2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

Asynchronous Decentralized Q-Learning in Stochastic Games

Bora Yongacoglu, Gürdal Arslan, and Serdar Yüksel

***Abstract*— Non-stationarity is a fundamental challenge in multi-agent reinforcement learning (MARL), where agents update their behaviour as they learn. In multi-agent settings, individual agents may have an incomplete view of the actions of others, which can complicate the learning process. Many theoretical advances in MARL avoid the challenge of non-stationarity by coordinating the policy updates of agents in various ways, including synchronizing times at which agents are allowed to revise their policies. In this paper, we study an asynchronous variant of the decentralized Q-learning algorithm, a recent MARL algorithm for stochastic games. We provide sufficient conditions under which the asynchronous algorithm drives play to equilibrium with high probability. In this generalization, players need not agree on the schedule of policy update times, and may change their policies at their own separately selected times. This work extends the applicability of the decentralized Q-learning algorithm to settings in which parameters are selected in an independent manner, and tames non-stationarity without imposing the coordination assumptions of prior work.**

I. INTRODUCTION

Multi-agent systems are characterized by the coexistence of many autonomous agents in a shared environment. In multi-agent reinforcement learning (MARL), agents in the system change their behaviour in response to feedback received after previous interactions. The system is therefore non-stationary from any one agent's perspective, and agents attempt to optimize their performance against a moving target [1]. The non-stationarity of MARL environments has been identified as one of the fundamental problems in MARL [2]. In contrast to the rich literature on single-agent learning theory, the theory of MARL is relatively underdeveloped, due in large part to its inherent challenges of non-stationarity and decentralized information.

This paper studies learning algorithms for stochastic games, a common framework for MARL in which the cost-relevant history of the system is summarized by a state variable. In this paper, we focus on stochastic games in which each agent fully observes the system's state variable but does not observe the actions of other agents, exacerbating the challenge of non-stationarity.

Early theoretical work on MARL in stochastic games avoided the problem of non-stationarity by studying applications in which joint actions were observed by all agents [3], [4], [5]. More recently, there has been interest in the *individual action learner* setting, where actions are not shared between agents. In the individual action learner setting, several rigorous contributions have been made recently, including [6], [7], [8], [9], [10], to be discussed shortly.

Of the recent theoretical advances in MARL in the individual action learner setting, many of the algorithms with strong guarantees have circumvented the challenge of non-stationarity by relying, implicitly or explicitly, on coordination between the agents. In particular, several algorithms rely on some form of synchrony, whereby agents agree on the times at which they may revise their behaviour and are constrained to fix their policies at other times. While this is justifiable in some settings, it can be restrictive in others, including applications where parameters are selected independently. As such, it would be desirable to provide MARL algorithms that do not require synchrony but still come with rigorous performance guarantees in the individual action learner setting.

Contributions: In this paper, we study a modification of the *decentralized Q-learning* algorithm of [6], a recent algorithm proposed for weakly acyclic N-player stochastic games. By employing a constant learning rate in the Q-learning algorithm, we show that inertial best-response dynamics provide a mechanism for taming non-stationarity without coordinating players' parameter choices ahead of play.[1] Under appropriate parameter selection, we show that this algorithm drives policies to equilibrium with arbitrarily high probability.

Notation: For a standard Borel space A, we let $\mathcal{P}(A)$ denote the set of probability measures on A with its Borel σ-algebra. For standard Borel spaces A and B, we let $\mathcal{P}(A|B)$ denote the set of transition kernels on A given B.

A. Related Work

This paper studies stochastic games in which each agent fully observes the system state but does not observe the actions of other players.[2] As such, we are interested in MARL algorithms that make use only of one's history of state observations, *individual* actions, and cost feedback. This information paradigm is common in the literature in MARL, with examples such as [6], [7], [8], [9], [10], [12], [13], and is sometimes called the independent learning paradigm. This terminology is, however, not uniform, as independent learning has also recently been used to refer to learners that update their policies in a (perhaps myopic) self-interested manner [14].

B. Yongacoglu is with the Department of Electrical and Computer Engineering, University of Toronto. G. Arslan is with the Department of Electrical Engineering, University of Hawaii at Manoa. S. Yüksel is with the Department of Mathematics and Statistics, Queen's University. Correspondence to bora.yongacoglu@utoronto.ca.

[1] Due to space constraints, proofs have been omitted and can be found in the full version of this paper, [11].

[2] We use the terms players and agents interchangeably.

979-8-3503-0123-6/23/$31.00 ©2023 IEEE

At least two challenges emerge when the joint action is not observed. First, agents cannot form estimates about the policies of other agents. In extreme examples, players may be unaware of the very existence of their counterparts. Second, players cannot form estimates about joint action values, yielding several promising joint action learning algorithms unusable.

Rather than estimating a global joint action Q-function, several works have studied the prospect of greedily changing one's policy, either in a best-response sense, using a local action Q-function, or in an iterative gradient descent sense. To handle the challenge of non-stationarity, some authors, e.g. [7], have proposed the use of a *multi-timescale approach*, whereby some agents change their policies faster than others, possibly in an alternating manner, [15]. In these works, agreement on a particular schedule for policy updating may be interpreted as an implicit form of parameter coordination.

An alternative approach involves responding to one's environment without accounting for the existence of other players *at all*. Works in this tradition follow the regret testing paradigm of Foster and Young [16], which presented an algorithm for stateless games. This approach was later studied by [17] and [18] among others in the context of stateless repeated games, where impressive convergence guarantees can be made due to the absence of state dynamics that complicate value estimation.

