A Law of Iterated Logarithm for Multi-Agent Reinforcement Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

In Multi-Agent Reinforcement Learning (MARL), multiple agents interact with a common environment and with each other, for solving a shared problem in sequential decision-making. Algorithms for MARL have a wealth of application in popular domains including gaming, robotics, and finance. In this work, we study a family of distributed nonlinear stochastic approximation schemes useful in MARL and derive a novel law of iterated logarithm. In particular, our result describes the convergence rate on almost every sample path where the algorithm converges. This result is the first of its kind in the distributed setup and provides deeper insights than the existing ones, which only discuss convergence rates in the expected or the CLT sense. Importantly, our result holds under significantly weaker assumptions: neither the gossip matrix needs to be doubly stochastic nor the stepsizes square summable. As an application, we show that, for the stepsize $n^{-\gamma}$ with $\gamma \in (0, 1)$, the distributed TD(0) algorithm with linear function approximation has a convergence rate of $\mathcal{O}(\sqrt{n^{-\gamma} \log n})$ a.s.; for the $1/n$ type stepsize, it is $\mathcal{O}(\sqrt{n^{-1} \log \log n})$ a.s. These growth rates do not depend on the graph depicting the interactions among the different agents.

1 Introduction

Can a machine train itself in the same way an infant learns to sit up, crawl, and walk? That is, can a device interact with the environment and figure out the action sequence required to complete a given task? The study of *algorithms* that enable such decision-making is what the field of Reinforcement Learning (RL) is all about [47]. In contrast, the *mathematics* needed to analyze such schemes is what forms the focus in Stochastic Approximation (SA) theory [1, 4]. More generally, SA refers to an iterative scheme that helps find zeroes or optimal points of a function, for which only noisy evaluations are possible. In this work, we analyze a family of Distributed Stochastic Approximation (DSA) algorithms [26] that are useful in Multi-Agent Reinforcement Learning (MARL) [24, 59].

In the MARL framework, we have multiple agents or learners that continually engage with a shared environment: the agents pick local actions, and the environment responds by transitioning to a new state and giving each agent a different local reward. Additionally, the agents also *gossip* about local computations with each other. The goal of the agents is to cooperatively find action policies that maximize the collective rewards obtained over time. Algorithms useful in this endeavor have found empirical success in domains as diverse as gaming [31], robotics [32], autonomous driving [41], communication networks [23], power grids [37], and economics [20]. However, theoretical analyses of such schemes are still very minimal, and this is what this paper aims to address.

For the purpose of analysis, MARL methods are often viewed as special cases of DSA algorithms. We now describe an archetypical form of a DSA scheme with m distributed nodes. Let $\mathcal{G}$ be a

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

directed graph representing the connections between these nodes, and $W \equiv (W_{ij}) \in [0,1]^{m\times m}$ a matrix whose ij-th entry denotes the strength of the edge $j \to i$. We assume that W is compatible with $\mathcal{G}$, i.e., $W_{ij} > 0$ only if $j \to i \in \mathcal{G}$. Then, at agent i, the above scheme has the update rule

$$x_{n+1}(i) = \sum_{j\in\mathcal{N}_i} W_{ij} x_n(j) + \alpha_n[h_i(x_n) + M_{n+1}(i)], \qquad n \geq 0, \tag{1}$$

where $x_n \in \mathbb{R}^{m\times d}$ is the joint estimate of the solution at time n, its j-th row, i.e., $x_n(j)$ denotes[1] the estimate obtained at agent j, $\mathcal{N}_i$ represents the set of in-neighbors of node i in $\mathcal{G}$, α_n is the stepsize, $h_i : \mathbb{R}^{m\times n} \to \mathbb{R}^d$ is the driving function at agent i, and $M_{n+1}(i) \in \mathbb{R}^d$ is the noise in its evaluation at time n. The first term in the update rule averages iterates between neighbors while the second term recursively refines the iterates at each node. The joint update rule of all the agents can be written as

$$x_{n+1} = Wx_n + \alpha_n[h(x_n) + M_{n+1}], \tag{2}$$

where M_{n+1} is the $m \times d$ matrix whose i-th row is $M_{n+1}(i)$, and h is the function that maps $x \in \mathbb{R}^{m\times d}$ to the $m \times d$ matrix whose i-th row is $h_i(x)$.

Two important points about the above framework are as follows: i.) we allow h_i to be a function of all of x_n and not just of $x_n(i)$, as is commonly assumed, and ii.) the computations at different nodes in the above setup run synchronously on a common clock.

Related Work: We now give a summary of relevant theoretical results from the DSA and MARL literature. For ease of discussion, we categorize them into i.) asymptotic and ii.) finite-time results.

Starting with the seminal work [50], most asymptotic results mainly concern almost sure (a.s.) convergence [2, 28, 26, 14, 60, 58, 46, 19]. The first four papers here provide convergence guarantees for a broad family of nonlinear DSA algorithms. The other articles also do the same, but in context of specific MARL schemes such as distributed Q-learning, distributed actor-critic methods, distributed TD methods, and their off-policy variations. Two other kinds of asymptotic results also exist in the literature. The first is the CLT shown in [29] for the average of estimates obtained at different nodes in a generic DSA scheme. The other is the convergence in mean result obtained in [56] for a distributed policy gradient method.

Finite-time literature, in contrast, majorly talks about expectation bounds. Assuming there exists a unique x_* that solves $\sum_{i=1}^m h_i(x) = 0$, these results describe the rate at which $\mathbb{E}\|x_n - x_*\|$ decays with n. Notable contributions for DSA here are [57, 52]. Compared to ours, these look at a slightly different setup: the measurement noise at each node has a Markov component in place of a martingale difference term. In this new setup, [57] shows that, for any sufficiently small but constant stepsize ϵ, the expected error decreases linearly to a ball of radius $\mathcal{O}(\sqrt{\epsilon\log(1/\epsilon)})$. On the other hand, [52] deals with the case where h is additionally non-convex and shows that $\mathbb{E}\|x_n - x_*\| = \mathcal{O}(n^{-1/4}\sqrt{\log n})$, which is comparable to the best known bound in the centralized setting.

Expectation bounds in the MARL framework primarily concern policy evaluation methods [8, 9, 44, 5]. The first three papers here deal with the distributed TD(0) method. These show a result similar to the one in [57] holds for this method under constant stepsizes. In contrast, when α_n is of the $1/n$ type, it is proven that $\mathbb{E}\|x_n - x_*\| = \mathcal{O}(1/\sqrt{n})$. Similar bounds have also been derived in [5] for two distributed variants of the TDC method. There are also some other works that derive finite-time bounds [53, 7, 54, 62, 11, 43, 36, 61], but we do not discuss them in this paper since the algorithms proposed there do not fit the update rule given in (2).

