonymous, in that exchanging the labels of agents will not create any difference.

Consider a set $\mathcal{N} = \{1, \ldots, n\}$ of n agents with $n \to \infty$ and a set $\mathcal{A} = \{a_1, \ldots, a_m\}$ of m actions available to each agent. Suppose that a population game will be played for T time steps. An agent's payoffs in a population game depend only on its own behavior and the aggregated effect of the other agents' behaviors which is usually termed as *population state*. For every time step t, the population state is represented by a vector $\vec{o}_t = [o_{1,t}, \ldots,\ o_{m,t}]^\top$, where $o_{j,t}$ is the proportion of agents taking action $a_j \in \mathcal{A}$ in the population at time t. The reward function is given by $R(a_j, \vec{o}_t, t)$ which determines the payoff of an agent by the action a_j it uses and the population state $\vec{o}_t$ at time t. In general, the reward function of a population game can change over time. The population state evolves as agents interact with one another.

3.2 Q-learning and Boltzmann Exploration

Q-learning [40] is typically defined in the context of a Markov decision process (MDP). In this work, we focus on population games where there are no environmental state transitions. Environmental statelessness is a common assumption made in theory for MAL [e.g., 1, 5, 16] and simplifies analysis and exposition. A stateless MDP consists of a set $\mathcal{A}$ of available actions and an immediate reward function that gives the reward of using each action. For a stateless MDP, Q-learning maintains a Q-value for each action. Consider an arbitrary Q-learning agent i. We define the set of Q-values of agent i at time t to be $\vec{Q}^i_t = [Q^i_{1,t}, \ldots, Q^i_{m,t}]^\top$ where $Q^i_{j,t}$ is the Q-value for action a_j. Suppose that at time t, agent i plays the action a_j and receives an immediate reward $r^i_{j,t} = R(a_j, \vec{o}_t, t)$ determined by the reward function of the population game. The agent i will update the Q-value of action a_j as follows:

$$Q^i_{j,t+1} = (1 - \alpha)Q^i_{j,t} + \alpha r^i_{j,t} \tag{1}$$

where α is the learning rate. Note that for every time step, only the Q-value of the action in use is updated; the Q-values of the other actions (that are not applied at this time step) remain *unchanged*.

There are multiple mechanisms for a Q-learning agent to select an action based on its Q-values. We define the policy of agent i at time t to be $\vec{x}^i_t = [x^i_{1,t}, \ldots, x^i_{m,t}]^\top$ where $x^i_{j,t}$ is the probability that agent i uses action a_j. For Boltzmann exploration, the value of $x^i_{j,t}$ is given by $x^i_{j,t} = e^{\tau Q^i_{j,t}} / [\sum_{k=1}^m e^{\tau Q^i_{k,t}}]$, where $\tau \in [0, \infty)$ is the Boltzmann temperature that controls how much the agent explores. The agent is in pure exploration (randomly taking each action) when τ is 0, and in pure exploitation (greedily taking the action with the highest Q-value) when $\tau \to \infty$.

4 REPLICATOR EQUATION MODEL REVISITED

In this section, we revisit the REM [32, 37] for Q-learning with Boltzmann exploration. Tuyls et al. [37] and Sato & Crutchfield [32] developed this model for Q-learning in 2-player games. Recent work [23] has applied the REM to population games with homogeneous populations. We describe this model in Section 4.1. In Section 4.2, we analyze two simplifications that this model makes for Q-learning. In Section 4.3, we discuss the application of this model to population games and show that this model can provide inexact predictions under this setting. For ease of presentation, in this paper, we consider a set $\mathcal{A} = \{a_1, a_2\}$ of two actions; generalization of our analysis/approach to cases with more than two actions is straightforward.

4.1 Replicator Equation Model

In their seminal work, Tuyls et al. [37] and Sato & Crutchfield [32] developed replicator equations to model the dynamics of Q-learning with Boltzmann exploration in 2-player games. Let i denote an arbitrary player in a 2-player game. The replicator equation that models the time evolution of the policy of agent i is given as follows:

$$\frac{dx^i_{j,t}}{dt} = \underbrace{\alpha\tau\, x^i_{j,t}\left(r^i_{j,t} - \sum_{a_k \in \mathcal{A}} x^i_{k,t} r^i_{k,t}\right)}_{T_1} \underbrace{+\alpha\, x^i_{j,t} \sum_{a_k \in \mathcal{A}} x^i_{k,t} \ln \frac{x^i_{k,t}}{x^i_{j,t}}}_{T_2} \quad (2)$$

where $x^i_{j,t}$ is the probability that agent i uses any action $a_j \in \mathcal{A}$ at time t. Note that the term T_1 is exactly the well-known replicator dynamics capturing the selection mechanism in EGT, and the term T_2 can be decomposed into two entropy terms handling the mutation mechanism in EGT [37]. Therefore, this model elegantly brings forward the connection between multi-agent Q-learning and EGT.

4.2 Replicator Equation Model vs Q-learning

Representation of Q-learners. One simplification in the REM is that it represents agents with their policies and does not differentiate between agents that have different Q-values but the same policy. Suppose at time t, agents i and j have the same policy but different Q-values, and both apply action a_1. Equation 2 dictates that if two agents have the same learning parameters and reward functions, they will develop *exactly the same* policy; the changes in their polices do not explicitly depend on their Q-values. However, this is generally *not* true. Consider the policy of agent i for time $t + 1$

$$\begin{aligned} x^i_{1,t+1} &= \frac{e^{\tau Q^i_{1,t+1}}}{e^{\tau Q^i_{1,t+1}} + e^{\tau Q^i_{2,t+1}}} = \frac{1}{1 + e^{\tau\left(Q^i_{1,t+1} - Q^i_{2,t+1}\right)}} \\ &= \frac{1}{1 + e^{\tau\left[Q^i_{1,t} - Q^i_{2,t} + \alpha r^i_{1,t} - \alpha Q^i_{1,t}\right]}} \end{aligned} \quad (3)$$

and $x^i_{2,t+1} = 1 - x^i_{1,t+1}$. The precondition of agents i and j having the same policy at time t only ensures $Q^i_{1,t} - Q^i_{2,t} = Q^j_{1,t} - Q^j_{2,t}$ (this can be inferred from the second equality). Therefore, agents will not necessarily develop the same policy for time $t + 1$ if they do not have the same Q-values at time t. This observation suggests that representing a Q-learner with its policy and tracking only its policy may not provide a good description of its dynamics.