In [6], the regret testing paradigm of Foster and Young was modified for multi-state stochastic games, where one must account for both the immediate cost of an action and also the cost-to-go, which depends on the evolution of the state variable. The *decentralized Q-learning* algorithm of [6] instructs agents to agree on an increasing sequence of policy update times, $(t_k)_{k\geq 0}$, and to fix one's policy within the intervals $[t_k, t_{k+1})$, called the k^{th} *exploration phase*. In so doing, the joint policy process is fixed over each exploration phase, and within each exploration phase, each agent faces an MDP. This allows for analysis of learning iterates using single-agent learning theory.

In effect, the exploration phase technique of [6] decouples learning from adaptation, and allows for separate analysis of learning iterates and policy dynamics. This allows for approximation arguments to be used, whereby the dynamics of the policy process resemble those of an idealized process in which players obtain noise-free learning iterates for use in their policy updates. This has lead to a series of theoretical contributions in MARL that all make use of the exploration phase technique, including [9] and [10].

One natural criticism of the exploration phase technique described above is the synchronization of policy updates. In the description above, agents agree on the policy update times $\{t_k\}_{k\geq 0}$ *exactly*, and no agent ever updates its policy in the interval $(t_k, t_{k+1}-1]$. This can be justified in some settings, but is demanding in decentralized settings where parameters are selected independently across players.[3] Indeed, the assumption of synchrony is made in various works in the regret testing tradition, including [16], [17] and [18].

Intuitively, asynchrony may be problematic for regret testers because the action-value estimates of players depend on historical data from each player's most recent exploration phase. As such, if other players change their policies during an individual's exploration phase, the individual receives feedback from different sources, and its learning iterates may not approximate any quantity relevant to the prevailing environment at the time of the agent's next policy update. These changes of policies during an exploration phase constitute potential disruptions of a player's learning, and analysis of the overall joint policy process is difficult when players do not reliably learn accurate action-values.

In [18], a heuristic argument suggested that the use of inertia in policy updating may allow one to relax the synchrony assumption for regret testers, with the following premise: if players occasionally abstain from changing their policies due to random inertia, then they will abstain from disrupting the learning of other agents. If the exploration phase of a given individual is allowed to proceed for a sufficiently long time without disruptions, then any errors in one's learning estimates may be corrected. In this way, it is argued that random inertia acts as a decentralized coordination mechanism and perfect synchrony may not be necessary. In this paper, we formalize this argument and show that it is essentially correct, with a caveat: our analysis reveals that the value estimation protocol must be modified to account for the non-stationarity in the environment. In particular, the algorithm of this paper uses a constant learning rate to ensure that learning iterates rapidly overcome outdated feedback data.

II. Model

A. Stochastic Games

Formally, a stochastic game $\mathcal{G}$ is described by a list:

$$\mathcal{G} = \left(\mathcal{N}, \mathbb{X}, \{\mathbb{U}^i, c^i, \beta^i : i \in \mathcal{N}\}, P, \nu_0\right). \qquad (1)$$

The components of $\mathcal{G}$ are as follows: $\mathcal{N} = \{1, 2, \dots, N\}$ is a finite set of N agents. The set $\mathbb{X}$ is a finite set of system states. For each player $i \in \mathcal{N}$, $\mathbb{U}^i$ is i's finite set of actions. We write $\mathbf{U} = \times_{i\in\mathcal{N}}\mathbb{U}^i$, and refer to elements of $\mathbf{U}$ as *joint actions*. For each player i, a function $c^i : \mathbb{X} \times \mathbf{U} \to \mathbb{R}$ determines player i's stage costs, which are aggregated using a discount factor $\beta^i \in [0, 1)$. The initial system state has distribution $\nu_0 \in \mathcal{P}(\mathbb{X})$, and state transitions are governed by a transition kernel $P \in \mathcal{P}(\mathbb{X}|\mathbb{X} \times \mathbf{U})$.

At time $t \in \mathbb{Z}_{\geq 0}$, the state variable is denoted by x_t, and each player i selects an action $u^i_t \in \mathbb{U}^i$ according to its policy, to be described shortly. The joint action at time t is denoted $\mathbf{u}_t$. Each player i incurs a cost $c^i_t := c^i(x_t, \mathbf{u}_t)$, and the state variable evolves according to $x_{t+1} \sim P(\cdot|x_t, \mathbf{u}_t)$. This process is then repeated at time $t+1$, and so on.

A policy is a rule for selecting actions according to the observed history of the system. Here, we assume that at time $t \geq 0$, player i observes the following information:

$$I^i_t = (x_0, u^i_0, c^i_0, x_1, \dots, c^i_{t-1}, x_t).$$

[3] The provision of algorithms for such decentralized settings has recently attracted interest from other authors using algorithms similar to—but distinct from—the regret testing tradition. See, for example, [19].

Player i fully observes the system state, its own actions, and its own cost *realizations,* but does not observe the actions of other players directly. We do not assume that player i knows the function c^i.

In general, action selection can incorporate randomness, and players may use arbitrarily complicated, history-dependent policies. However, our analysis will focus on stationary (Markov) policies, a subset of policies that randomly select actions in a time invariant manner that conditions only on the currently observed state variable. The set of stationary policies for player i is denoted Γ^i_S and we identify Γ^i_S with the set of transition kernels on $\mathbb{U}^i$ given $\mathbb{X}$. Henceforth, unqualified reference to a policy shall be understood to mean a stationary policy.

Definition 1: For $i \in \mathcal{N}$, $\xi > 0$, a policy $\pi^i \in \Gamma^i_S$ is called *ξ-soft* if $\pi^i(a^i|x) \geq \xi$ for all $(x, a^i) \in \mathbb{X} \times \mathbb{U}^i$. A policy $\pi^i \in \Gamma^i_S$ is called *soft* if it is ξ-soft for some $\xi > 0$.