The different finite-time results, as also the asymptotic CLT, provide insights into the rate at which an iterative method converges. However, for algorithms that mirror (2), there is a significant issue with studies discussing such results: all, except [29], require the gossip matrix to be doubly stochastic, at least in the mean. While this assumption simplifies the analysis, it also severely restricts the communication protocol choices. In fact, as pointed out in [29], this condition even limits the use of a natural broadcast node, one that transmits its local estimate to all the neighbors without expecting all of them to respond. Another critical limitation of the present works is that they only talk about convergence rates in the expected or the CLT sense. By their very nature, these results do not reveal much about the decay rates along different sample paths. Finally, all current results, including the ones on convergence, only apply for constant or square-summable stepsizes. However, the slowly-decaying non-square-summable ones are generally preferable since they give similar benefits as

[1] By default, all our vectors are row vectors. We use $'$ for transpose.

constant stepsizes and, often, also guarantee convergence. It is worth noting that such issues also plague much of the distributed stochastic optimization literature [55, 45, 21, 15, 35, 34].

Key Contributions: The highlights of this work that help us address the above issues are as follows.

1. **Law of Iterated Logarithm (LIL)**: We derive a novel law of iterated logarithm for the DSA scheme given in (2). That is, a.s. on every sample path where $x_n \to x_*$, we show that $\limsup[\alpha_n \log t_{n+1}]^{-1/2}\|x_n - x_*\| \leq C$, where $C \geq 0$ is some constant and $t_n = \sum_{k=0}^{n-1} \alpha_k$. To our knowledge, this result is the first of its kind in the distributed setup. Further, as discussed in Remark 2.3, it provides deeper insights about the asymptotic behaviour of $(x_n)_{n\geq 0}$ than other existing results, which discuss convergence rates in the expected or the CLT sense.
2. **Analysis and Gossip Matrix**: The above result is obtained via a new approach we develop here for analyzing DSA schemes. Let π be the stationary distribution associated with W and let $Q := \mathbb{I} - \mathbf{1}'\pi$, where $\mathbf{1}$ denote the vector of all ones. Then an outline of our approach is that we express $x_n - x_*$ as a sum of $\mathbf{1}'\pi(x_n - x_*)$ and Qx_n and, thereafter, analyze each summand by treating its update rule as a distinct SA scheme. This contrasts the usual approach (e.g., [29, 8, 9]) where the error is split into $\mathbf{1}'\mathbf{1}x_n$ and $(\mathbb{I} - \mathbf{1}'\mathbf{1})x_n$. In fact, this is the precise reason why, unlike other existing results, ours does not require that the gossip matrix be doubly stochastic.
3. **Concentration Inequality and Stepsizes**: We also improve upon a concentration result ([10, Corollary 6.4.25]) for a sum of martingale differences; see Lemma 4.8. Specifically, by modifying the original proof from [10], we show that the result stated there actually holds under a broader set of conditions. The benefit of this is that, unlike other related results, our LIL result does not require that the stepsize sequence be square summable.
4. **MARL Application**: We use our theory to show a law of iterated logarithm for the distributed TD(0) algorithm with linear function approximation. This is the first such result in MARL.

Contents: The rest of the paper is structured as follows. In Section 2, we formally state our main result along with all the assumptions needed. We also pinpoint the new insights that our result provides. In Section 3, we give a demonstration of how our result can be applied in the MARL setup. In particular, there we talk about the distributed TD(0) algorithm with linear function approximation and prove that it indeed satisfies all the assumptions of our main result. Section 4 has two parts. In the first part, we state some key intermediate lemmas and then use the same to derive our main result. The latter part, in contrast, focuses on proofs of these intermediate results; note that we only sketch their proofs here and leave the details to the appendix. Finally, in Section 5, we conclude with a summary of our findings and discuss some interesting future directions.

2 Assumptions and Main Result

We first state our four technical assumptions, denoted $\mathcal{A}_1, \ldots, \mathcal{A}_4$, for the DSA scheme in (2).

$\mathcal{A}_1$. **Property of the Averaging Matrix**: *W is an irreducible aperiodic row stochastic matrix.*

This condition implies there exists a unique vector $\pi \in \mathbb{R}^m$ such that

$$\pi W = \pi. \tag{3}$$

Subsequently, from the discussions in [26], one would expect (2) to eventually track the solution of the m-fold product of the d-dimensional ODE

$$\dot{y}(t) = \sum_{i=1}^{m} \pi_i h_i(\mathbf{1}'y(t)) = \pi h(\mathbf{1}'y(t)). \tag{4}$$

Let $x_* = \mathbf{1}'y_*$, where $y_* \in \mathbb{R}^d$ is an asymptotically stable equilibrium of (4). Notice that we don't assume (4) has a unique attractor.

$\mathcal{A}_2$. **Nature of h near x_***: *There exists a neighbourhood $\mathcal{U}$ of x_* such that, for $x \in \mathcal{U}$,*

$$h(x) = -\mathbf{1}'\pi(x - x_*)A + \mathbf{1}'\pi f_1(x) + (\mathbb{I} - \mathbf{1}'\pi)(B + f_2(x)), \tag{5}$$

where $A \in \mathbb{R}^{d\times d}$ is such that $yAy' > 0$ for all $y \neq 0$, $B \in \mathbb{R}^{m\times d}$ is some constant matrix, $f_2 : \mathcal{U} \to \mathbb{R}^{m\times d}$ is some arbitrary continuous function, while $f_1 : \mathcal{U} \to \mathbb{R}^{m\times d}$ is another continuous function that additionally satisfies

$$\|\boldsymbol{I}'\pi f_1(x)\| = \mathcal{O}(\|\boldsymbol{I}'\pi(x - x_*)\|^a) \tag{6}$$

under some norm $\|\cdot\|$ and for some $a > 1$.

Note that $\mathcal{A}_2$ is a generalization of **Assumption (A1)** stated in [33]. Also, observe that this assumption is local in nature. In that, it only prescribes a specific behavior close to x_*. A simple example that satisfies the conditions stated here is $h(x) = B - xA$, where B and A are as defined above. As shown in Section 3, this specific choice of h is an extremely useful for policy evaluation in MARL.

$\mathcal{A}_3$. **Stepsize Behaviour**: *There exists some decreasing positive function α defined on $[0, \infty)$ such that the stepsize $\alpha_n = \alpha(n)$. Further, α is either of Type 1 or Type γ. That is, either*

(a) *Type 1: $\alpha(n) = \frac{\alpha_0}{n}$ with $\alpha_0 > \frac{1}{2\lambda_{\min}}$, where $\lambda_{\min} := \min\{Re(\lambda) : \lambda \in \text{spectrum}(A)\}$ or*

(b) *Type γ : The function α is differentiable and its derivative varies regularly with exponent $-1-\gamma$, where $0 < \gamma < 1$.*

The regularly varying condition in the Type γ case implies that $\left|\frac{\mathrm{d}\alpha(x)}{\mathrm{d}x}\right| = x^{-\gamma-1}L(x)$ for some slowly varying function L, e.g., $L(x) = c$ for some $c > 0$, or $L(x) = (\log x)^\eta$ for some $\eta \in \mathbb{R}$. Thus, examples of α_n here include $cn^{-\gamma}$, $n^{-\gamma}(\log n)^\eta$, etc.