Update frequency of Q-values. Another difference between the REM and Q-learning is that the REM implicitly assumes that at each time step, a Q-learner will update the Q-values for *every* action rather than only the action in use. To see this, we make use of the equality $\ln(x^i_{2,t}/x^i_{1,t}) = \tau(Q^i_{2,t} - Q^i_{1,t})$, and rewrite Equation 1 as

$$\frac{dx^i_{1,t}}{dt} = \frac{\partial x^i_{1,t}}{\partial Q^i_{1,t}} \alpha(r^i_{1,t} - Q^i_{1,t}) + \frac{\partial x^i_{1,t}}{\partial Q^i_{2,t}} \alpha(r^i_{2,t} - Q^i_{2,t}). \quad (4)$$

By the chain rule, we also have $\frac{dx^i_{1,t}}{dt} = \frac{\partial x^i_{1,t}}{\partial Q^i_{1,t}} \frac{dQ^i_{1,t}}{dt} + \frac{\partial x^i_{1,t}}{\partial Q^i_{2,t}} \frac{dQ^i_{2,t}}{dt}$, which suggests the model assumes that for each action $a_j \in \mathcal{A}$, $\frac{dQ^i_{j,t}}{dt} = \alpha(r^i_{j,t} - Q^i_{j,t})$. From this, we see that the Q-value of *every* action is always updated at a given time step. This contradicts the standard asynchronous update rule of Q-learning — only the action in use should be updated.

Note that our above analysis is *not* limited to only a specific type of games (e.g. 2-player games). In other words, the above two issues in representation and update frequency are inherent in the model no matter what games the model is applied to.

Importantly, we emphasize that we do *not* claim that the REM is wrong or inferior in general. Indeed, if one considers that each agent updates its Q-values for all actions synchronously, the simplifications pointed out above will vanish, and the REM will provide an accurate and precise description of the learning dynamics. There are two possibility for such synchronous updates: (i) agents perform many interactions before updating their Q-values (or put differently, the learning dynamics is very slow compared to interactions [32]), and (ii) agents apply the Frequency Adjusted Q-learning [18].[2] Nevertheless, as defined by Watkins & Dayan [40], the *asynchronous* update rule of Q-learning is standard and important; this is a norm in the literature for Q-learning and its variants (examples include impactful algorithms [10, 13, 34]). Therefore, the simplifications pointed out above are non-trivial and require formal treatment for the dynamics of Q-learning.

4.3 Application to Population Games

Recent work [23] has applied the REM to population games in *homogeneous* MASs where all agents have the same initial policy. Due to the symmetry of agents, the superscript i in Equation 2 can be dropped; thus, the model describes how the policy, which is the same for every agent, evolves as time t progresses. This model can also be applied to population games in *heterogeneous* MASs where agents have diverse initial Q-values and start with different policies. To achieve this, in Equation 2, one can replace $x^i_{j,t}$ with $o_{j,t}$ and $r^i_{j,t}$ with $R(a_j, \vec{o}_t, t)$; here, Equation 2 models the dynamics of population action frequencies.[3]

We hypothesize that the two issues pointed out in Sec. 4.2 above, coupled with potential population heterogeneity in population games, conspire to cause inexact descriptions under the concerned setting. Intuitively, because the REM implicitly assumes synchronous updates of Q-values, the learning speed predicted by the model is likely to deviate. In addition, because the REM considers agents that have the same policy but different Q-values to be identical, the effects of population heterogeneity are underestimated. Unlike 2-player games, population heterogeneity generally plays an important role in population games given infinitely many agents. Thus, the REM may provide a less accurate description of Q-learning dynamics in population games than in 2-player games.

To verify our hypothesis, we compare the dynamics predicted by the model against agent-based simulation results (which are the ground truth), given the same initial settings of MASs. In this work, for each comparison, we performed 100 independent simulation runs to generate the simulation results; for each run, there were 1,000 agents. It is clear in Figure 1(a) that there is a noticeable discrepancy in the speed of convergence even for homogeneous MASs playing the relatively simple public goods game (where there is a unique Nash equilibrium). As shown in Figure 1(b), for heterogeneous MASs playing the time-varying product choice game (where

[2] Kaisers and Tuyls [18] reported a similar finding on the update frequency that the model assumes. They argued that the behaviors predicted by the REM are more desirable and proposed the Frequency Adjusted Q-learning whose dynamics in 2-player games is more consistent with what the REM predicts.

[3] Alternatively, one can maintain an separate Equation 2 for each initial policy, but this approach is intractable due to infinitely many agents.

there are two pure-strategy Nash equilibria), the model predicts a system outcome that is completely different from the ground truth. To be more specific, the model predicts that the population would quickly flock to use action a_1; however, in 100 simulation runs, the population always converged to use action a_2.

In summary, when applied to population games, the REM can provide inexact predictions on both *speeds* and *outcomes* of Q-learning. Hence, we caution against using this model when examining Q-learning in population games.

5 CONTINUITY EQUATION MODEL

In this section, we present a new model — the continuity equation model (CEM, Equation 14) — which provides an accurate description of Q-learning in population games. The two issues of the REM and the potential heterogeneity of MASs are non-trivial to address. Our approach is inspired by statistical physics, where studying the dynamics of a probability distribution rather than the dynamics of individuals, greatly reduces the degrees-of-freedom involved and simplifies the analysis. In Section 5.1, we highlight the key steps in the development of our model, and in Section 5.2, we discuss its key properties.