Definition 2: A policy $\pi^i \in \Gamma^i_S$ is called *deterministic* if for each $x \in \mathbb{X}$, there exists $a^i \in \mathbb{U}^i$ such that $\pi^i(a^i|x) = 1$.

The set of deterministic stationary policies for player i is denoted by Γ^i_{SD} and is identified with the set of functions from $\mathbb{X}$ to $\mathbb{U}^i$.

Notation: We let $\boldsymbol{\Gamma}_S := \times_{i \in \mathcal{N}} \Gamma^i_S$ denote the set of *joint policies*. To isolate player i's component in a particular joint policy $\boldsymbol{\pi} \in \boldsymbol{\Gamma}_S$, we write $\boldsymbol{\pi} = (\pi^i, \boldsymbol{\pi}^{-i})$, where $-i$ is used in the agent index to represent all agents other than i. Similarly, we write the joint policy set as $\boldsymbol{\Gamma}_S = \Gamma^i_S \times \boldsymbol{\Gamma}^{-i}_S$, and so on.

For any joint policy $\boldsymbol{\pi}$ and initial distribution $\nu \in \mathcal{P}(\mathbb{X})$, there is a unique probability measure on the set $(\mathbb{X} \times \mathbf{U})^\infty$. We denote this measure by $\text{Pr}^{\boldsymbol{\pi}}_\nu$, and let $E^{\boldsymbol{\pi}}_\nu$ denote its expectation. We use this to define player i's value function:

$$J^i(\boldsymbol{\pi}, \nu) := E^{\boldsymbol{\pi}}_\nu \left[\sum_{t=0}^{\infty} \beta^t c^i_t \right] = E^{\boldsymbol{\pi}}_\nu \left[\sum_{t=0}^{\infty} \beta^t c^i(x_t, \mathbf{u}_t) \right].$$

When $\nu = \delta_s$ places full probability on some state $s \in \mathbb{X}$, we write $J^i(\boldsymbol{\pi}, s)$ instead of $J^i(\boldsymbol{\pi}, \delta_s)$. For $\boldsymbol{\pi} = (\pi^i, \boldsymbol{\pi}^{-i})$, we will also write $J^i(\pi^i, \boldsymbol{\pi}^{-i}, \nu)$ to isolate the role of π^i.

Definition 3: Let $\epsilon \geq 0$, $i \in \mathcal{N}$. A policy $\pi^{*i} \in \Gamma^i_S$ is called an *ϵ-best-response to $\boldsymbol{\pi}^{-i}$* $\in \boldsymbol{\Gamma}^{-i}_S$ if, for every $s \in \mathbb{X}$,

$$J^i(\pi^{*i}, \boldsymbol{\pi}^{-i}, s) \leq \inf_{\tilde{\pi}^i \in \Gamma^i_S} J^i(\tilde{\pi}^i, \boldsymbol{\pi}^{-i}, s) + \epsilon.$$

The set of ϵ-best-responses to $\boldsymbol{\pi}^{-i}$ is denoted $\text{BR}^i_\epsilon(\boldsymbol{\pi}^{-i})$. It is well-known that for any $\boldsymbol{\pi}^{-i} \in \boldsymbol{\Gamma}^{-i}_S$, player i's set of 0-best-responses $\text{BR}^i_0(\boldsymbol{\pi}^{-i})$ is non-empty, and the infimum above is in fact attained.

Definition 4: Let $\epsilon \geq 0$. A joint policy $\boldsymbol{\pi}^* \in \boldsymbol{\Gamma}_S$ is called an *ϵ-equilibrium* if $\pi^{*i} \in \text{BR}^i_\epsilon(\boldsymbol{\pi}^{*-i})$ for all $i \in \mathcal{N}$.

For $\epsilon \geq 0$, we let $\boldsymbol{\Gamma}^{\epsilon\text{-eq}}_S \subseteq \boldsymbol{\Gamma}_S$ denote the set of ϵ-equilibrium policies. It is known that the set $\boldsymbol{\Gamma}^{0\text{-eq}}_S$ is non-empty [20]. We also let $\boldsymbol{\Gamma}^{\epsilon\text{-eq}}_{SD} \subset \boldsymbol{\Gamma}_{SD}$ denote the subset of stationary deterministic ϵ-equilibrium policies, which may be empty in general.

B. Weakly Acyclic Stochastic Games

We now introduce weakly acyclic games, an important subclass of games that will be the main focus of this paper.

Definition 5: A sequence $\{\boldsymbol{\pi}_k\}_{k \geq 0}$ in $\boldsymbol{\Gamma}_{SD}$ is called a *strict best-response path* if for any $k \geq 0$ there is a unique player $i \in \mathcal{N}$ such that $\pi^i_{k+1} \neq \pi^i_k$ and $\pi^i_{k+1} \in \text{BR}^i_0(\boldsymbol{\pi}^{-i}_k)$.

Definition 6: The stochastic game $\mathcal{G}$ is *weakly acyclic* if (i) $\boldsymbol{\Gamma}^{0\text{-eq}}_{SD} \neq \varnothing$, and (ii) for any $\boldsymbol{\pi}_0 \in \boldsymbol{\Gamma}_{SD}$, there is a strict best-response path from $\boldsymbol{\pi}_0$ to some $\boldsymbol{\pi}^* \in \boldsymbol{\Gamma}^{0\text{-eq}}_{SD}$.

The multi-state formulation above was stated in [6], though weakly acyclic games had previously been studied in stateless games [21]. An important special case is that of stochastic teams, where $c^i = c^j$ for each i, j, and the interests of all agents are perfectly aligned. Markov potential games, [22], [23], [24] constitute another special case of weakly acyclic games.

