Toward Resilient Multi-Agent Actor-Critic Algorithms for Distributed Reinforcement Learning

Yixuan Lin Shripad Gade Romeil Sandhu Ji Liu

Abstract— **This paper considers a distributed reinforcement learning problem in the presence of Byzantine agents. The system consists of a central coordinating authority called "master agent" and multiple computational entities called "worker agents". The master agent is assumed to be reliable, while, a small fraction of the workers can be Byzantine (malicious) adversaries. The workers are interested in cooperatively maximize a convex combination of the honest (non-malicious) worker agents' long-term returns through communication between the master agent and worker agents. A distributed actor-critic algorithm is studied which makes use of entry-wise trimmed mean. The algorithm's communication-efficiency is improved by allowing the worker agents to send only a scalar-valued variable to the master agent, instead of the entire parameter vector, at each iteration. The improved algorithm involves computing a trimmed mean over only the received scalar-valued variable. It is shown that both algorithms converge almost surely.**

I. Introduction

Multi-agent reinforcement learning (MARL) involves a system of agents interacting with a common environment to learn to accomplish required task. In particular, agents take an action at each step, receive local (private) rewards and move to the next state. The action is decided based on the both the current state and the rewards. In general, agents only have access to local reward information, and because of privacy constraints [1], [2], agents are not allowed to share their local information with others.

In MARL problems, agents could be collaborative, competitive, or a mixture of the two. Under collaborative agents assumption, agents have the same goal, which is to maximize the long-term return over the network through interaction with the environment and communication among the agents. In [3]–[6], agents share a common reward function, then in [7]–[11], authors extended it and allowed agents to have heterogeneous reward functions, where, reward functions encode local information. In particular, these works focused on a decentralized setting. Different from the distributed setting, agents can exchange information with the neighbors on the network instead of communicating with the central controller. It is worth noting that the above works in decentralized settings allow each agent to know the actions of all other agents while treat its local rewards as private information, which is in contrast to some classic works in stochastic control [12]–[15] where the only information shared is the local rewards or locally computed statistics based upon local rewards and neighbors' rewards. As for the MARL in competitive and mixed settings, [16]–[19] paid more attention to the empirical works, and they do not have much theoretical convergence guarantees. Moreover, [20] discussed MARL in the distributed setting.

The works mentioned above assume agents will share correct information at each step. However, in realistic scenarios this may not happen due to plethora of reasons such as data corruption, communication delays and communication failures. The information received by the master agent may be grossly incorrect. In addition, the result may be worse if some worker agents are subjected to malicious manipulation and coordinated attacks. To model this, we consider Byzantine setting [21], where the behaviors of malicious/adversarial agents are completely arbitrary and the adversaries are allowed to cooperate with each other.

Recently, algorithmic approaches have been proposed for Byzantine resilience. Examples include geometric median in [22], coordinate-wise median (or marginal median) in [22]–[24], mean and coordinate-wise trimmed mean in [23], [25], Krum and multi-Krum in [24], [26]–[28], and Bulyan and multi-Bulyan in [28], [29].

Distributed algorithm often require worker agents to send entire parameter vectors to the master nodes, resulting in, high communication cost. In general, the communication cost will increase linearly in the number of workers and the complexity of the model. This is especially difficult in federated setting, where agents may have bounded communication capacities. We address this in our paper.

In this work, we focus on the distributed and collaborative MARL setting, which means the master agent collects information from, and broadcasts information back to – worker agents. With the motivation of the Byzantine problem in MARL in the distributed setting, we propose an algorithm by using the trimmed mean, so that worker agents can collaboratively maximize the long-term reward. Considering the communication cost, we propose one approach for the distributed situation Byzantine problem in which each worker agent broadcasts only one (scaled) entry of the vector at each step. Thus, communication cost at each iteration is significantly reduced.

The contribution of this paper is three-fold. First, we propose a distributed algorithm for solving the MARL problem. Second, we analyze the distributed algorithm using entry-

Y. Lin is with the Department of Applied Mathematics and Statistics at Stony Brook University (yixuan.lin.1@stonybrook.edu). S. Gade is with the Department of Electrical and Computer Engineering at University of Illinois at Urbana-Champaign (gade3@illinois.edu). R. Sandhu is with the Departments of Bioinformatics and Computer Science at Stony Brook University (romeil.sandhu@stonybrook.edu). J. Liu is with the Department of Electrical and Computer Engineering at Stony Brook University (ji.liu@stonybrook.edu).