5.1 Development of the Model

The key idea underlying our approach is to work with the distribution of Q-values in the population and derive a differential equation that describes the temporal evolution of this distribution. By working with the distribution of Q-values, we represent agents with their Q-values and the issue caused by representing Q-learners with their policies disappears. In addition, we address the asynchronous update of Q-values in our model by capturing its stochastic effect on Q-values.

Let $M_n(\vec{q}, t)$ where $\vec{q} = [q_1, q_2]^\top \in \mathbb{R}^2$ be the empirical cumulative distribution function (CDF) of the Q-values in the population at time t, i.e. $M_n(\vec{q}, t) = \frac{1}{n}\sum_{i\in\mathcal{N}} \mathbb{1}(Q^i_{1,t} \le q_1, Q^i_{2,t} \le q_2)$ where $\mathbb{1}(\cdot)$ is the indicator function. With a slight abuse of notation, let $\vec{Q}_t = [Q_{1,t}, Q_{2,t}]^\top \in \mathbb{R}^2$ be a pair of random variables denoting the Q-values of an agent that is randomly drawn from the population at time t. We define $f(\vec{q}, t)$ as the probability density function (PDF) for $\vec{Q}_t$ such that the corresponding CDF $F(\vec{q}, t)$ is the asymptotic distribution of the empirical CDF $M_n(\vec{q}, t)$. That is, $f(\vec{q}, t) = \frac{dF(\vec{q},t)}{d\vec{q}}$ such that $M_n(\vec{q}, t) \xrightarrow{\mathcal{D}} F(\vec{q}, t)$.

We are interested in the time evolution of the PDF $f(\vec{q}, t)$. Let $\theta(\vec{q})$ be a test function of Q-values and $\delta \in (0, 1]$ be the amount of time that passes between two repetitions of the population game. We compute the quantity Y defined as

$$Y = \frac{\mathbb{E}[\theta(\vec{Q}_{t+\delta})] - \mathbb{E}[\theta(\vec{Q}_t)]}{\delta} = \int \theta(\vec{q})\frac{f(\vec{q}, t+\delta) - f(\vec{q}, t)}{\delta} d\vec{q}. \tag{5}$$

Intuitively, Y tracks the change of the expected value of $\theta(\vec{Q}_t)$ between two repetitions of the population game, where the PDF $f(\vec{q}, t)$ and $f(\vec{q}, t+\delta)$ are generally different after the game play at time t.

At time t, for an arbitrary agent, let $\vec{Z}_t = [Z_{1,t}, Z_{2,t}]^\top \in \{0, 1\}^2$ be a pair of random variables indicating the action applied at time t such that $Z_{j,t} = 1$ means action a_j is applied and $Z_{j,t} = 0$ means action a_j is not applied. Note that the probability of applying each action is determined by an agent's current Q-values and the exploration mechanism it uses. We define such probability as $p_j(\vec{q})$ for each action a_j. As such, $Z_{j,t} \sim \text{Bernoulli}\left(p_j(\vec{q})\right)$. By the update rule of Q-learning,

$$\vec{Q}_{t+\delta} = \vec{Q}_t + \delta\alpha\vec{Z}_t \cdot (\vec{r}_t - \vec{Q}_t) \tag{6}$$

where $\vec{r}_t = [r_{1,t}, r_{2,t}]^\top$ represents the immediate reward of taking each action and is given by the reward function of the population game. Let $\beta = \delta\alpha$. Based on this equation,

$$\begin{aligned}\mathbb{E}[\theta(\vec{Q}_{t+\delta})] &= \mathbb{E}\left[\theta(\vec{Q}_t + \beta\vec{Z}_t \cdot (\vec{r}_t - \vec{Q}_t))\right] \\ &= \int f(\vec{q}, t) \sum_{j\in\{1,2\}} p_j(\vec{q})\theta(\vec{q} + \beta\vec{e}_j \cdot (\vec{r}_t - \vec{q}))d\vec{q}\end{aligned} \tag{7}$$

where $\vec{e}_j$ is the unit vector such that $\vec{e}_1 = [1, 0]^\top$ and $\vec{e}_2 = [0, 1]^\top$. The Taylor series for $\theta(\vec{q} + \beta\vec{e}_j \cdot (\vec{r}_t - \vec{q}))$ at $\vec{q}$ is

$$\begin{aligned}&\theta(\vec{q}) + \left[\beta\vec{e}_j \cdot (\vec{r}_t - \vec{q})\right]\partial_{q_j}\theta(\vec{q}) + \frac{1}{2}\left[\beta\vec{e}_j \cdot (\vec{r}_t - \vec{q})\right]^2 \partial_{q_jq_j}\theta(\vec{q}) \\ &\quad + o(\left[\beta\vec{e}_j \cdot (\vec{r}_t - \vec{q})\right]^2).\end{aligned} \tag{8}$$

Rearranging terms, we obtain

$$\begin{aligned}\mathbb{E}[\theta(\vec{Q}_{t+\delta})] &= \int f(\vec{q}, t)\theta(\vec{q})d\vec{q} \\ &+ \beta\int f(\vec{q}, t) \sum_{j\in\{1,2\}} p_j(\vec{q})\left[\vec{e}_j \cdot (\vec{r}_t - \vec{q})\right]\partial_{q_j}\theta(\vec{q})d\vec{q} \\ &+ \frac{\beta^2}{2}\int f(\vec{q}, t) \sum_{j\in\{1,2\}} p_j(\vec{q})\left[\vec{e}_j \cdot (\vec{r}_t - \vec{q})\right]^2 \partial_{q_jq_j}\theta(\vec{q})d\vec{q} \\ &+ \beta^2\int f(\vec{q}, t) \sum_{j\in\{1,2\}} p_j(\vec{q})o(\left[\vec{e}_j \cdot (\vec{r}_t - \vec{q})\right]^2)d\vec{q}.\end{aligned} \tag{9}$$