C. Q-Functions in Stochastic Games

In the stochastic game $\mathcal{G}$, when the other players use a stationary policy $\boldsymbol{\pi}^{-i} \in \boldsymbol{\Gamma}^{-i}_S$, player i faces an environment that is equivalent to a single-agent MDP. The MDP in question depends on the policy $\boldsymbol{\pi}^{-i}$ as well as the game $\mathcal{G}$, and (stationary Markov) optimal policies for this MDP are equivalent to 0-best-responses to $\boldsymbol{\pi}^{-i}$ in the game $\mathcal{G}$.

Player i's best-responses to a policy $\boldsymbol{\pi}^{-i} \in \boldsymbol{\Gamma}^{-i}_S$ can be characterized using an appropriately defined *Q-function*, $Q^{*i}_{\boldsymbol{\pi}^{-i}} : \mathbb{X} \times \mathbb{U}^i \to \mathbb{R}$.[4] The function $Q^{*i}_{\boldsymbol{\pi}^{-i}}$ can be defined by a fixed point equation of a Bellman operator, but here we give an equivalent definition in terms of the optimal policy of the corresponding MDP:

$$Q^{*i}_{\boldsymbol{\pi}^{-i}}(x, a^i) := E^{\boldsymbol{\pi}^*}_\nu \left[\sum_{t=0}^{\infty} (\beta^i)^t c^i(x_t, \mathbf{u}_t) \middle| x_0 = x, u^i_0 = a^i \right],$$

for all $(x, a^i) \in \mathbb{X} \times \mathbb{U}^i$, where $\boldsymbol{\pi}^* = (\pi^{*i}, \boldsymbol{\pi}^{-i})$ and $\pi^{*i} \in \text{BR}^i_0(\boldsymbol{\pi}^{-i}) \cap \Gamma^i_{SD}$.

Definition 7: For $Q^i : \mathbb{X} \times \mathbb{U}^i \to \mathbb{R}$ and $\epsilon \geq 0$, we define

$$\begin{aligned}\widehat{\text{BR}}^i_\epsilon(Q^i) := \{\pi^{*i} \in \Gamma^i_{SD} : Q^i(x, \pi^{*i}(x)) \\ \leq \min_{a^i \in \mathbb{U}^i} Q^i(x, a^i) + \epsilon, \forall x \in \mathbb{X}\}.\end{aligned}$$

The set $\widehat{\text{BR}}^i_\epsilon(Q^i) \subseteq \Gamma^i_{SD}$ consists of policies that are ϵ-greedy with respect to Q^i. For $Q^i = Q^{*i}_{\boldsymbol{\pi}^{-i}}$, we have $\widehat{\text{BR}}^i_0(Q^{*i}_{\boldsymbol{\pi}^{-i}}) = \text{BR}^i_0(\boldsymbol{\pi}^{-i}) \cap \Gamma^i_{SD}$.

When the remaining players follow a stationary policy, player i can use Q-learning to estimate its action-values, which can then be used to estimate a 0-best-response policy. The situation is more complicated when the remaining players revise their policies over time. Under this non-stationarity, Q-learning may not be guaranteed to converge, and this procedure for estimating a best-response may be ineffective. These issues were considered by [6], who proposed the *Decentralized Q-learning algorithm* as a means of estimating best-response policies in the presence of policy updating, but required synchronized policy updating. In the next section, we present Algorithm 1, a modification of Decentralized Q-learning that allows for decentralized parameter selection and can tolerate non-stationarity of the learning environment.

[4] We use "Q-function" and "action-value function" interchangeably.

III. Asynchronous Decentralized Q-Learning

An asynchronous variant of Decentralized Q-learning is presented in Algorithm 1. Unlike in the original decentralized Q-learning algorithm of [6], Algorithm 1 allows for the sequence of exploration phase lengths $\{T^i_k\}_{k\geq 0}$ to vary by agent, employs constant learning rate, and does not reset Q-factors at the end of an exploration phase.

Algorithm 1: Asynchronous Decentralized Q-Learning

```
Set Parameters
    $\{T^i_k\}_{k\geq 0}$: a sequence in $\mathbb{N}$ of learning phase lengths
        Put $t^i_0 = 0$ and $t^i_{k+1} = t^i_k + T^i_k$ for all $k \geq 0$.
    $\rho^i \in (0,1)$: experimentation probability
    $\lambda^i \in (0,1)$: inertia during policy update
    $\delta^i \in (0,\infty)$: tolerance level for sub-optimality
    $\alpha^i \in (0,1)$: step-size parameter
Initialize $\pi^i_0 \in \Gamma^i_{SD}$ (arbitrary), $\widehat{Q}^i_0 = 0 \in \mathbb{R}^{\mathbb{X}\times\mathbb{U}^i}$
for $k \geq 0$ ($k^{th}$ exploration phase for agent $i$ )
    for $t = t^i_k, t^i_k + 1, \ldots, t^i_{k+1} - 1$
        Observe $x_t$
        Select $u^i_t = \begin{cases} \pi^i_k(x_t), & \text{w.p. } 1-\rho^i \\ u^i \sim \text{Unif}(\mathbb{U}^i), & \text{w.p. } \rho^i \end{cases}$
        Observe cost $c^i_t := c(x_t, \mathbf{u}_t)$, state $x_{t+1}$
        Put $\Delta^i_t = c^i_t + \beta^i \min_{a^i} \widehat{Q}^i_t(x_{t+1}, a^i)$
        $\widehat{Q}^i_{t+1}(x_t, u^i_t) = (1-\alpha^i)\widehat{Q}^i_t(x_t, u^i_t) + \alpha^i \Delta^i_t$
        $\widehat{Q}^i_{t+1}(x, u^i) = \widehat{Q}^i_t(x, u^i)$, for all $(x, u^i) \neq (x_t, u^i_t)$
    if $\pi^i_k \in \widehat{\text{BR}}^i_{\delta^i}(\widehat{Q}^i_{t^i_{k+1}})$, then
        $\pi^i_{k+1} \leftarrow \pi^i_k$
    else
        $\pi^i_{k+1} \leftarrow \begin{cases} \pi^i_k, & \text{w.p. } \lambda^i \\ \pi^i \in \widehat{\text{BR}}^i_{\delta^i}(\widehat{Q}^i_{t^i_{k+1}}), & \text{w.p. } 1-\lambda^i \end{cases}$
```