The research of Sandhu was supported by the U.S. Air Force Office of Scientific Research (AFOSR) grant FA9550-18-1-0130 and National Science Foundation (NSF) grant ECCS-1749937.

wise trimmed mean to ensure Byzantine resilient reinforcement learning. Third, we propose a communication-efficient algorithm for resilience against a bounded fraction of Byzantine adversaries. In this algorithm, workers only sends a scalar to the master agent at each iteration. We present convergence (correctness) analysis for both the algorithms.

II. Distributed Reinforcement Learning

A. Multi-Agent Markov Decision Process

Consider a team of $N+1$ agents consisting of one master agent, denoted by 0, and N worker agents, denoted by $\mathcal{N} = \{1, 2, \ldots, N\}$, operating in a common environment. Each worker agent can exchange information only with the master agent. A multi-agent Markov decision process (MDP) is defined by a tuple $(\mathcal{S}, \{\mathcal{A}^i\}_{i\in\mathcal{N}}, P, \{R^i\}_{i\in\mathcal{N}})$ in which $\mathcal{S}$ is the state space shared by all the agents in $\mathcal{N}$, $\mathcal{A}^i$ is the action space of agent i, $P : \mathcal{S}\times\mathcal{A}\times\mathcal{S} \to [0,1]$ is the state transition probability of the MDP, and $R^i : \mathcal{S}\times\mathcal{A} \to \mathbb{R}$ is the local reward function for agent i, where $\mathcal{A} = \Pi_{i=1}^N \mathcal{A}^i$ is the joint action space. It is assumed that each agent can observe all others' actions, while each agent's rewards are private information and thus unobservable by any others.

At each discrete time $t \in \{0, 1, 2, \ldots\}$, given state s_t, each worker agent $i \in \mathcal{N}$ chooses its own action a_t^i according to a local policy $\pi^i : \mathcal{S}\times\mathcal{A}^i \to [0,1]$, i.e., the probability of choosing action a^i at state s_t. Note that the joint policy of all worker agents is denoted by $\pi\colon \mathcal{S}\times\mathcal{A} \to [0,1]$ which satisfies $\pi(s,a) = \prod_{i\in\mathcal{N}} \pi^i(s,a^i)$. After executing the action, each agent i will receive a reward r_{t+1}^i. We assume that the local policy for each agent i is parameterized by $\pi_{\theta^i}^i$, where $\theta^i \in \Theta^i$ is the parameter and $\Theta^i \subseteq \mathbb{R}^{m_i}$ is a compact set. Let $\theta = [(\theta^1)^\top \ \cdots \ (\theta^N)^\top]^\top \in \Theta$ where $\Theta = \prod_{i=1}^N \Theta^i$. The joint policy is thus given by $\pi_\theta(s,a) = \prod_{i\in\mathcal{N}} \pi_{\theta^i}^i(s,a_i)$. We impose the following standard assumption on the model and the policy parameterization [30], [31].

Assumption 1: For any $i \in \mathcal{N}$, $s \in \mathcal{S}$, and $a^i \in \mathcal{A}^i$, the policy function $\pi_{\theta^i}^i(s,a^i) > 0$ for any $\theta^i \in \Theta^i$ and is continuously differentiable with respect to the parameter θ^i over Θ^i. In addition, the Markov chain $\{s_t\}_{t\geq 0}$ is irreducible and aperiodic under any π_θ, with the stationary distribution denoted by d_θ.

The assumption implies that the Markov chain of the state-action pair $\{(s_t, a_t)\}_{t\geq 0}$ has a stationary distribution $d_\theta(s)\cdot\pi_\theta(s,a)$ for any $s \in \mathcal{S}$ and $a \in \mathcal{A}$.