$\mathcal{A}_4$. **Noise Attributes**: *With $\mathcal{F}_n = \sigma(x_0, M_1, \ldots, M_n)$, $n \geq 0$, we have*:

(a) $\mathbb{E}(M_{n+1}|\mathcal{F}_n) = 0$ *a.s.*

(b) *There exists $C \geq 0$ such that $\|QM_{n+1}\| \leq C\,(1 + \|Q(x_n - x_*)\|)$ a.s. on $\mathcal{E}(x_*)$.*

(c) *There is a non-random symmetric positive semi-definite matrix $M \in \mathbb{R}^{d\times d}$ such that*

$$\lim_{n\to\infty} \mathbb{E}(M'_{n+1}\pi'\pi M_{n+1} \,|\, \mathcal{F}_n) = M \quad \text{a.s. on } \mathcal{E}(x_*). \tag{7}$$

(d) *There exists $b > 2$ such that $\sup_{n\geq 0} \mathbb{E}(\|\pi M_{n+1}\|^b|\mathcal{F}_n) < \infty$ a.s. on $\mathcal{E}(x_*)$.*

These noise conditions are extensions of the standard assumptions in the SA literature [33, 27, 4].

Our main result can now be stated as follows. This extends Theorem 1 stated in [33].

Theorem 2.1 (**Main Result: Law of Iterated Logarithm**). *Let $t_n = \sum_{k=0}^{n-1} \alpha_k$, $n \geq 0$. Suppose $\mathcal{A}_1$—$\mathcal{A}_4$ hold and $\gamma > 2/b$. Then, there exists some deterministic constant $C \geq 0$ such that*

$$\limsup[\alpha_n \log t_{n+1}]^{-1/2}\|x_n - x_*\| \leq C \quad \text{a.s. on } \mathcal{E}(x_*).$$

This result is called a law of iterated logarithm since its proof relies on Lemma 4.8, which in turn is a generalization of the standard law of iterated logarithm for a sum of scaled martingale differences. It provides the following insights into the behavior of the DSA scheme given in (2).

Remark 2.2. *For the Type 1 case, our result shows $\|x_n - x_*\| = \mathcal{O}(\sqrt{n^{-1}\log\log n})$ a.s. on $\mathcal{E}(x_*)$. In contrast, for the Type γ case, we get $\|x_n - x_*\| = \mathcal{O}(\sqrt{n^{-\gamma}\log n})$ a.s. on $\mathcal{E}(x_*)$.*

Remark 2.3. *Our result provides deeper insights than the convergence rates that exist in the DSA/MARL literature. For this discussion, we suppose $\mathbb{P}\{\mathcal{E}(x_*)\} = 1$. As discussed in Section 1, the existing results are of two kinds: finite-time expectation bounds and the CLT. Indeed a finite-time bound has several benefits and is not directly comparable to an asymptotic result. Nevertheless, an expectation bound only describes the average behaviour, while our result characterizes the decay rate on almost every sample path. Furthermore, while a CLT can at the best show that $\limsup \alpha_n^{-1/2}\|x_n - x_*\| = \infty$ a.s., our result is more precise in stating that the expression becomes bounded if it is divided by an additional $\sqrt{\log t_{n+1}}$ term.*

3 Application to Reinforcement Learning

We apply our result here to a variant of the distributed TD(0) algorithm [8, 9] with linear function approximation. This method is useful for policy evaluation in the multi-agent setup. The discussion here is divided into the following three parts: i.) setup, ii.) objective and algorithm, and iii.) analysis.

Setup: We consider a distributed system of m agents modeled by a Markov Decision Process. This can be characterized by the tuple $(\mathcal{S}, \{\mathcal{U}_i\}, \mathcal{P}, \{\mathcal{R}_i\}, \gamma, \mathcal{G})$. Here, $\mathcal{S} = \{1, \ldots, L\}$ is the global state space, $\mathcal{U}_i$ and $\mathcal{R}_i$ are the set of actions and the reward function at agent i, respectively, $\mathcal{P}$ describes the transition probabilities, γ is the discount factor, and $\mathcal{G} \equiv (\mathcal{V}, \mathcal{E})$ is a directed graph that represents the connectivity structure among the m agents.

Let $\mathcal{N}_i$ be the set of in-neighbours of agent i in $\mathcal{G}$. Also, let $W \equiv (W_{ij}) \in \mathbb{R}^{m\times m}$ be a weight matrix compatible with $\mathcal{G}$, i.e., $W_{ij} > 0$ only if $j \in \mathcal{N}_i$. We assume that this matrix satisfies the conditions in $\mathcal{A}_1$. Then, for this W, there is a unique vector $\pi \equiv (\pi_i)$ satisfying (3).

Let μ_i be the stationary policy of agent i and let $\mu \equiv (\mu_i)$. Also, let $\mu(a|s) = \prod_i \mu_i(a_i|s)$ be the probability for choosing the joint action $a \equiv (a_i) \in \prod_i \mathcal{U}_i$. This policy μ then induces a Markov chain on $\mathcal{S}$, which we assume is aperiodic and irreducible. Therefore, it also has a unique stationary distribution and we denote the same by $\varphi \in \mathbb{R}^L$.

At each step, the above system evolves as follows. First, each agent i sees the current state s and applies an action $a_i \in \mathcal{U}_i$ sampled from $\mu_i(\cdot|s)$. Based on the joint action a, the system then moves to a new state $\tilde{s}$. Equivalently, the joint action a and the state $\tilde{s}$ can be seen as samples of $\mu(\cdot|s)$ and $\mathcal{P}(\cdot|s, a)$, respectively. Finally, each agent i receives an instantaneous reward $\mathcal{R}_i(s, a, \tilde{s})$.

Objective and Algorithm: The goal of the multi-agent system is to cooperatively estimate the value function $J^\mu \in \mathbb{R}^L$ corresponding to μ. This is defined as the solution to the Bellman equation

$$J^\mu(s) = \mathbb{E}\Big[\sum_i \pi_i \mathcal{R}_i(s, a, \tilde{s}) + \gamma J^\mu(\tilde{s})\Big], \quad s \in \mathcal{S},$$

where the expectation is taken over $a \sim \mu(\cdot|s)$ and $\tilde{s} \sim \mathcal{P}(\cdot|s, a)$. When L is large, estimating J^μ directly is intractable. An alternative then is to make use of linear function approximation. Here, for some d, we have a feature matrix $\Phi \in \mathbb{R}^{L\times d}$ with full column rank. And, with $\phi(s)$ denoting the s-th row of Φ, the goal is to find a $\theta \in \mathbb{R}^d$ such that $J^\mu(s) \approx \phi(s)\theta'$ for all $s \in \mathcal{S}$.