A. Primitive Random Variables

We now introduce several collections of *primitive* random variables that will be used in describing the assumptions and implementation of Algorithm 1. For any player $i \in \mathcal{N}$ and $t \geq 0$, we define the following random variables:

- $\{W_t\}_{t\geq 0}$ is an identically distributed, $[0,1]$-valued stochastic process. For some $f : \mathbb{X} \times \mathbf{U} \times [0,1] \to \mathbb{X}$, state transitions are driven by $\{W_t\}_{t\geq 0}$ via f:
$$\begin{aligned} &\Pr(x_{t+1} = s' | x_t = s, \mathbf{u}_t = \mathbf{a}) = P(s'|s,\mathbf{a}) \\ =&\Pr\left(W_t \in \{w : f(s,\mathbf{a},w) = s'\}\right), \end{aligned}$$
for any $(s,\mathbf{a},s') \in \mathbb{X} \times \mathbf{U} \times \mathbb{X}$ and $t \geq 0$;
- $\tilde{u}^i_t \sim \text{Unif}(\mathbb{U}^i)$;
- $\tilde{\rho}^i_t \sim \text{Unif}([0,1])$;
- $\tilde{\lambda}^i_t \sim \text{Unif}([0,1])$;
- For non-empty $B^i \subseteq \Gamma^i_{SD}$, $\tilde{\pi}^i_t(B^i) \sim \text{Unif}(B^i)$;
- T^i_t is an $\mathbb{N}$-valued random variable, elaborated below.

Assumption 1: The collection of primitive random variables $\mathcal{V}_1 \cup \mathcal{V}_2$ is mutually independent, where

$$\mathcal{V}_1 := \bigcup_{i\in\mathcal{N}, t\geq 0} \left\{W_t, \tilde{\rho}^i_t, \tilde{u}^i_t, \tilde{\lambda}^i_t, T^i_t\right\}, \quad \text{and}$$

$$\mathcal{V}_2 := \bigcup_{i\in\mathcal{N}, t\geq 0} \left\{\tilde{\pi}^i_t(B^i) : B^i \subseteq \Gamma^i_{SD}, B^i \neq \varnothing\right\}.$$

Remark: The random variables in $\mathcal{V}_1 \cup \mathcal{V}_2$ are taken to be primitive random variables that, with the exception of exploration phase lengths $\{T^i_k\}_{i\in\mathcal{N},k\geq 0}$, do not depend on any player's choice of hyperparameters. The primitive random variables $\{\tilde{u}^i_t : i \in \mathcal{N}, t \geq 0\}$ should not be conflated with the action process $\{u^i_t : i \in \mathcal{N}, t \geq 0\}$, which depends on the sample path of play.

B. Assumptions

In order to state our main result, Theorem 1, we now impose some assumptions on the underlying game $\mathcal{G}$ and on the choices of hyperparameters at each player.

Assumption 2: For any pair of states $(s,s') \in \mathbb{X}\times\mathbb{X}$, there exists $H = H(s,s') \in \mathbb{N}$ and a sequence of joint actions $\mathbf{a}_0, \ldots, \mathbf{a}_H \in \mathbf{U}$ such that

$$\Pr(x_{H+1} = s' | x_0 = s, \mathbf{u}_0 = \mathbf{a}_0, \ldots, \mathbf{u}_H = \mathbf{a}_H) > 0.$$

Assumption 2 is a rather weak assumption on the transition kernel P; c.f. [14, Assumption 4.1, Case iv].

Our next assumption restricts the hyperparameter selections in Algorithm 1. Let $\bar{\delta} := \min\left(\mathfrak{A} \setminus \{0\}\right)$, where

$$\begin{aligned} \mathfrak{A} := \big\{ \left| Q^{*i}_{\boldsymbol{\pi}^{-i}}(s, a^i_1) - Q^{*i}_{\boldsymbol{\pi}^{-i}}(s, a^i_2) \right| : \\ i \in \mathcal{N}, \boldsymbol{\pi}^{-i} \in \boldsymbol{\Gamma}^{-i}_{SD}, s \in \mathbb{X}, a^i_1, a^i_2 \in \mathbb{U}^i \big\}. \end{aligned}$$

The quantity $\bar{\delta}$, defined originally by [6] and recalled above, is the minimum *non-zero* separation between two optimal Q-factors with matching states, minimized over all agents $i \in \mathcal{N}$ and over all policies $\pi^{-i} \in \Pi^{-i}$.