The goal of the agents is to collaboratively find a policy π_θ that maximizes the averaged long-term return over the network based on local information, i.e,

$$\begin{aligned}\max_\theta \quad J(\theta) &= \lim_T \quad \frac{1}{T}\mathbb{E}\left(\sum_{t=0}^{T-1} \frac{1}{N}\sum_{i\in\mathcal{N}} r_{t+1}^i\right) \\ &= \sum_{s\in\mathcal{S}, a\in\mathcal{A}} d_\theta(s)\pi_\theta(s,a)\cdot\overline{R}(s,a), \qquad (1)\end{aligned}$$

where $\overline{R}(s,a) = N^{-1}\cdot\sum_{i\in\mathcal{N}} R^i(s,a)$ is the globally averaged reward function. Let $\overline{r}_t = N^{-1}\cdot\sum_{i\in\mathcal{N}} r_t^i$ and $\overline{R}(s,a) = \mathbb{E}[\overline{r}_{t+1} \mid s_t = s, a_t = a]$. Thus, under policy π_θ, the global relative action-value function can be written as

$$Q_\theta(s,a) = \sum_t \mathbb{E}\big[\overline{r}_{t+1} - J(\theta) \mid s_0 = s, a_0 = a, \pi_\theta\big],$$

the global relative state-value function $V_\theta(s)$ is defined as $V_\theta(s) = \sum_{a\in\mathcal{A}} \pi_\theta(s,a)Q_\theta(s,a)$, and the advantage function can be defined as $A_\theta(s,a) = Q_\theta(s,a) - V_\theta(s)$.

The work of [8] establishes the following policy gradient theorem for MARL (see Theorem 3.1 in [8]). For any $\theta\in\Theta$ and any agent $i \in \mathcal{N}$, we define the local advantage function $A_\theta^i\colon \mathcal{S}\times\mathcal{A}\to\mathbb{R}$ as

$$A_\theta^i(s,a) = Q_\theta(s,a) - \tilde{V}_\theta^i(s,a^{-i}), \qquad (2)$$

where $\tilde{V}_\theta^i(s,a^{-i}) = \sum_{a^i\in\mathcal{A}^i} \pi_{\theta^i}^i(s,a^i)\cdot Q_\theta(s,a^i,a^{-i})$ and a^{-i} denotes the actions of all agents except for i. Then, the gradient of $J(\theta)$ with respect to θ^i is given by

$$\begin{aligned}\nabla_{\theta^i} J(\theta) &= \mathbb{E}_{s\sim d_\theta, a\sim\pi_\theta}\left[\nabla_{\theta^i}\log\pi_{\theta^i}^i(s,a^i)\cdot A_\theta(s,a)\right] \\ &= \mathbb{E}_{s\sim d_\theta, a\sim\pi_\theta}\left[\nabla_{\theta^i}\log\pi_{\theta^i}^i(s,a^i)\cdot A_\theta^i(s,a)\right]. \qquad (3)\end{aligned}$$

B. Distributed Actor-Critic

In this section, we propose a multi-agent actor-critic algorithm for the distributed setting based on the algorithm in [8]. The algorithm is based on the local advantage function A_θ^i defined in (2), which requires estimating the action-value function Q_θ of policy π_θ. Consider $Q(\cdot,\cdot;\omega)\colon \mathcal{S}\times\mathcal{A}\to\mathbb{R}$, a family of functions parametrized by $\omega \in \mathbb{R}^K$, where $K \ll |\mathcal{S}|\cdot|\mathcal{A}|$. It is assumed that each agent i maintains its own parameter ω^i and uses $Q(\cdot,\cdot;\omega^i)$ to be the local estimate of Q_θ.

The algorithm consists of two steps, the actor and critic step. The critic step is based on temporal difference (TD) learning, followed by an averaging of all worker agents' parameter estimates. At each time t, each worker agent i transmits $\tilde{\omega}_t^i$, its estimate of ω, to the master agent, which then sends ω_t^0, the average among all worker agents, back to each of them. Specifically, the critic step iterates for each $i\in\mathcal{N}$ as follows:

$$\begin{cases} \mu_{t+1}^i = (1-\beta_{\omega,t})\cdot\mu_t^i + \beta_{\omega,t}\cdot r_{t+1}^i, \\ \tilde{\omega}_t^i = \omega_t^i + \beta_{\omega,t}\cdot\delta_t^i\cdot\nabla_\omega Q_t(\omega_t^i), \\ \omega_t^0 = \frac{1}{N}\sum_{j=1}^N \tilde{\omega}_t^j, \\ \omega_{t+1}^i = \omega_t^0, \end{cases} \qquad (4)$$

where ω_t^0 denotes the information sent to each worker from the master agent at time t, μ_t^i tracks the long-term return of agent i, $\beta_{\omega,t} > 0$ is the stepsize, $Q_t(\omega) = Q(s_t, a_t; \omega)$ for any ω, and the local action-value TD-error δ_t^i is defined as

$$\delta_t^i = r_{t+1}^i - \mu_t^i + Q_{t+1}(\omega_t^i) - Q_t(\omega_t^i). \qquad (5)$$