The first term on the right hand side equals $\mathbb{E}[\theta(\vec{Q}_t)]$. Moving the first term to the left hand side and dividing both sides by δ, we have the quantity of interest

$$\begin{aligned}Y = \alpha &\int f(\vec{q}, t) \sum_{j\in\{1,2\}} p_j(\vec{q})[\vec{e}_j \cdot (\vec{r}_t - \vec{q})]\partial_{q_j}\theta(\vec{q})d\vec{q} \\ &+ \frac{\alpha^2\delta}{2}\int f(\vec{q}, t) \sum_{j\in\{1,2\}} p_j(\vec{q})\left[\vec{e}_j \cdot (\vec{r}_t - \vec{q})\right]^2 \partial_{q_jq_j}\theta(\vec{q})d\vec{q} \\ &+ \alpha^2\delta\int f(\vec{q}, t) \sum_{j\in\{1,2\}} p_j(\vec{q})o(\left[\vec{e}_j \cdot (\vec{r}_t - \vec{q})\right]^2)d\vec{q}.\end{aligned} \tag{10}$$

Taking the limit of Y with $\delta \to 0$ (assuming the continuous time limit), the contribution of the second and third terms on the right hand side vanishes.

On the other hand, according to the definition of Y,

$$\begin{aligned}\lim_{\delta\to 0} Y &= \lim_{\delta\to 0}\int \theta(\vec{q})\frac{f(\vec{q}, t+\delta) - f(\vec{q}, t)}{\delta}d\vec{q} \\ &= \int \theta(\vec{q})\partial_t f(\vec{q}, t)d\vec{q}.\end{aligned} \tag{11}$$

Combining Equations 10 and 11 yields

$$\int \theta(\vec{q})\partial_t f(\vec{q},t)d\vec{q} \\ = \alpha \int f(\vec{q},t) \sum_{j\in\{1,2\}} p_j(\vec{q})[\vec{e}_j \cdot (\vec{r}_t - \vec{q})]\partial_{q_j}\theta(\vec{q})d\vec{q}. \tag{12}$$

Using integration by parts, for a typical PDF such that $f(\vec{q},t)$ approaches 0 as $q_1, q_2 \to \pm\infty$, we have

$$\int \theta(\vec{q})\partial_t f(\vec{q},t)d\vec{q} \\ = -\alpha \int \theta(\vec{q}) \sum_{j\in\{1,2\}} \partial_{q_j}\left[f(\vec{q},t)p_j(\vec{q})\left[\vec{e}_j \cdot (\vec{r}_t - \vec{q})\right]\right]d\vec{q}. \tag{13}$$

Note that this equation holds for any test function $\theta(\vec{q})$. From this, we obtain our key result — the *continuity equation model* (CEM) — as follows:

$$\partial_t f(\vec{q},t) + \alpha \sum_{j\in\{1,2\}} \partial_{q_j}\left[f(\vec{q},t)p_j(\vec{q})(R(a_j,\vec{o}_t,t) - q_j)\right] = 0 \\ \text{s.t.} \quad o_{j,t} = \mathbb{E}[p_j(\vec{Q}_t)] \tag{14}$$

where p_j is the probability of using action a_j given by the Q-values and the applied exploration mechanism, and $R(a_j,\vec{o}_t,t)$ is the reward function of the population game. This equation describes the temporal evolution of the PDF $f(\vec{q},t)$. Recall that the value of $f(\vec{q},t)$ at a given point $\vec{q}$ is asymptotically the fraction of agents having their Q-values equal to $\vec{q}$ in the population at time t. Thus, this equation expresses over time how a system of Q-learners concentrates on each possible pair of Q-values $\vec{q}$ in the Q-value space $\mathbb{R}^2$ during their repeated plays of population games.

5.2 Key Properties of the Model

Formal connections to continuity equations. We recognize that Equation 14 can be viewed as a *continuity equation* well known in physics. To see this, let $\rho = f(\vec{q},t)$ and $\vec{v} = [v_1, v_2]^\top$ such that $v_j = \alpha p_j(\vec{q})\left(R(a_j,\vec{o}_t,t) - q_j\right), \forall j \in \{1,2\}$, and we recover the continuity equation

$$\partial_t \rho + \sum_{j\in\{1,2\}} \partial_{q_j}(\rho v_j) = 0. \tag{15}$$

A continuity equation describes the transport of some quantity, such as mass and energy, in a physical system. In the equation, ρ is the density of the quantity and $\vec{v}$ is the velocity field for that quantity. Thus, this suggests a physical interpretation of our model — the Q-learning dynamics in population games is analogously the transport of the agent mass in the m-dimensional Q-value space (where m is the number of actions) such that the velocity of the agent mass is given by $\alpha p_j(\vec{q})(R(a_j,\vec{o}_t,t) - q_j)$ in each direction. The essence of the continuity equation is a local form of conservation law, indicating that over time, the agent mass can neither be created nor destroyed. In addition, the agent mass moves in a continuous flow and cannot "teleport" from one position in the Q-value space to another.

Applicability to different exploration mechanisms. Balancing the exploitation-exploration trade-off is challenging in MAL and it has been shown that the choice of exploration mechanism has a significant impact. Our model wraps the probability of taking each action into a term $p_j(\vec{q})$ in Equation 14. As such, the CEM can describe the dynamics of Q-learning with different exploration mechanisms simply by instantiating $p_j(\vec{q})$. Due to space constraints, we illustrate this in Section 4 of the supplementary by contrasting the Boltzmann exploration and the power probability form [4] common in behavioral economics.