For any baseline policy $\boldsymbol{\pi} \in \boldsymbol{\Gamma}_{SD}$ and fixed exploration parameters $\{\rho^i\}_{i\in\mathcal{N}}$, we use the notation $\hat{\boldsymbol{\pi}} \in \boldsymbol{\Gamma}_S$ to denote a corresponding behaviour policy, which is stationary but not deterministic. When using $\hat{\pi}^i$, agent $i \in \mathcal{N}$ follows π^i with probability $1-\rho^i$ and mixes uniformly over $\mathbb{U}^i$ with probability ρ^i. The optimal Q-functions for these two environments will be close provided ρ^i is sufficiently small for all players i [6, Lemma B3]. In particular, there exists $\bar{\rho} > 0$ such that if $\rho^i \in (0,\bar{\rho})$ $\forall i \in \mathcal{N}$, then for any player j

$$\| Q^{*j}_{\boldsymbol{\pi}^{-j}} - Q^{*j}_{\hat{\boldsymbol{\pi}}^{-j}} \|_\infty < \frac{\min_i\{\delta^i, \bar{\delta} - \delta^i\}}{4}, \ \forall \boldsymbol{\pi}^{-j} \in \boldsymbol{\Gamma}^{-j}_{SD}.$$

Assumption 3: For all $i \in \mathcal{N}$, $\delta^i \in (0,\bar{\delta})$ and $\rho^i \in (0,\bar{\rho})$.

Assumption 4: There exists integers $R, T \in \mathbb{N}$ such that

$$\Pr\left(\cap_{i\in\mathcal{N},k\geq 0}\left\{T^i_k \in [T, RT]\right\}\right) = 1.$$

When all players use Algorithm 1, the resulting sequence of policies $\{\pi^i_k\}_{k\geq 0}$ is player i's *baseline* policy process, where π^i_k is i's baseline policy during $[t^i_k, t^i_{k+1})$, player i's k^{th} exploration phase. The sequence $\{\pi^i_k\}_{k\geq 0}$ is indexed by the coarser timescale of exploration phases. We also introduce a sequence of baseline policies indexed by the finer timescale of stage games. For $t \geq 0$ with $t \in [t^i_k, t^i_{k+1})$, let $\phi^i_t = \pi^i_k$ denote player i's baseline policy during the stage game at time t. The baseline joint policy at stage game t is then denoted $\boldsymbol{\phi}_t = (\phi^i_t)_{i\in\mathcal{N}}$. Furthermore, we refer to the collection of Q-factor step-size parameters $\{\alpha^i\}_{i\in\mathcal{N}}$ as $\boldsymbol{\alpha} \in (0,1)^N$.

Theorem 1: Let $\mathcal{G}$ be a weakly acyclic game and suppose each player uses Algorithm 1 to play $\mathcal{G}$. Suppose Assumptions 1–4 hold, and let $\epsilon > 0$. There exists $\bar{\alpha}_\epsilon > 0$ and a function $\bar{T}_\epsilon : (0,1)^N \times \mathbb{N} \to \mathbb{N}$ such that if

$$\max_{i\in\mathcal{N}} \alpha^i < \bar{\alpha}_\epsilon, \text{ and } T \geq \bar{T}_\epsilon(\boldsymbol{\alpha}, R),$$

then $\Pr(\boldsymbol{\phi}_t \in \boldsymbol{\Gamma}^{0\text{-eq}}_{SD}) \geq 1 - \epsilon$, for all sufficiently large $t \in \mathbb{N}$.

For a proof of Theorem 1, see [11].

IV. Discussion

A. Proof Outline

The proof of Theorem 1 involves three major steps. First, we introduce a sequence of equilibrium events, $\{B_k\}_{k\geq 0}$, defined using a sequence of random time intervals $\{[\tau^{\min}_k, \tau^{\max}_k]\}_{k\geq 0}$, to be elaborated in the sequel. For $k \geq 0$, we put

$$B_k := \left\{ \boldsymbol{\phi}_t = \boldsymbol{\phi}_{\tau^{\min}_k} \in \boldsymbol{\Gamma}^{0\text{-eq}}_{SD} : t = \tau^{\min}_k + 1, \ldots, \tau^{\max}_k \right\}.$$

In words, B_k is the event in which the baseline policy did not change during the interval $[\tau^{\min}_k, \tau^{\max}_k]$ and moreover the baseline policy was an equilibrium during this time.

Second, for a distinguished $L < \infty$, we argue that the probability of driving play to this equilibrium event in L time steps can be lower bounded: $\Pr(B_{k+L}|B^c_k) \geq \hat{p}_{\min} > 0$. Third, we argue that we can control and lower bound the probability of remaining at the equilibrium event for L steps: that is, $\Pr(B_{k+L}|B_k)$ can be made arbitrarily large relative to $\hat{p}_{\min}$. One can then explicitly lower bound $\Pr(B_{k+mL})$ for suitably large m, as in [6] and [10].

B. The Proof Under Synchrony

In the synchronous variant of the algorithm, we have $T^i_k = T^j_k = T_k$ for any agents $i, j \in \mathcal{N}$ and $k \geq 0$. Agents always begin and end their exploration phases in synch. Crucially, no agent ever switches its policy while another agent is actively learning its Q-factors. As a result, each agent faces an MDP during each exploration phase, and one can study the convergence of Q-factors using single-agent theory. In particular, when player i employs suitably *decreasing* step-sizes, one has that $\widehat{Q}^i_t \to Q^{*i}_{\boldsymbol{\pi}^{-i}_k}$, as $T_k \to \infty$.

When analyzing the synchronous algorithm, one defines $\tau^{\min}_k = \tau^{\max}_k = t_k$, the stage game time marking the beginning of the k^{th} shared exploration phase. Importantly, in the synchronous case, one is guaranteed that the joint policy is fixed at $\boldsymbol{\pi}_k$ throughout the interval $[t_k, t_{k+1})$. Thus, each player i spends T_k stages learning against the policy $\boldsymbol{\phi}^{-i}_{\tau^{\max}_k} = \boldsymbol{\pi}^{-i}_k$. With these choices of $\tau^{\min}_k$ and $\tau^{\max}_k$, B_k is equivalent to $\{\boldsymbol{\pi}_k \in \boldsymbol{\Gamma}^{0\text{-eq}}_{SD}\}$.