The distributed TD(0) algorithm is helpful in this latter context. Let $(s_n, a_n, \tilde{s}_n)$, $n \geq 0$, be IID[2] samples of $(s, a, \tilde{s})$, where $s \sim \varphi(\cdot)$, $a \sim \mu(\cdot|s)$, and $\tilde{s} \sim \mathcal{P}(\cdot|s, a)$. Then, at agent i, the aforementioned algorithm has the update rule:

$$\theta_{n+1}(i) = \sum_{j\in\mathcal{N}_i} W_{ij}\theta_n(j) + \alpha_n(b_n(i) - \theta_n(i)A_n), \tag{8}$$

where $b_n(i) = \mathcal{R}_i(s_n, a_n, \tilde{s}_n)\phi(s_n) \in \mathbb{R}^d$ and $A_n = \phi'(s_n)\phi(s_n) - \gamma\phi'(\tilde{s}_n)\phi(s_n) \in \mathbb{R}^{d\times d}$.

Analysis: We begin by rewriting (8) in the standard form for distributed stochastic approximation. Let[3] $A = \mathbb{E}[A_n]$ and $B = \mathbb{E}[B_n]$, where $B_n \in \mathbb{R}^{m\times d}$ is the matrix whose i-th row is $b_n(i)$. Both A and B do not depend on n since $(s_n, a_n, \tilde{s}_n)$, $n \geq 0$, is IID. Next, for $n \geq 0$, let $x_n \in \mathbb{R}^{m\times d}$ be the matrix whose i-th row is $\theta_n(i)$. Then, the update rule in (8) has the form given in (2) for

$$h(x) = B - xA \qquad \text{and} \qquad M_{n+1} = (B_n - B) - x_n(A_n - A). \tag{9}$$

Next, we look at the limiting ODE for (8). As shown in (4), this has the form $\dot{y}(t) = \pi B - y(t)A$. Now, A is known to be positive definite [47], i.e., $\theta A\theta' > 0$ for all $\theta \neq 0$. Hence, it is invertible and the real parts of all its eigenvalues are positive, i.e., $-A$ is Hurwitz stable. This then shows that $\theta_* = \pi BA^{-1}$ is the unique globally asymptotically stable equilibrium for the above ODE.

[2]The IID assumption is standard in literature [6, 25, 48, 51] and is needed to ensure that the update rule only has martingale noise. Otherwise, the update rule will additionally have Markovian noise, the analysis of which is beyond the scope of this paper. The good news though is that, as shown in previous works [13, 8, 9], the asymptotic behaviours with and without the Markovian noise are often similar.

[3]The matrix A is usually defined to be the transpose of the expression we use.

We now verify the assumptions stated in Section 2. Let $f_1(x) = 0$ and $f_2(x) = -xA$ for $x \in \mathbb{R}^{m\times d}$. Further, let $x_* = \mathbf{1}'\theta_*$. Then, $h(x) = -\mathbf{1}\pi(x - x_*)A + (\mathbb{I} - \mathbf{1}\pi)(B + f_2(x))$, which verifies $\mathcal{A}_2$.

In order to satisfy $\mathcal{A}_3$, we simply choose a stepsize sequence that fulfils one of the criteria mentioned there. Separately, in the setup, W was assumed to directly satisfy the conditions in $\mathcal{A}_1$.

It now only remains to establish $\mathcal{A}_4$. Let $\mathcal{F}_n$ be as defined there. Then, part (a) follows from the definitions of A and B and the fact that $(s_n, a_n, \tilde{s}_n)$ is independent of the past. On the other hand, part (b) can be shown by building upon the arguments used in the proof of [6, Lemma 5.1]. Next observe that, since $(s_n, a_n, \tilde{s}_n)$ is independent of the past, the only quantity that is random in $\mathbb{E}[M'_{n+1}\pi'\pi M_{n+1}|\mathcal{F}_n]$ is x_n. Also, trivially, $\mathbb{E}[M'_{n+1}\pi'\pi M_{n+1}|\mathcal{F}_n]$ is a symmetric positive semi-definite matrix. Therefore, on the event $\mathcal{E}(x_*)$, it is easy to see that part (c) holds as well. Finally, notice that $\|\pi M_{n+1}\| \leq C(1+\|x_n - x_*\|)$ for some $C \geq 0$; this follows as in part (b) above. Hence, on $\mathcal{E}(x_*)$, $\sup_{n\geq 0} \mathbb{E}[\|\pi M_{n+1}\|^b|\mathcal{F}_n] < \infty$ a.s. for any $b \geq 0$. This verifies part (d).

We can now conclude that Theorem 2.1 indeed holds for the distributed TD(0) algorithm.

4 Theoretical Analysis

We now turn to the technical details of our analysis of the distributed stochastic approximation scheme. The analysis hinges on the novel error decomposition and analysis of each error term. Let $Q := \mathbb{I} - \mathbf{1}'\pi$. Since $\mathbf{1}'\pi x_* = x_*$, we have $x_n - x_* = \mathbf{1}'\pi(x_n - x_*) + Qx_n$, where we refer the first term as the agreement component of the error and the second as the disagreement component of the error. Our key idea is to obtain bounds for two summands separately as follows.

Lemma 4.1. *(Agreement Error) Almost surely on $\mathcal{E}(x_*)$,*

$$\limsup_{n\to\infty} \frac{\|\boldsymbol{1}'\pi(x_n - x_*)\|}{\sqrt{\alpha_n \log t_{n+1}}} \leq \sqrt{m}\Lambda, \tag{10}$$

where Λ is some deterministic constant.

Observe the term $\log t_{n+1}$ in the above convergence rate bound as the deviations above those predicted by a central limit theorem. This term under the step-size choice $\alpha_n = \frac{1}{n}$ lends our law of 'iterated' logarithm, since $\|\mathbf{1}'\pi(x_n - x_*)\| = \mathcal{O}\left(\sqrt{\frac{\log\log n}{n}}\right)$. For step-sizes of type γ we get slower convergence rate and a law of logarithm with $\|\mathbf{1}'\pi(x_n - x_*)\| = \mathcal{O}\left(\sqrt{\frac{\log n}{n^\gamma}}\right)$. While this result describes the progression of the 'consensus'/agreement of iterates the next result describes the rate at which the disagreement component decays.

Lemma 4.2. *(Disagreement Error) Almost surely on $\mathcal{E}(x_*)$, for any $\delta > 0$*

$$\limsup_{n\to\infty} \frac{\|Qx_n\|}{\alpha_n(\log n)^{1+\delta}} \leq \sqrt{m}\kappa, \tag{11}$$

where κ is some deterministic constant.