The actor step is motivated by (3) and is the same as that of Algorithm 1 in [8], which is

$$\theta_{t+1}^i = \theta_t^i + \beta_{\theta,t}A_t^i\psi_t^i, \qquad (6)$$

where $\beta_{\theta,t} > 0$ is the stepsize,

$$A_t^i = Q_t(\omega_t^i) - \sum_{a^i \in \mathcal{A}^i} \pi^i_{\theta^i_t}(s_t, a^i) Q(s_t, a^i, a_t^{-i}; \omega_t^i),$$

and $\psi_t^i = \nabla_{\theta^i} \log \pi^i_{\theta^i_t}(s_t, a_t^i)$.

The update (4) can be rewritten in state form as:

$$\begin{cases} \mu^i_{t+1} = (1-\beta_{\omega,t}) \cdot \mu_t^i + \beta_{\omega,t} \cdot r^i_{t+1}, \\ \tilde{\omega}_t^i = \omega_t^i + \beta_{\omega,t} \cdot \delta_t^i \cdot \nabla_\omega Q_t(\omega_t^i), \\ \omega_{t+1} = C\tilde{\omega}_t, \end{cases}$$

where $\omega_t = [(\omega_t^1)^\top \cdots (\omega_t^N)^\top]^\top$ and $C = \frac{1}{N}(\mathbf{1}_N \mathbf{1}_N^\top) \otimes I_K$. Here $\mathbf{1}_N$ denotes the N-dimensional vector whose entries all equal one, $\otimes$ denotes the Kronecker product, and I_K denotes the $K \times K$ identity matrix. Note that the communication between the N worker agents and the master agent is essentially the same as that among the N worker agents in the decentralized setting, as considered in [8], with a complete graph.

Next, we impose some standard and mild assumptions for the actor-critic algorithm; see [8] for detailed discussions on these assumptions.

Assumption 2: The instantaneous reward r_t^i is uniformly bounded for any $i \in \mathcal{N}$ and $t \geq 0$.

Assumption 3: The stepsizes $\beta_{\omega,t}$ and $\beta_{\theta,t}$ satisfy $\sum_t \beta_{\omega,t} = \sum_t \beta_{\theta,t} = \infty$ and $\sum_t \beta^2_{\omega,t} + \beta^2_{\theta,t} < \infty$. In addition, $\beta_{\theta,t} = o(\beta_{\omega,t})$ and $\lim_t \beta_{\omega,t+1} \cdot \beta^{-1}_{\omega,t} = 1$.

Assumption 4: For each agent i, the function $Q(s, a; \omega)$ is parametrized as $Q(s, a; \omega) = \omega^\top \phi(s, a)$, where $\phi(s, a) = [\phi_1(s, a) \ \cdots \ \phi_K(s, a)]^\top \in \mathbb{R}^K$ is the feature associated with (s, a). The feature vector $\phi(s, a)$ is uniformly bounded for any $s \in \mathcal{S}$ and $a \in \mathcal{A}$. The feature matrix $\Phi \in \mathbb{R}^{|\mathcal{S}|\cdot|\mathcal{A}|\times K}$ has full column rank, where the k-th column of Φ is $[\phi_k(s, a), s \in \mathcal{S}, a \in \mathcal{A}]^\top$ for any $k \in \{1, 2, \ldots, K\}$. For any $u \in \mathbb{R}^K$, $\Phi u \neq \mathbf{1}_K$.

Assumption 5: The update of the policy parameter θ_t^i includes a local projection operator, $\Gamma^i : \mathbb{R}^{m_i} \to \Theta^i \subset \mathbb{R}^{m_i}$, that projects any θ_t^i onto the compact set Θ^i. Also, $\Theta = \prod_{i=1}^N \Theta^i$ is large enough to include at least one local maximum of $J(\theta)$.