Relations to mean dynamics. Equation 14 can also be used to investigate the dynamics of the population state $\vec{o}_t$. The dynamics of $\vec{o}_t$ corresponds to the conventional definition of *mean dynamics* in evolutionary game theory [30]. For each action $a_j \in \mathcal{A}$, the time derivative of $o_{j,t}$ (as derived in the supplementary) is

$$\frac{do_{j,t}}{dt} = \alpha \sum_{j\in\{1,2\}} \int f(\vec{q},t)p_j(\vec{q})(R(a_j,\vec{o}_t,t) - q_j)\partial_{q_j}p_j(\vec{q})d\vec{q}. \tag{16}$$

Similarly, we also have the dynamics of the mean Q-value

$$\frac{d\mathbb{E}[Q_{j,t}]}{dt} = \alpha \int f(\vec{q},t)p_j(\vec{q})\left(R(a_j,\vec{o}_t,t) - q_j\right)d\vec{q}. \tag{17}$$

Note that the typical approach [30] of analyzing the mean dynamics in evolutionary game theory *cannot* be applied here because the mean dynamics has an explicit dependence on the PDF $f(\vec{q},t)$.

Reducibility to ODEs for homogeneous populations. Consider a homogeneous population where agents have the same initial Q-values $\vec{Q}_0^*$, i.e. $\vec{Q}_0^i = \vec{Q}_0^*, \forall i \in \mathcal{N}$. The PDF $f(\vec{q},t)$ at time $t = 0$ can be represented by a Dirac delta function, i.e. $f_0 = \delta(\vec{Q}_0^*)$. According to Equation 14, the probability density will remain concentrated on a single point at the next time step and beyond. Let $Q_{j,t}^*$ denote the Q-value of action a_j (the same for every agent) at time t. Based on Equation 17, we obtain the dynamics of the Q-value

$$\frac{dQ_{j,t}^*}{dt} = \frac{d\mathbb{E}[Q_{j,t}]}{dt} = \alpha p_j(\vec{Q}_t^*)(R(a_j,\vec{o}_t,t) - Q_{j,t}^*) \tag{18}$$

where $o_{j,t} = p_j(\vec{Q}_t^*)$. We observe that there is no explicit dependence on the PDF $f(\vec{q},t)$. This suggests that for homogeneous populations, the dynamics of Q-learning in population games can be characterized by a system of coupled ODEs (Equation 18 for each action $a_j \in \mathcal{A}$).

Extension to 2-player & n-player games. A homogeneous agent population can be viewed as an individual agent, since every agent in the homogeneous population has the same initial Q-values and develops the same Q-values afterwards. Hence, Equation 18 can be extended to Q-learning in 2-player or n-player games. Consider a *finite* set $\mathcal{M}$ of agents such that $|\mathcal{M}| = 2$ for 2-player games and $|\mathcal{M}| = n$ for n-player games. To achieve this, for each pair of agent $i \in \mathcal{M}$ and action $a_j \in \mathcal{A}_i$ (where $\mathcal{A}_i$ is the set of actions available to agent i), one can maintain a separate ODE

$$\frac{dQ_{j,t}^i}{dt} = \alpha p_j(\vec{Q}_t^i)(r_{j,t}^i - Q_{j,t}^i) \tag{19}$$

where $r_{j,t}^i$ is the reward of agent i that takes action a_j in the 2-player or n-player games at time t. Compared with Equation 18, the main difference is the substitution of reward functions. Note that the dynamics described by this equation and the REM (Equation 2) are not equivalent due to the reasons explained in Section 4.2. We

Game	Available Actions	Reward Function	Remarks
Public Goods	Cooperate (a_1)	$1.5 \times o_{1,t} - 0.5$	unique NE:
	Defect (a_2)	$1.5 \times o_{1,t}$	all defects
Product Choice	Mac (a_1)	$0.5 + o_{1,t}$	two pure-strategy NE:
	Windows (a_2)	$1 - 1.5 \times o_{1,t}$	all choose Mac or all choose Windows
Time-Varying Product Choice	Mac (a_1)	$0.5 + o_{1,t}$	two pure-strategy NE:
	Windows (a_2)	$1 - 1.5 \times o_{1,t}$ if $t \in [0, 10]$, $1.5 - o_{1,t}$ else	all choose Mac or all choose Windows
El Farol Bar	Stay Home (a_1)	0	numerous pure-strategy NE:
	Go to the Bar (a_2)	1 if $o_{2,t} \in [0, 0.6)$, -1 else	exactly 60% of agents go to the bar

Table 1: Summary of the game configurations considered in this work. NE is short for Nash equilibrium. $\vec{o}_t = [o_{1,t}, o_{2,t}]^\top$ is the population state at time t where $o_{1,t}$ and $o_{2,t}$ represent the proportions of agents that use actions a_1 and a_2, respectively, at time t.

leave the study of Q-learning dynamics in 2-player and n-player games using Equation 19 to future work.

Generalization of the 2-player FPE model. The continuity equation model is a generalization of the Fokker-Planck equation model [16] beyond 2-player games and Boltzmann exploration. The setting in [16] considers agents that each play a 2-player symmetric game with *every* other agent in a large population. Our key observation is that for each 2-player symmetric game in their setting, there exists a reward-equivalent population game. Let $U(a_1, a_2)$ be the reward function of a 2-player-2-action symmetric game. For any agent i in the population $\mathcal{N}$, at time t, its reward for using action a_1 in the 2-player games with every other agent j in the population is

$$\sum_{j \in \mathcal{N} \setminus \{i\}} \sum_{k \in \{1,2\}} \mathbb{1}(j \text{ uses } a_k) U(a_1, a_k) = \frac{1}{n-1} \sum_{k \in \{1,2\}} U(a_1, a_k) o_{k,t} \tag{20}$$

where $n = |\mathcal{N}|$ and $\mathbb{1}(\cdot)$ is the indicator function. The left hand side is the agent's reward in the 2-player games, and the right hand side corresponds to the reward-equivalent population game. Therefore, our model can describe the learning dynamics of each 2-player symmetric game in the setting of [16].