For step sizes of type 1 or type γ notice the rate term $\alpha_n(\log n)^{1+\delta}$ is only affected through α_n and we get a law of logarithm in either case. This is notably different from the consensus case where an iterated logarithm rate is possible.

Notice the common $\sqrt{m}$ factor showing up in both bounds as the scaling factors for number of agents in the network. Whilst the temporal/step-size asymptotics vary in the two bounds yet they are independent of the number of agents or graph structure. Thus, two very different connectivity structures over the same set nodes could provably have similar rates.

Remark 4.3. *An important consequence of Lemma 4.1 and 4.2 is that the relative rate of decay of error in the disagreement space is squared of the error rate in the consensus space (up to logarithm factors). Thus the overall rate is essentially dictated by the agreement error in the consensus space.*

With these two ingredients at hand, our main result is arrived at via the following short calculation.

Proof of Theorem 2.1. Observe that

$$\|x_n - x_*\| \leq \|\mathbf{1}'\pi(x_n - x_*)\| + \|Qx_n\|. \tag{12}$$

From Lemma (4.1) and (4.2) we have for step sizes of Type γ, $\log t_{n+1} = O(\log n)$ and

$$\limsup_{n\to\infty} \frac{\|(x_n - x_*)\|}{(\alpha_n \log t_{n+1})^{1/2}} = \sqrt{m}\Lambda + \limsup_{n\to\infty} \frac{\alpha_n (\log n)^{1+\delta}}{(\alpha_n \log n^{1-\gamma})^{1/2}} = \sqrt{m}\Lambda, \tag{13}$$

where the last display holds because $\alpha_n = o((\log n)^{-1/2-\delta})$.
For gains of the Type 1, we have $\log t_{n+1} = O(\log\log n)$ and

$$\limsup_{n\to\infty} \frac{\|(x_n - x_*)\|}{(\alpha_n \log t_{n+1})^{1/2}} = \sqrt{m}\Lambda + \limsup_{n\to\infty} \frac{(\log n)^{1+\delta}}{n^{1/2}(\log\log n)^{1/2}} = \sqrt{m}\Lambda \tag{14}$$

where the last display holds because $\frac{(\log n)^{1+\delta}}{n^{1/2}} = o(1)$. □

We now focus on the details of the analysis for the first ingredient, agreement error.

4.1 Bound on agreement error $\|\mathbf{1}'\pi(x_n - x_*)\|$

We now state two lemmas on which the proof of Lemma 4.1 is based.

For $n \geq 0$, let

$$\psi_{n+1} := \sum_{k=0}^{n} \alpha_k \mathbf{1}'\pi M_{k+1} \mathrm{e}^{-(t_{n+1}-t_{k+1})A}. \tag{15}$$

Also, for $n \geq 0$, define Δ_n using the relation

$$\Delta_n := \mathbf{1}'\pi(x_n - x_*) - \psi_n. \tag{16}$$

We derive the following variants of [33, Lemma 1,3] for the distributed case. The reason we need to derive variants is because naively applying the vector bound would yield only a bound on the Frobenius norm and not exploit the specific consensus structure. The basic arguments of the proof are similar except for the following points. First, the noise sequence $\{M_n\}$ is matrix valued here, while the analogous variant in [33] is vector valued. Second, and more importantly, if $b > 4$, then our result is applicable even for non-square summable stepsizes.

The first result is a law of iterated logarithm for the propagated noise across the iterates.

Lemma 4.4. *Suppose $\gamma > 2/b$, where b is as in $\mathcal{A}_4$. Then, there exists some deterministic constant Λ such that, almost surely,*

$$\limsup_{n\to\infty} (\alpha_n \log t_{n+1})^{-1/2} \|\psi_{n+1}\| \leq \sqrt{m}\Lambda. \tag{17}$$

The second result for agreement error is the contributing term from the dynamics and discretization.

Lemma 4.5 (Upper Bound on Δ_n)**.** *Suppose $\mathcal{A}_2$, $\mathcal{A}_3$ and $\mathcal{A}_4$, then almost surely on $\mathcal{E}(x_*)$,*

$$\|\Delta_n\| = \mathcal{O}\left(\max\left\{e^{-\lambda'\sum_{k=0}^{n}\alpha_k}, \sum_{j=0}^{n} \alpha_j e^{-\lambda'\sum_{k=j+1}^{n}\alpha_k} \|\psi_j\|^a\right\}\right). \tag{18}$$

Furthermore,

1. *If step sizes are of the Type 1, then, almost surely on $\mathcal{E}(x_*)$,*

$$\|\Delta_n\| = \mathcal{O}\left(\max(n^{-\lambda'\alpha_0}; n^{-\frac{a}{2}}; n^{-1.5})(\log n)^{\frac{a}{2}+1}\right),$$

2. *If gains are of the Type γ with $2/b < \gamma < 1$, then, almost surely on $\mathcal{E}(x_*)$,*

$$\|\Delta_n\| = \mathcal{O}\left(\alpha_n\sqrt{\alpha_n \log(t_{n+1})} + (\sqrt{\alpha_n \log(t_{n+1})})^a\right).$$

We refer the reader to Appendix for proofs of Lemma 4.4 and 4.5 and present the proof of 4.1 here which is a direct consequence of Lemma 4.4 and 4.5.

Proof of Lemma 4.1. Observe that

$$\|\mathbf{1}'\pi(x_n - x_*)\| = \|\Delta_n\| + \|\psi_n\|. \tag{19}$$

From Lemma (4.4) and (4.5) we have for step sizes of Type γ

$$\limsup_{n\to\infty} \frac{\|\mathbf{1}'\pi(x_n - x_*)\|}{(\alpha_n \log t_{n+1})^{1/2}} = \sqrt{m}\Lambda + \limsup_{n\to\infty} \left(\alpha_n + \left(\sqrt{\alpha_n \log(t_{n+1})}\right)^{a-1}\right) = \sqrt{m}\Lambda,$$

where the last display holds because $\lim_{n\to\infty} \alpha_n = 0$, $\lim_{n\to\infty} \alpha_n \log(t_{n+1}) = 0$ and $a > 1$.

For gains of the Type 1, we have

$$\limsup_{n\to\infty} \frac{\|\mathbf{1}'\pi(x_n - x_*)\|}{(\alpha_n \log t_{n+1})^{1/2}} = \sqrt{m}\Lambda + \limsup_{n\to\infty} \frac{\max(n^{-\lambda'\alpha_0}; n^{-\frac{a}{2}}; n^{-1.5})(\log n)^{1+a/2}}{\sqrt{\frac{\gamma_0 \log n}{n}}} = \sqrt{m}\Lambda$$

where the last display holds because $\lambda'\alpha_0 > \frac{1}{2}$ and $a > 1$. Further, $\limsup_{n\to\infty} n^{-\delta} = 0$, $\forall \delta > 0$. □

We now turn to the detailed analysis of the second ingredient, disagreement error.