To simplify the notation, let $P^\theta(s', a' \,|\, s, a) = P(s' \,|\, s, a)\pi_\theta(s', a')$, $\mathrm{D}_\theta^{s,a} = \mathrm{diag}[d_\theta(s) \cdot \pi_\theta(s, a), s \in \mathcal{S}, a \in \mathcal{A}]$, and $R^i = [R^i(s, a), s \in \mathcal{S}, a \in \mathcal{A}]^\top \in \mathbb{R}^{|\mathcal{S}|\cdot|\mathcal{A}|}$. Define a vector $\hat{\Gamma}^i(\cdot)$ as

$$\hat{\Gamma}^i[g(\theta)] = \lim_{0<\eta\to 0} \{\Gamma^i[\theta^i + \eta \cdot g(\theta)] - \theta^i\}/\eta \tag{7}$$

for any $\theta \in \Theta$ and continuous function $g : \Theta \to \mathbb{R}^{\sum_{i\in\mathcal{N}} m_i}$. In case the limit above is not unique, $\hat{\Gamma}^i[g(\theta)]$ is defined as the set of all possible limit points of (7). Then, the following result is an immediate consequence of Theorems 4.6 and 4.7 in [8].

Theorem 1: Suppose that Assumptions 1-4 hold. Then, for any given policy π_θ with the sequence $\{\mu_t^i\}$ generated from (4), $\lim_t \sum_{i\in\mathcal{N}} \mu_t^i \cdot N^{-1} = J(\theta)$, $\lim_t \mu_t^i = \mu^i$ and $\lim_t \omega_t^i = \omega_\theta$ almost surely for any $i \in \mathcal{N}$, where $J(\theta)$ is the globally averaged return as defined in (1), and ω_θ is the unique solution to

$$\Phi^\top \mathrm{D}_\theta^{s,a} \left(\sum_{i=1}^N R^i - 1_{|\mathcal{S}||\mathcal{A}|} \sum_{i=1}^N \mu^i + (P^\theta - I)\Phi\omega \right) = 0.$$

Suppose further that Assumption 5 holds. Then, the sequence $\{\theta_t^i\}$ obtained from (6) converges almost surely to a point in the set of the asymptotically stable equilibria of

$$\dot{\theta}^i = \hat{\Gamma}^i \big[\mathbb{E}_{s_t \sim d_\theta, a_t \sim \pi_\theta} \big(A^i_{t,\theta} \cdot \psi^i_{t,\theta} \big) \big], \quad i \in \mathcal{N}.$$

C. Communication-Efficient Algorithm

In the algorithm described above, agents need to transmit entire vector $\tilde{\omega}$ to the master agent, which can be expensive (communication cost) when the size of $\tilde{\omega}$ is very large. A natural idea to reduce the communication cost is to allow each agent to transmit partial entries of its estimate at each step, as done in [32] for a decentralized setting.

We introduce the following communication-efficient variant, in which, at each iteration, worker agents transmit the same entry (coordinate) of their $\tilde{\omega}$ to the master agent, which then transmits the average of the entry back. To be more precise, suppose all agents transmit the k-th entry at time t, the critic step iterates as follows:

$$\begin{cases} \mu^i_{t+1} = (1-\beta_{\omega,t}) \cdot \mu_t^i + \beta_{\omega,t} \cdot r^i_{t+1}, \\ \tilde{\omega}_t^i = \omega_t^i + \beta_{\omega,t} \cdot \delta_t^i \cdot \nabla_\omega Q_t(\omega_t^i), \\ \omega_t^{0k} = \frac{1}{N} \sum_{j=1}^N \tilde{\omega}_t^{jk}, \\ \omega^{il}_{t+1} = \begin{cases} \omega_t^{0k} & \text{if } l = k, \\ \tilde{\omega}_t^{il} & \text{if } l \neq k, \end{cases} \end{cases} \tag{8}$$

where δ_t^i is defined in (5). The actor step is the same as (6). From Theorem 2 of [32], it is easy to see that for the update (8), Theorem 1 still holds.

III. Byzantine-Tolerant Algorithms

Our system allows a fraction of the worker agents to be Byzantine adversaries. Such malicious workers share adversarially perturbed updates with the master agent. In order for the system to reach a reasonably correct solution – or be resilient to Byzantine adversaries – we need to either identify the adversarial workers or reduce the effect of their erroneous updates. We use the latter technique.