6 EXPERIMENTS

In this section, we first validate that our CEM indeed provides a more accurate description of the learning dynamics in population games, compared with the REM. Then, through a concrete example, we show that our CEM can provide insights on the effects of algorithm parameters, which potentially guides parameter tuning.

6.1 Manifesting Nontrivial Temporal Dynamics

For MAL, the learning dynamics can be far from trivial even in a simple 2-player matrix game [3, 25, 31]. However, with an accurate formal model, the temporal evolution of a MAS manifests itself. To validate the descriptive power of our CEM, we considered three typical types of population games: product choice games, public goods games, and the El Farol bar problems. The game configurations are summarized in Table 1. Due to space restrictions, here we focus on the product choice games. The results of the other games are presented in Section 3 of the supplementary. Unless otherwise stated, the Boltzmann temperature is 3 and the learning rate is 0.05.

The product choice games model the *network effect* phenomena commonly observed in economics. When the network effect is present, the value of a product monotonically increases in the number of its users; however, for the network effect to take hold, the number of users needs to reach a critical mass. Here, the critical mass for action a_1 is 20% of agents in the population. We compared our CEM and the REM in terms of which model provides a more accurate description of the population state $\vec{o}_t$ over time, given different settings of initial Q-value distributions.

Figure 2 clearly shows that our CEM indeed has better descriptive power across all the considered settings. In particular, for the second setting in which around 25% of agents takes action a_1 at time $t = 0$, the network effect phenomena suggests that the population share of action a_1 should increase since the critical mass for action a_1 has been reached. However, our CEM accurately captures a somewhat surprising phenomena — although the critical mass for action a_1 was reached at time $t = 0$, the population share of a_1 first experienced a decreasing trend for around 50 time steps and gradually increased thereafter. Such interesting phenomena, unfortunately, was not captured by the REM, which predicts that the population quickly flocks to the use of action a_1.

6.2 Shedding Light on Algorithm Parameters

Inspired by Leonardo et al.'s work [22, 23], we utilize our CEM to investigate the exploration parameter (the Boltzmann temperature τ), which balances the exploitation-exploration trade-off in Q-learning. Traditionally, finding an appropriate exploration parameter is cumbersome and requires many simulation runs. With our CEM, the effects of the parameter on the long-term learning behavior are readily observable and thus, one can draw insights into appropriate choices of the parameter for a desired learning behavior.

As a concrete example, we consider homogeneous populations that play the product choice game. For homogeneous populations, our CEM can be reduced to coupled ODEs (Equation 18). Figure 3 visualizes the solution to these equations, given different choices of temperature τ. The plotted slope fields illustrate the long-term learning process — a homogeneous population starting with a given pair of Q-values (i.e., at a given point in the field) will adapt its Q-values following the trajectory specified by the field. Using the slope fields obtained by our CEM, we can readily observe the effects of τ on the fixed point (or steady state) to which a population converges and the manner by which the fixed point is reached.

Let us consider the populations that start with negative Q-values of two actions (i.e. the left bottom corner of the plots). We observe that in general, as τ increases, the fixed point moves towards the direction of a higher Q-value (Q_1) of action a_1 and a lower Q-value (Q_2) of action a_2, suggesting that the population will stabilize with a larger population share of action a_1. When τ is not larger than 1, the Q-values of two actions increase almost linearly to reach the fixed point. However, when τ becomes larger, the ways by which a population reaches the fixed point change drastically. For the

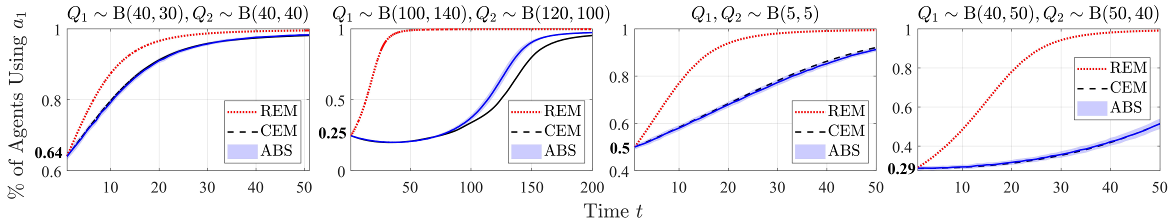


Figure 2: Comparison of the population share $o_{1,t}$ of playing action a_1 in the product choice game for heterogeneous MASs. $\mathbb{B}$ denotes the Beta distribution with support $[-1.5, 1.5]$. Our CEM better captures the qualitative and quantitative dynamics of populations across different initial Q-value distributions. In particular, as shown in the second plot, our CEM captures a somewhat surprising phenomena — although the critical mass for action a_1 was reached initially, the population did not behave exactly the same as predicted by the network effect; rather, the population exhibited a decreasing trend in the use of action a_1 for the first 50 time steps.

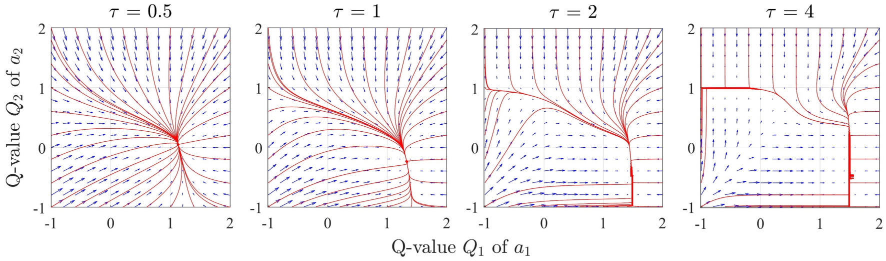


Figure 3: Slope field plots of the Q-value dynamics in homogeneous MASs that play the product choice game. The arrows give the slope dQ_2/dQ_1, the red lines highlight the trajectories, and the points where the red lines converge are the fixed points ($dQ_2/dQ_1 = 0$). Our CEM shows how the Boltzmann temperature τ affects the position of the fixed point and the manner by which it is reached. In particular, our CEM suggests that a high temperature τ causes the phenomena shown in the second plot of Figure 2.