4.2 Bound on disagreement error $\|Qx_n\|$

For the analysis here, we define χ_n a term that explicitly marks the sequential propagation of noise through the iterates, for $n \geq -1$, let

$$\chi_{n+1} := \sum_{j=0}^{n} \alpha_j W^{n-j} Q M_{j+1} e^{-(t_{n+1}-t_{j+1})A}. \tag{20}$$

It is then easy to see that

$$\chi_{n+1} = W\chi_n e^{-\alpha_n A} + \alpha_n Q M_{n+1}, \quad n \geq 0. \tag{21}$$

Separately, for $n \geq 0$, let $\Gamma_n = Qx_n - \chi_n$. The following provides a law of logarithm type convergence rate for the noise contribution in the disagreement subspace.

Lemma 4.6. *Almost surely on $\mathcal{E}(x_*)$, for any $\delta > 0$*

$$\limsup_{n\to\infty} \frac{\|\chi_{n+1}\|}{\alpha_n (\log n)^{1+\delta}} \leq \sqrt{m}\kappa, \tag{22}$$

where κ is some deterministic constant.

Compared the Lemma 4.4 for the noise contribution in the consensus space, χ_n in the disagreement space has similar scaling $\sqrt{m}$ and exhibits a law of logarithm for step-sizes of type 1 as well as type type γ. The second result in the disagreement space is the contribution from the dynamics and the discretization as follows.

Lemma 4.7. *Almost surely on $\mathcal{E}(x_*)$,*

$$\|\Gamma_{n+1}\| = \mathcal{O}\left(\rho^n\right), \tag{23}$$

where ρ (< 1) is the spectral radius of W.

The convergence of γ is exponential in nature unlike the polynomial rates we have seen so far. Furthermore it is dictated solely by the Gossip matrix and independent of the step-size schedule. We refer readers to the Appendix for proofs of Lemma 4.6 and 4.7. While the overall disagreement error can thus be computed by the following calculation.

Proof of Lemma 4.2. Observe that

$$\|Qx_n\| \leq \|\Gamma_n\| + \|\chi_n\|. \tag{24}$$

Using lemmas 4.6 and 4.7, we have

$$\|Qx_n\| = \mathcal{O}\left(\rho^n\right) + \mathcal{O}\left(\alpha_n (\log n)^{1+\delta}\right) = \mathcal{O}\left(\alpha_n (\log n)^{1+\delta}\right). \tag{25}$$

□

4.3 Foundational Iterated Logarithm Lemma

In this section, we look at an auxiliary yet important result which assists us in proving the main convergence rates lemmas. We prove the following generalization of [10, Corollary 6.4.25] which is also applicable to step-sizes which are not necessarily square summable, a key limiting assumption in classical stochastic approximation theory.
Let $\mathrm{LL}(x) = \log\log(x)$.

Lemma 4.8. *For $n \geq 0$, let $U_{n+1} = \sum_{k=0}^{n} \phi_k \epsilon_{k+1}$, where $\{\epsilon_n\}$ is a real-valued martingale difference sequence adapted to a filtration $\{\mathcal{F}_n\}$, and $\{\phi_n\}$ is a sequence of real-valued scalars, again adapted to $\{\mathcal{F}_n\}$.*

Let $\{T_n\}$, also adapted to $\{\mathcal{F}_n\}$, be such that, for $n \geq 0$, $|\phi_n| \leq T_n$ a.s. and $\tau_n = \sum_{k=0}^{n} T_k^2$ satisfies $\lim_{n\to\infty} \tau_n = \infty$ a.s. Further, assume $\sup_{n\geq 0} \mathbb{E}[\epsilon_{n+1}^2|\mathcal{F}_n] \leq \sigma^2$ a.s. for some constant σ^2. Also, let $\beta > 0$ be such that $\sum_n T_n^{2+2\beta}\tau_n^{-1-\beta}[\mathrm{LL}(\tau_n)]^{\beta} < \infty$ and $\sup_{n\geq 0} \mathbb{E}\left[|\epsilon_{n+1}|^{2+2\beta}\middle|\mathcal{F}_n\right] < \infty$ a.s.

Then, almost surely,

$$\limsup [2\tau_n \mathrm{LL}\tau_n]^{-1/2}|U_{n+1}| \leq \sigma \tag{26}$$

Remark 4.9. *The condition $\sum_n T_n^{2+2\beta}\tau_n^{-1-\beta}[\mathrm{LL}(\tau_n)]^{\beta} < \infty$ differs from the one in [10, Corollary 6.4.25]; in that, it includes the additional term $[\mathrm{LL}(\tau_n)]^{\beta}$. The impact of this is that we no longer require β to be in $(0,1)$ as was the case in [10, Corollary 6.4.25]. Instead, β now take any positive value. This is precisely what allows Theorem 2.1 to be applicable even when the stepsizes are non-square summable.*

Remark 4.10. *The above result goes through even if we have $\limsup_{n\to\infty} \mathbb{E}[\epsilon_{n+1}^2|\mathcal{F}_n] \leq \sigma^2$ instead of $\sup_{n\geq 0} \mathbb{E}[\epsilon_{n+1}^2|\mathcal{F}_n] \leq \sigma^2$; cf. [33, Result 1].*

5 Discussion

We have analyzed distributed non-linear stochastic approximation algorithms for MARL and proved law of iterated logarithm type convergence rates. We propose a novel error decomposition for the distributed setting and generalize the existing law of iterated logarithm for martingales to not square summable sequences. Our results are the first of its kind in the distributed setting and give precise asymptotics on both agreement and disagreement components of error without requiring doubly stochastic matrix or square summable step-sizes.

From the perspective of stochastic approximation theory, an interesting future direction is to study effects of time varying communication protocols on convergence rates. While our framework is fairly general, one of the key limitations is the restriction of the curvature of the function in (5) and it will be interesting to relax that to find stationary points of a broader class of functions, e.g. the one considered in [56] for distributed stochastic optimization. On the MARL side, important algorithms like Q-learning [18] and its variants need more careful analysis and we believe our techniques would be instrumental to such future research.

References

[1] Michel Benaïm. Dynamics of stochastic approximation algorithms. In *Seminaire de probabilites XXXIII*, pages 1–68. Springer, 1999.

[2] Pascal Bianchi, Gersende Fort, and Walid Hachem. Performance of a distributed stochastic approximation algorithm. *IEEE Transactions on Information Theory*, 59(11):7405–7418, 2013.

[3] Nicholas M Boffi and Jean-Jacques E Slotine. A continuous-time analysis of distributed stochastic gradient. *Neural computation*, 32(1):36–96, 2020.