In the synchronous setting, the second step of the proof outline, showing $\Pr(B_{k+L}|B_k) > \hat{p}_{\min}$, can be established using weak acyclicity and conditioning on an event where players learn their Q-factors approximately correctly.

At equilibrium, if players recover their optimal Q-functions, then they will opt to remain with their current policies. Under synchrony, the probability of (approximately) recovering optimal Q-functions can be controlled directly by taking the *shared* exploration phase lengths $\{T_k\}_{k\geq 0}$ to be large. Thus, the third step of the proof outline, of showing that $\Pr(B_{k+L}|B_k)$ can be made large relative to $\hat{p}_{\min}$, is straightforward in the synchronous setting.

C. Challenges under Asynchrony

Moving to the asynchronous setting, where $T^i_k \neq T^j_k$ is allowed, we observe that the preceding definitions of $\tau^{\min}_k$ and $\tau^{\max}_k$ are not meaningful, since there is no shared k^{th} exploration phase. Consequently, a new definition of the equilibrium event B_k is required.

Any useful definition of $\{B_k\}_{k\geq 0}$ must be *self-reinforcing*, in the sense that $\Pr(B_{k+L}|B_k)$ should be controllable by appropriate choice of hyperparameters. In particular, this can be done by controlling the conditional probability that players approximately recover their equilibrium Q-functions given that the equilibrium event occurred.

Since each player's Q-factors are constructed using historical feedback data, one must account for the recent history of play when analyzing Q-iterates. For example, in its m^{th} exploration phase, player i learns from time t^i_m to time $t^i_{m+1} = t^i_m + T^i_m$. During this interval, player i's counterparts may have switched policies several times. If one wishes to ensure that i's Q-estimates at time t^i_{m+1} reflect the environment determined by $\boldsymbol{\phi}^{-i}_{t^i_{m+1}}$, then a natural condition to impose is that i spent a significant period learning against the most recent baseline policy, $\boldsymbol{\phi}^{-i}_{t^i_{m+1}}$.

To properly account for the recent history of play in our definition of the equilibrium event, we define the sequences $\{\tau^{\min}_k, \tau^{\max}_k\}$ recursively as follows: put $\tau^{\min}_0 = \tau^{\max}_0 = 0$. For $k \geq 0$, let

$$\begin{aligned}
\tau^{\min}_{k+1} &:= \inf\{t^i_n : t^i_n > \tau^{\max}_k, i \in \mathcal{N}, n \geq 0\} \\
\tau^{\max}_{k+1} &:= \inf\big\{t \geq \tau^{\min}_{k+1} : \forall\, i \in \mathcal{N}, \exists\, n \text{ s.t. } t^i_n \in [\tau^{\min}_{k+1}, t], \\
&\qquad \text{and } \inf\{t^i_{\bar{n}} > t : i \in \mathcal{N}, \bar{n} \in \mathbb{N}\} \geq t + T/N\big\}.
\end{aligned}$$

The intervals $[\tau^{\min}_k, \tau^{\max}_k]$ represent active phases, during which players may change their policies.[5] Various important consequences of the definitions are described in [11], some of which we include below:

(1) No policy updates occur in $(\tau^{\max}_k, \tau^{\min}_{k+1})$.
(2) Each agent has at least one opportunity to switch policies during $[\tau^{\min}_{k+1}, \tau^{\max}_{k+1}]$.
(3) For each $k \geq 0$, $\tau^{\min}_{k+1} \geq \tau^{\max}_k + T/N$.
(4) Any player has at most $R + 1$ opportunities to switch its policy during $[\tau^{\min}_k, \tau^{\max}_k]$.

Since $\tau^{\min}_{k+1}$ is the first time after $\tau^{\max}_k$ at which any agent has the opportunity to switch its policy, the first and

[5] If $T^i_k = T$ for each i, k, then we return to the synchronous case, and these definitions yield $\tau^{\min}_k = \tau^{\max}_k = t_k$ for $k \geq 0$, coinciding with the synchronous analysis.

third items above offers a means by which B_k can be self-reinforcing: given B_k, one is guaranteed that each player i spends at least T/N stages learning against the unchanging policy $\phi^{-i}_{\tau^{\max}_k}$. Indeed, other formulations of $\tau^{\min}_k$ and $\tau^{\max}_k$ were considered, but simpler alternatives could not yield a meaningful analog of item (3).

The extent to which $\Pr(B_{k+L}|B_k)$ can be controlled depends on the conditional probability that, given B_k, agents learn their equilibrium Q-factors approximately correctly. This, in turn, depends on the specific Q-learning update used and the length of time spent learning against outdated environments.

If an agent uses decreasing step-sizes and has a relatively long exploration phase, then by the time play first arrives at equilibrium, this player's learning rates will be very small and the player will have spent the majority of its exploration phase learning against older, possibly irrelevant joint policies. These considerations lead to the utilization of constant learning rates in Algorithm 1. Constant learning rates allow for an agent to quickly change its Q-factors in response to a change in the joint policy.

For full derivations and an expanded discussion, see [11].

D. Future Work

Due to space constraints, some interesting aspects of the proof were not discussed above. One notable example is a confounding effect involving conditional probabilities: the event B_k carries information about the state-action trajectory before $\tau^{\max}_k$, which in turn is correlated with the evolution of Q-iterates for times after $\tau^{\max}_k$. Our results, hypotheses, and lemmas are described in terms of primitive random variables and hypothetical Q-factors to address this analytical challenge. This approach rules out *adaptive exploration phase lengths*, where players may choose to curtail or prolong an exploration phase as a function of history. Whether regret testing algorithms can accommodate asynchronous, adaptive exploration phases is an interesting open question for future research.