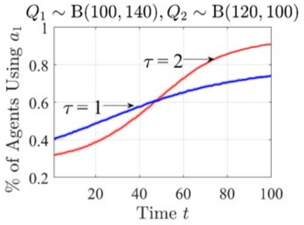


Figure 4: Simulation results verify CEM's prediction that decreasing the temperature τ makes the phenomena shown in the second plot of Figure 2 disappear.

population that starts with a higher Q_1 (i.e. below the diagonal), Q_1 surges directly to reach the fixed point. In contrast, for populations that start with a higher Q_2 (i.e. above the diagonal), Q_2 initially surges but gradually decreases to the fixed point.

The above observations lead to the following insights on the choice of Boltzmann temperature τ in the product choice games: (i) a higher temperature should in general lead to more agents eventually using action a_1, (ii) with a sufficiently low temperature (e.g. $\tau \leq 1$), the Q-values and the policy of agents should quickly become stable, and (iii) with a sufficiently high temperature (e.g. $\tau > 1$), the populations that start with a higher Q-value of action a_2 stick to using a_2 for a significant period of time before finally converging to use action a_1.

We find that these insights not only directly apply to homogeneous populations, but also potentially guide parameter tuning for the more general heterogeneous populations. In particular, the last insight suggests a cause of the phenomena that we observed in the second plot of Figure 2: the high Boltzmann temperature. To validate this, we decreased the temperature parameter in agent-based simulations. As shown in Figure 4, given the same initial Q-value distribution and learning rate as in the second plot of Figure 2, with a lower temperature, the population share of action a_1 increases over time and the phenomena of interest disappears.

7 DISCUSSION

In this paper, we examined the dynamics of Q-learning in population games. We began by pointing out the limitations of the replicator equation model when applied to this setting. As a remedy, we developed our continuity equation model (CEM) and analyzed its key properties. We provided extensive numerical validation for the descriptive power of our model and also illustrated two use cases.

In general, our model works well for a sufficiently large agent population (e.g. consisting of at least hundreds of agents) with a continuously differentiable probability density function (PDF) of Q-values. However, our model may be inaccurate when (i) the agent population is small (e.g. consisting of only dozens of agents), and (ii) the PDF of Q-values is not smooth. Nevertheless, we believe our model is widely applicable; the assumption of a large agent population is standard as population games (by default) are frameworks for large-scale MASs. Regarding the PDF of Q-values, many common probability distributions (e.g. Beta and normal) enjoy the smoothness property.

We believe that our CEM is an important step towards more general models; as future work, it would be interesting to consider stateful population games, variants of Q-learning, and population games with multiple populations (e.g., with different kinds of agents including human models [35]) or network structure. We hope that our work can encourage more work along this line of research.

ACKNOWLEDGMENTS

This research is supported by the National Research Foundation Singapore under its AI Singapore Programme (Award Number: AISG-RP-2019-011). The work described in this paper was partially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CUHK 14206820).

REFERENCES

[1] James P Bailey and Georgios Piliouras. 2019. Multi-agent learning in network zero-sum games is a Hamiltonian system. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 233–241.

[2] Daan Bloembergen, Karl Tuyls, Daniel Hennes, and Michael Kaisers. 2015. Evolutionary dynamics of multi-agent learning: A survey. *Journal of Artificial Intelligence Research* 53 (2015), 659–697.

[3] Victor Boone and Georgios Piliouras. 2019. From Darwin to Poincaré and von Neumann: Recurrence and cycles in evolutionary and algorithmic game theory. In *International Conference on Web and Internet Economics*. Springer, 85–99.

[4] Colin Camerer and Teck Hua Ho. 1999. Experience-weighted attraction learning in normal form games. *Econometrica* 67, 4 (1999), 827–874.

[5] Yun Kuen Cheung. 2018. Multiplicative Weights Updates with Constant Step-Size in Graphical Constant-Sum Games. In *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Eds.), Vol. 31. 3528–3538.

[6] Drew Fudenberg, Fudenberg Drew, David K Levine, and David K Levine. 1998. *The theory of learning in games*. Vol. 2. MIT press.

[7] Aram Galstyan. 2013. Continuous strategy replicator dynamics for multi-agent Q-learning. *Autonomous agents and multi-agent systems* 26, 1 (2013), 37–53.

[8] Nicola Gatti, Fabio Panozzo, and Marcello Restelli. 2013. Efficient evolutionary dynamics with extensive-form games. In *Twenty-Seventh AAAI Conference on Artificial Intelligence*.

[9] Xin Guo, Anran Hu, Renyuan Xu, and Junzi Zhang. 2019. Learning mean-field games. In *Advances in Neural Information Processing Systems*. 4967–4977.

[10] Hado V Hasselt. 2010. Double Q-learning. In *Advances in Neural Information Processing Systems*. 2613–2621.

[11] Daniel Hennes, Dustin Morrill, Shayegan Omidshafiei, Rémi Munos, Julien Perolat, Marc Lanctot, Audrunas Gruslys, Jean-Baptiste Lespiau, Paavo Parmas, Edgar Duéñez-Guzmán, et al. 2020. Neural replicator dynamics: Multiagent learning via hedging policy gradients. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*. 492–501.

[12] Daniel Hennes, Karl Tuyls, and Matthias Rauterberg. 2009. State-coupled replicator dynamics. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*. 789–796.