[4] Vivek S Borkar. *Stochastic approximation: a dynamical systems viewpoint*, volume 48. Springer, 2009.

[5] Ziyi Chen, Yi Zhou, and Rongrong Chen. Multi-agent off-policy td learning: Finite-time analysis with near-optimal sample complexity and communication complexity. *arXiv preprint arXiv:2103.13147*, 2021.

[6] Gal Dalal, Balázs Szörényi, Gugan Thoppe, and Shie Mannor. Finite sample analyses for td (0) with function approximation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[7] Dongsheng Ding, Xiaohan Wei, Zhuoran Yang, Zhaoran Wang, and Mihailo R Jovanovic. Fast multi-agent temporal-difference learning via homotopy stochastic primal-dual method. In *Optimization Foundations for Reinforcement Learning Workshop, 33rd Conference on Neural Information Processing Systems*, 2019.

[8] Thinh Doan, Siva Maguluri, and Justin Romberg. Finite-time analysis of distributed td (0) with linear function approximation on multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 1626–1635, 2019.

[9] Thinh T Doan, Siva Theja Maguluri, and Justin Romberg. Finite-time performance of distributed temporal-difference learning with linear function approximation. *SIAM Journal on Mathematics of Data Science*, 3(1):298–320, 2021.

[10] Marie Duflo. *Random iterative models*, volume 34. Springer Science & Business Media, 2013.

[11] Paulo Heredia and Shaoshuai Mou. Finite-sample analysis of multi-agent policy evaluation with kernelized gradient temporal difference. In *2020 59th IEEE Conference on Decision and Control (CDC)*, pages 5647–5652. IEEE, 2020.

[12] Paulo C Heredia and Shaoshuai Mou. Distributed multi-agent reinforcement learning by actor-critic method. *IFAC-PapersOnLine*, 52(20):363–368, 2019.

[13] Maxim Kaledin, Eric Moulines, Alexey Naumov, Vladislav Tadic, and Hoi-To Wai. Finite time analysis of linear two-timescale stochastic approximation with markovian noise. In *Conference on Learning Theory*, pages 2144–2203. PMLR, 2020.

[14] Soummya Kar, José MF Moura, and H Vincent Poor. QD-learning: A collaborative distributed strategy for multi-agent reinforcement learning through consensus + innovations. *IEEE Transactions on Signal Processing*, 61(7):1848–1862, 2013.

[15] Anastasia Koloskova, Nicolas Loizou, Sadra Boreiri, Martin Jaggi, and Sebastian Stich. A unified theory of decentralized sgd with changing topology and local updates. In *International Conference on Machine Learning*, pages 5381–5393. PMLR, 2020.

[16] Harold J Kushner and G Yin. Asymptotic properties of distributed and communicating stochastic approximation algorithms. *SIAM Journal on Control and Optimization*, 25(5):1266–1290, 1987.

[17] HJ Kushner and G Yin. Stochastic approximation algorithms for parallel and distributed processing. *Stochastics: An International Journal of Probability and Stochastic Processes*, 22(3-4):219–250, 1987.

[18] Martin Lauer and Martin Riedmiller. An algorithm for distributed reinforcement learning in cooperative multi-agent systems. In *In Proceedings of the Seventeenth International Conference on Machine Learning*. Citeseer, 2000.

[19] Donghwan Lee, Hyungjin Yoon, and Naira Hovakimyan. Primal-dual algorithm for distributed reinforcement learning: distributed gtd. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 1967–1972. IEEE, 2018.

[20] Jae Won Lee, Byoung-Tak Zhang, et al. Stock trading system using reinforcement learning with cooperative agents. In *Proceedings of the Nineteenth International Conference on Machine Learning*, pages 451–458, 2002.

[21] Xiangru Lian, Ce Zhang, Huan Zhang, Cho-Jui Hsieh, Wei Zhang, and Ji Liu. Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent. *arXiv preprint arXiv:1705.09056*, 2017.

[22] Qifeng Lin and Qing Ling. Decentralized td (0) with gradient tracking. *IEEE Signal Processing Letters*, 2021.

[23] Michael Littman and Justin Boyan. A distributed reinforcement learning scheme for network routing. In *Proceedings of the international workshop on applications of neural networks to telecommunications*, pages 55–61. Psychology Press, 2013.

[24] Michael L Littman. Markov games as a framework for multi-agent reinforcement learning. In *Machine learning proceedings 1994*, pages 157–163. Elsevier, 1994.

[25] Bo Liu, Ji Liu, Mohammad Ghavamzadeh, Sridhar Mahadevan, and Marek Petrik. Finite-sample analysis of proximal gradient td algorithms. In *UAI*, pages 504–513. Citeseer, 2015.

[26] Adwaitvedant S Mathkar and Vivek S Borkar. Nonlinear gossip. *SIAM Journal on Control and Optimization*, 54(3):1535–1557, 2016.

[27] Abdelkader Mokkadem, Mariane Pelletier, et al. Convergence rate and averaging of nonlinear two-time-scale stochastic approximation algorithms. *The Annals of Applied Probability*, 16(3):1671–1702, 2006.

[28] Gemma Morral, Pascal Bianchi, and Gersende Fort. Success and failure of adaptation-diffusion algorithms for consensus in multi-agent networks. In *53rd IEEE Conference on Decision and Control*, pages 1476–1481. IEEE, 2014.

[29] Gemma Morral, Pascal Bianchi, and Gersende Fort. Success and failure of adaptation-diffusion algorithms with decaying step size in multiagent networks. *IEEE Transactions on Signal Processing*, 65(11):2798–2813, 2017.

[30] Angelia Nedic, Asuman Ozdaglar, and Pablo A Parrilo. Constrained consensus and optimization in multi-agent networks. *IEEE Transactions on Automatic Control*, 55(4):922–938, 2010.

[31] Openai: Openai five. `https://openai.com/blog/openai-five/`. Accessed: 2021-05-24.

[32] Jun Ota. Multi-agent robot systems as distributed autonomous systems. *Advanced engineering informatics*, 20(1):59–70, 2006.

[33] Mariane Pelletier. On the almost sure asymptotic behaviour of stochastic algorithms. *Stochastic processes and their applications*, 78(2):217–244, 1998.

[34] Shi Pu and Angelia Nedić. Distributed stochastic gradient tracking methods. *Mathematical Programming*, pages 1–49, 2020.

[35] Shi Pu, Alex Olshevsky, and Ioannis Ch Paschalidis. Asymptotic network independence in distributed stochastic optimization for machine learning: Examining distributed and centralized stochastic gradient descent. *IEEE Signal Processing Magazine*, 37(3):114–122, 2020.

[36] Jineng Ren, Jarvis Haupt, and Zehua Guo. Communication-efficient hierarchical distributed optimization for multi-agent policy evaluation. *Journal of Computational Science*, 49:101280, 2021.