Recall, the master agent is assumed to be reliable and there are at most f Byzantine workers. Throughout, we require,

$$N > 4f + 2.$$

To reduce the effect of faulty information, we use the coordinate-wise trimmed mean, as elaborated in Section III-A. Let $\mathcal{N}_g$ be the agent set with non-faulty (normal) agents, and $\mathcal{N}_b$ be the agent set with Byzantine agents. Without loss of generality, we assume that the first $|\mathcal{N}_g|$ worker agents are normal, i.e., $\mathcal{N}_g = \{1, 2, \ldots, |\mathcal{N}_g|\}$. Note, this assumption is only for ease of exposition; agents are unaware of such labeling.

A. Trimmed-mean-based Algorithm

The trimmed mean operation is a widely used robust estimation method. For a set of vectors $x^i \in \mathbb{R}^K, i \in \mathcal{N}$, the coordinate-wise f-trimmed mean is a vector with k-th entry equal to $\frac{1}{N-2f}\sum_{y\in\mathcal{V}^k} y$, where $\mathcal{V}^k$ is a subset of $\{x^{1k}, \cdots, x^{Nk}\}$ obtained by removing the largest and smallest f elements, and x^{ik} is the k-th entry of vector x^i.

To mitigate the effect of Byzantine worker agents, we augment the distributed algorithm presented in the last section with entry-wise trimmed mean. Agents share the vector $\tilde{\omega}_t^i$ with the master agent at each step, and the master agent sends back the coordinate-wise f-trimmed mean. Let $\mathcal{V}_t^k$ be the subset of $\{\tilde{\omega}_t^{1k}, \cdots, \tilde{\omega}_t^{Nk}\}$, obtained by removing the largest and smallest f elements, and $\mathcal{U}_t^k = \{i \in \mathcal{N} | \tilde{\omega}_t^{ik} \in \mathcal{V}_t^k\}$ be an agent set. Then, the critic step iterates for agent $i \in \mathcal{N}_g$ as follows:

$$\begin{cases} \mu_{t+1}^i = (1-\beta_{\omega,t})\cdot\mu_t^i + \beta_{\omega,t}\cdot r_{t+1}^i, \\ \tilde{\omega}_t^i = \omega_t^i + \beta_{\omega,t}\cdot\delta_t^i\cdot\nabla_\omega Q_t(\omega_t^i), \\ \omega_t^{0k} = \frac{1}{N-2f}\sum_{j\in\mathcal{U}_t^k}\tilde{\omega}_t^{jk}, \\ \omega_{t+1}^{ik} = \omega_t^{0k}, \end{cases} \tag{9}$$

where ω_t^{ik} is the k-th entry of agent i at time t and the local action-value TD-error δ_t^i is defined in (5). Besides, the actor step is:

$$\theta_{t+1}^i = \theta_t^i + \beta_{\theta,t} A_t^i \psi_t^i. \tag{10}$$

From the definition of trimmed mean, it is easy to see the following lemma.

Lemma 1: For each entry k and at any time t, the value of trimmed mean always lies in the interval $[\min_{i\in\mathcal{N}_g}\tilde{\omega}_t^{ik}, \max_{i\in\mathcal{N}_g}\tilde{\omega}_t^{ik}]$.

Lemma 2: If a graph is a complete graph with $N > 4f+2$ and we remove any $2f$ in-neighbors for each agent, then, any two agents in the network still share at least one in-neighbor in their remaining neighbor sets.

The proof of this Lemma is simple and thus omitted.

The distributed method for entry-wise f-trimmed mean computation, as discussed above, can also be emulated in the decentralized setting (over an incomplete graph). Moreover, based on Lemma 1, the coordinate-wise f-trimmed mean of all agents can be regarded as a vector of coordinate-wise convex combinations of normal agents.

A stochastic matrix S is a scrambling matrix if for any pair of distinct row indices i and j, there always exists a column index k such that both s_{ik} and s_{jk} are positive. The graph of scrambling matrix has the property that each pair of nodes share at least one in-neighbor.

Lemma 3: There exists a scrambling matrix B_t^k, for all coordinates $k \in \{1, 2, \ldots, K\}$, at each time t such that $[\omega_{t+1}^{1k}, \cdots, \omega_{t+1}^{|\mathcal{N}_g|k}]^\top = B_t^k[\tilde{\omega}_t^{1k}, \cdots, \tilde{\omega}_t^{|\mathcal{N}_g|k}]^\top$.