For other directions of future work, see [11].

V. Conclusions

In this paper, we considered an asynchronous variant of the Decentralized Q-learning algorithm of [6]. To accommodate asynchronous policy updating and non-stationarity in each agent's learning environment, we have utilized a constant learning rate that can rapidly overcome errors in learning estimates that are artifacts of outdated information. With this algorithmic change, we have shown that Decentralized Q-learning can still drive policies to equilibrium in weakly acyclic games without making strong coordination assumptions.

References

[1] P. Hernandez-Leal, B. Kartal, and M. E. Taylor, "A survey and critique of multiagent deep reinforcement learning," *Autonomous Agents and Multi-Agent Systems*, vol. 33, no. 6, pp. 750–797, 2019.

[2] P. Hernandez-Leal, M. Kaisers, T. Baarslag, and E. M. de Cote, "A survey of learning in multiagent environments: Dealing with non-stationarity," *arXiv preprint arXiv:1707.09183*, 2017.

[3] M. L. Littman and C. Szepesvári, "A generalized reinforcement-learning model: Convergence and applications," in *ICML*, vol. 96, pp. 310–318, Citeseer, 1996.

[4] M. L. Littman, "Friend-or-foe Q-learning in general-sum games," in *ICML*, vol. 1, pp. 322–328, 2001.

[5] J. Hu and M. P. Wellman, "Nash Q-learning for general-sum stochastic games," *Journal of Machine Learning Research*, vol. 4, no. Nov, pp. 1039–1069, 2003.

[6] G. Arslan and S. Yüksel, "Decentralized Q-learning for stochastic teams and games," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1545–1558, 2017.

[7] C. Daskalakis, D. J. Foster, and N. Golowich, "Independent policy gradient methods for competitive reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 5527–5540, 2020.

[8] M. Sayin, K. Zhang, D. Leslie, T. Başar, and A. Ozdaglar, "Decentralized Q-learning in zero-sum Markov games," *Advances in Neural Information Processing Systems*, vol. 34, pp. 18320–18334, 2021.

[9] B. Yongacoglu, G. Arslan, and S. Yüksel, "Decentralized learning for optimality in stochastic dynamic teams and games with local control and global state information," *IEEE Transactions on Automatic Control*, vol. 67, no. 10, pp. 5230–5245, 2022.

[10] B. Yongacoglu, G. Arslan, and S. Yüksel, "Satisficing paths and independent multi-agent reinforcement learning in stochastic games," *SIAM Journal on Mathematics of Data Science*, to appear.

[11] B. Yongacoglu, G. Arslan, and S. Yüksel, "Asynchronous decentralized Q-learning: Two timescale analysis by persistence," *arXiv preprint arXiv:2308.03239*, 2023.

[12] C. Claus and C. Boutilier, "The dynamics of reinforcement learning in cooperative multiagent systems," in *Proceedings of the Tenth Innovative Applications of Artificial Intelligence Conference, Madison, Wisconsin*, pp. 746–752, 1998.

[13] L. Matignon, G. J. Laurent, and N. Le Fort-Piat, "Independent reinforcement learners in cooperative markov games: a survey regarding coordination problems.," *Knowledge Engineering Review*, vol. 27, no. 1, pp. 1–31, 2012.

[14] A. Ozdaglar, M. O. Sayin, and K. Zhang, "Independent learning in stochastic games," *arXiv preprint arXiv:2111.11743*, 2021.

[15] H. Nekoei, A. Badrinaaraayanan, A. Sinha, M. Amini, J. Rajendran, A. Mahajan, and S. Chandar, "Dealing with non-stationarity in decentralized cooperative multi-agent deep reinforcement learning via multi-timescale learning," *arXiv preprint arXiv:2302.02792*, 2023.

[16] D. Foster and H. P. Young, "Regret testing: Learning to play Nash equilibrium without knowing you have an opponent," *Theoretical Economics*, vol. 1, pp. 341–367, 2006.

[17] F. Germano and G. Lugosi, "Global Nash convergence of Foster and Young's regret testing," *Games and Economic Behavior*, vol. 60, no. 1, pp. 135–154, 2007.

[18] J. R. Marden, H. P. Young, G. Arslan, and J. S. Shamma, "Payoff-based dynamics for multiplayer weakly acyclic games," *SIAM Journal on Control and Optimization*, vol. 48, no. 1, pp. 373–396, 2009.

[19] C. Maheshwari, M. Wu, D. Pai, and S. Sastry, "Independent and decentralized learning in markov potential games," *arXiv preprint arXiv:2205.14590*, 2022.

[20] A. M. Fink, "Equilibrium in a stochastic n-person game," *Journal of Science of the Hiroshima University, Series AI (Mathematics)*, vol. 28, no. 1, pp. 89–93, 1964.

[21] H. P. Young, *Strategic Learning and its Limits*. Oxford University Press., 2004.

[22] D. H. Mguni, Y. Wu, Y. Du, Y. Yang, Z. Wang, M. Li, Y. Wen, J. Jennings, and J. Wang, "Learning in nonzero-sum stochastic games with potentials," in *International Conference on Machine Learning*, pp. 7688–7699, PMLR, 2021.

[23] S. Leonardos, W. Overman, I. Panageas, and G. Piliouras, "Global convergence of multi-agent policy gradient in Markov potential games," *arXiv preprint arXiv:2106.01969*, 2021.

[24] R. Zhang, Z. Ren, and N. Li, "Gradient play in multi-agent Markov stochastic games: Stationary points and convergence," *arXiv preprint arXiv:2106.00198*, 2021.