[37] Martin Riedmiller, Andrew Moore, and Jeff Schneider. Reinforcement learning for cooperating and communicating reactive agents in electrical power grids. In *Workshop on Balancing Reactivity and Social Deliberation in Multi-Agent Systems*, pages 137–149. Springer, 2000.

[38] Jeffrey S Rosenthal. Convergence rates for markov chains. *Siam Review*, 37(3):387–405, 1995.

[39] Kevin Scaman, Francis Bach, Sébastien Bubeck, Yin Lee, and Laurent Massoulié. Optimal convergence rates for convex distributed optimization in networks. *Journal of Machine Learning Research*, 20:1–31, 2019.

[40] Kevin Seaman, Francis Bach, Sébastien Bubeck, Yin Tat Lee, and Laurent Massoulié. Optimal algorithms for smooth and strongly convex distributed optimization in networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3027–3036. JMLR. org, 2017.

[41] Shai Shalev-Shwartz, Shaked Shammah, and Amnon Shashua. Safe, multi-agent, reinforcement learning for autonomous driving. *arXiv preprint arXiv:1610.03295*, 2016.

[42] Ohad Shamir and Nathan Srebro. Distributed stochastic optimization and learning. In *2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 850–857. IEEE, 2014.

[43] Miloš S Stanković, Marko Beko, and Srdjan S Stanković. Distributed gradient temporal difference off-policy learning with eligibility traces: Weak convergence. *IFAC-PapersOnLine*, 53(2):1563–1568, 2020.

[44] Jun Sun, Gang Wang, Georgios B Giannakis, Qinmin Yang, and Zaiyue Yang. Finite-time analysis of decentralized temporal-difference learning with linear function approximation. In *International Conference on Artificial Intelligence and Statistics*, pages 4485–4495. PMLR, 2020.

[45] Tao Sun, Tianyi Chen, Yuejiao Sun, Qing Liao, and Dongsheng Li. Decentralized markov chain gradient descent. *arXiv preprint arXiv:1909.10238*, 2019.

[46] Wesley Suttle, Zhuoran Yang, Kaiqing Zhang, Zhaoran Wang, Tamer Başar, and Ji Liu. A multi-agent off-policy actor-critic algorithm for distributed reinforcement learning. *IFAC-PapersOnLine*, 53(2):1549–1554, 2020.

[47] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[48] Richard S Sutton, Csaba Szepesvári, and Hamid Reza Maei. A convergent o (n) temporal-difference algorithm for off-policy learning with linear function approximation. In *NIPS*, 2008.

[49] Gugan Thoppe and Vivek Borkar. A concentration bound for stochastic approximation via alekseev's formula. *Stochastic Systems*, 9(1):1–26, 2019.

[50] John Tsitsiklis, Dimitri Bertsekas, and Michael Athans. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE transactions on automatic control*, 31(9):803–812, 1986.

[51] John N Tsitsiklis and Benjamin Van Roy. An analysis of temporal-difference learning with function approximation. *IEEE transactions on automatic control*, 42(5):674–690, 1997.

[52] Hoi-To Wai. On the convergence of consensus algorithms with markovian noise and gradient bias. In *2020 59th IEEE Conference on Decision and Control (CDC)*, pages 4897–4902. IEEE, 2020.

[53] Hoi-To Wai, Zhuoran Yang, Zhaoran Wang, and Mingyi Hong. Multi-agent reinforcement learning via double averaging primal-dual optimization. *arXiv preprint arXiv:1806.00877*, 2018.

[54] Yue Xu, Zengde Deng, Mengdi Wang, Wenjun Xu, Anthony Man-Cho So, and Shuguang Cui. Voting-based multi-agent reinforcement learning for intelligent iot. *IEEE Internet of Things Journal*, 2020.

[55] Kun Yuan, Qing Ling, and Wotao Yin. On the convergence of decentralized gradient descent. *SIAM Journal on Optimization*, 26(3):1835–1854, 2016.

[56] Sihan Zeng, Aqeel Anwar, Thinh Doan, Justin Romberg, and Arijit Raychowdhury. A decentralized policy gradient approach to multi-task reinforcement learning. *arXiv preprint arXiv:2006.04338*, 2020.

[57] Sihan Zeng, Thinh T Doan, and Justin Romberg. Finite-time analysis of decentralized stochastic approximation with applications in multi-agent and multi-task learning. *arXiv preprint arXiv:2010.15088*, 2020.

[58] Kaiqing Zhang, Zhuoran Yang, and Tamer Basar. Networked multi-agent reinforcement learning in continuous spaces. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 2771–2776. IEEE, 2018.

[59] Kaiqing Zhang, Zhuoran Yang, and Tamer Başar. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *arXiv preprint arXiv:1911.10635*, 2019.

[60] Kaiqing Zhang, Zhuoran Yang, Han Liu, Tong Zhang, and Tamer Basar. Fully decentralized multi-agent reinforcement learning with networked agents. In *International Conference on Machine Learning*, pages 5872–5881. PMLR, 2018.

[61] Kaiqing Zhang, Zhuoran Yang, Han Liu, Tong Zhang, and Tamer Basar. Finite-sample analysis for decentralized batch multi-agent reinforcement learning with networked agents. *IEEE Transactions on Automatic Control*, 2021.

[62] Xiaoxiao Zhao, Peng Yi, and Li Li. Distributed policy evaluation via inexact admm in multi-agent reinforcement learning. *Control Theory and Technology*, 18(4):362–378, 2020.

[63] Shaofeng Zou, Tengyu Xu, and Yingbin Liang. Finite-sample analysis for sarsa with linear function approximation. In *Advances in Neural Information Processing Systems*, pages 8665–8675, 2019.

6 Checklist

- Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?
  [Yes]
- Have you read the ethics review guidelines and ensured that your paper conforms to them?
  [Yes]
- Did you discuss any potential negative societal impacts of your work?
  [No] , we view this as foundational work studying theoretical convergence rates and don't anticipate imminent negative societal impacts.
- Did you describe the limitations of your work?
  [Yes] , see Section 5 for discussion on limitations and potential future research.
- Did you state the full set of assumptions of all theoretical results?
  [Yes] , we state and discuss all assumptions for the proofs in Section 2.
- Did you include complete proofs of all theoretical results?
  [Yes] , all proofs have been included either in Section 4 or the Appendix.
- Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)?
  [N/A]
- Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)?
  [N/A]
- Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)?
  [N/A]
- Did you include the amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)?
  [N/A]
- If your work uses existing assets, did you cite the creators?
  [N/A]
- Did you mention the license of the assets?
  [N/A]
- Did you include any new assets either in the supplemental material or as a URL?
  [N/A]
- Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
- Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content?
  [N/A]

- Did you include the full text of instructions given to participants and screenshots, if applicable?
  [N/A]
- Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable?
  [N/A]
- Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation?
  [N/A]