[13] Junling Hu and Michael P Wellman. 2003. Nash Q-learning for general-sum stochastic games. *Journal of machine learning research* 4, Nov (2003), 1039–1069.

[14] Shuyue Hu. 2022. The Dynamics of Q-learning in Population Games: Supplementary Material. (Jan 2022). http://sites.google.com/view/shuyue-hu

[15] Shuyue Hu. 2022. The Dynamics of Q-learning in Population Games: Supplementary Material. (Jan 2022). https://clear-nus.github.io/papers/CEMsupp.pdf

[16] Shuyue Hu, Chin-wing Leung, and Ho-fung Leung. 2019. Modelling the Dynamics of Multiagent Q-Learning in Repeated Symmetric Games: a Mean Field Theoretic Approach. In *Advances in Neural Information Processing Systems*. 12102–12112.

[17] Michael Kaisers, Daan Bloembergen, and Karl Tuyls. 2012. A common gradient in multi-agent reinforcement learning. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 3*. International Foundation for Autonomous Agents and Multiagent Systems, 1393–1394.

[18] Michael Kaisers and Karl Tuyls. 2010. Frequency adjusted multi-agent Q-learning. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1*. International Foundation for Autonomous Agents and Multiagent Systems, 309–316.

[19] Ardeshir Kianercy and Aram Galstyan. 2012. Dynamics of Boltzmann Q learning in two-player two-action games. *Physical Review E* 85, 4 (2012), 041145.

[20] Ratul Lahkar and Robert M Seymour. 2013. Reinforcement learning in population games. *Games and Economic Behavior* 80 (2013), 10–38.

[21] Jean-Michel Lasry and Pierre-Louis Lions. 2007. Mean field games. *Japanese journal of mathematics* 2, 1 (2007), 229–260.

[22] Stefanos Leonardos and Georgios Piliouras. 2021. Exploration-Exploitation in Multi-Agent Learning: Catastrophe Theory Meets Game Theory. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 11263–11271.

[23] Stefanos Leonardos, Iosif Sakos, Costas Courcoubetis, and Georgios Piliouras. 2020. Catastrophe by Design in Population Games: Destabilizing Wasteful Locked-In Technologies. In *Proceedings of the 16th International Conference on Web and Internet Economics*. 7–11.

[24] Pinxin Long, Tingxiang Fanl, Xinyi Liao, Wenxi Liu, Hao Zhang, and Jia Pan. 2018. Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 6252–6259.

[25] Sai Ganesh Nagarajan, David Balduzzi, and Georgios Piliouras. 2020. From chaos to order: Symmetry and conservation laws in game dynamics. In *International Conference on Machine Learning*. PMLR, 7186–7196.

[26] Fabio Panozzo, Nicola Gatti, and Marcello Restelli. 2014. Evolutionary dynamics of Q-learning over the sequence form. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 28.

[27] Eduardo Rodrigues Gomes and Ryszard Kowalczyk. 2009. Dynamic analysis of multiagent Q-learning with ε-greedy exploration. In *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, 369–376.

[28] Ahmad EL Sallab, Mohammed Abdou, Etienne Perot, and Senthil Yogamani. 2017. Deep reinforcement learning framework for autonomous driving. *Electronic Imaging* 2017, 19 (2017), 70–76.

[29] William H Sandholm. 2010. *Population games and evolutionary dynamics*. MIT press.

[30] William H Sandholm. 2015. Population games and deterministic evolutionary dynamics. In *Handbook of game theory with economic applications*. Vol. 4. Elsevier, 703–778.

[31] Yuzuru Sato, Eizo Akiyama, and J Doyne Farmer. 2002. Chaos in learning a simple two-person game. *Proceedings of the National Academy of Sciences* 99, 7 (2002), 4748–4751.

[32] Yuzuru Sato and James P Crutchfield. 2003. Coupled replicator equations for the dynamics of learning in multiagent systems. *Physical Review E* 67, 1 (2003), 015206.

[33] Jayakumar Subramanian and Aditya Mahajan. 2019. Reinforcement learning in stationary mean-field games. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*. 251–259.

[34] Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction (Chapter 6.4: SARSA)*. MIT press.

[35] Nicholas Teh, Shuyue Hu, and Harold Soh. 2021. A Theoretical Framework for Large-Scale Human-Robot Interaction with Groups of Learning Agents. In *Companion of the 2021 ACM/IEEE International Conference on Human-Robot Interaction (HRI '21)*. Association for Computing Machinery, New York, NY, USA.

[36] Karl Tuyls and Simon Parsons. 2007. What evolutionary game theory tells us about multiagent learning. *Artificial Intelligence* 171, 7 (2007), 406–416.

[37] Karl Tuyls, Katja Verbeeck, and Tom Lenaerts. 2003. A selection-mutation model for q-learning in multi-agent systems. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*. ACM, 693–700.

[38] Emmanouil-Vasileios Vlatakis-Gkaragkounis, Lampros Flokas, and Georgios Piliouras. 2019. Poincaré recurrence, cycles and spurious equilibria in gradient-descent-ascent for non-convex non-concave zero-sum games. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019*.

[39] Jun Wang, Weinan Zhang, and Shuai Yuan. 2016. Display advertising with real-time bidding (RTB) and behavioural targeting. *arXiv preprint arXiv:1610.03013* (2016).

[40] Christopher JCH Watkins and Peter Dayan. 1992. Q-learning. *Machine Learning* 8, 3-4 (1992).

[41] Michael Wunder, Michael L Littman, and Monica Babes. 2010. Classes of multiagent q-learning dynamics with epsilon-greedy exploration. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*. Citeseer, 1167–1174.

[42] Yaodong Yang, Rui Luo, Minne Li, Ming Zhou, Weinan Zhang, and Jun Wang. 2018. Mean Field Multi-Agent Reinforcement Learning. In *International Conference on Machine Learning*. 5567–5576.