Proof: For any entry k, from Lemma 1, there exist two normal agents $i, j \in \mathcal{N}_g$, so that $\omega_t^{0k} = a_1\tilde{\omega}_t^{ik} + a_2\tilde{\omega}_t^{jk}$, where $a_1 \geq 0, a_2 \geq 0$ and $a_1 + a_2 = 1$. With Lemma 2, the weight matrix for entry k at time t can be a scrambling matrix, with i-th column being $a_1\mathbf{1}_K$, j-th column being $a_2\mathbf{1}_K$ and the remaining elements being 0. ■

Then, the critic step (9) for all agents in $\mathcal{N}_g$ can be rewritten as follows:

$$\begin{cases} \mu_{t+1}^i = (1-\beta_{\omega,t})\cdot\mu_t^i + \beta_{\omega,t}\cdot r_{t+1}^i, \\ \tilde{\omega}_t^i = \omega_t^i + \beta_{\omega,t}\cdot\delta_t^i\cdot\nabla_\omega Q_t(\omega_t^i), \\ \omega_{t+1}^{ik} = \sum_{j\in\mathcal{N}_g} b_t^k(i,j)\tilde{\omega}_t^{jk}, \end{cases}$$

where $B_t^k = [b_t^k(i,j)] \in \mathbb{R}^{|\mathcal{N}_g|\times|\mathcal{N}_g|}$ is a scrambling matrix for all $k \in \{1, 2, \ldots, K\}$. Let $B_t = \sum_{k=1}^K B_t^k \otimes (e_k e_k^\top)$. Then, for all normal agents,

$$\omega_{t+1,g} = B_t\tilde{\omega}_{t,g},$$

where $\omega_{t,g} = [(\omega_t^1)^\top \ \cdots \ (\omega_t^{|\mathcal{N}_g|})^\top]^\top$.

For $l \geq k$, define $B(l:k) = \Pi_{t=k}^l B_t$ and $B(l:k) = I_{|\mathcal{N}_g|K}$ for $k > l$. Then, based on Lemma 3, we have the following result, which is an immediate consequence of the property of scrambling matrices [33].

Lemma 4: Let $\{B_t^k\}$ are scrambling matrices, and $\{B_t = \sum_{k=1}^K B_t^k \otimes (e_k e_k^\top)\}$. Then, there exists a matrix B, so that $B = \lim_{T\to\infty}\mathbb{E}[\Pi_{t=k}^T B_t]$ for any k, and has the form $B = \mathbf{1}_{|\mathcal{N}_g|} \otimes [B^1, \cdots, B^{|\mathcal{N}_g|}]$, where $B^i \in \mathbb{R}^{K\times K}$ and $B\mathbf{1}_{|\mathcal{N}_g|K} = \mathbf{1}_{|\mathcal{N}_g|K}$. Moreover, $B(\infty:k) = \lim_{t\to\infty} B(t:k)$ exists w.p.1 and its rows are all equal. Furthermore, $\mathbb{E}[\|B(t:k) - B(\infty:k)\|_1] \to 0$ geometrically as $t-k \to \infty$, uniformly in k.

Theorem 2: Suppose that Assumptions 1-4 hold. Then, for any given policy π_θ with the sequence $\{\mu_t^i\}$ generated from (9), $\lim_t \mu_t^i = \mu^i = \mathbb{E}_{s,a}[R^i(s,a)]$ and $\lim_t \omega_t^i = \omega_\theta$ almost surely for any $i \in \mathcal{N}_g$, where ω_θ is the unique solution to

$$\sum_{i\in\mathcal{N}_g} B^i\Phi^\top D_\theta^{s,a}(R^i - 1_{|\mathcal{S}||\mathcal{A}|}\mu^i) + \Phi^\top D_\theta^{s,a}(P^\theta - I_{|\mathcal{S}||\mathcal{A}|})\Phi\omega = 0.$$

Suppose further that Assumption 5 holds. Then, the sequence $\{\theta_t^i\}$ obtained from (10) converges almost surely to a point in the set of the asymptotically stable equilibria of

$$\dot{\theta}^i = \hat{\Gamma}^i\left[\mathbb{E}_{s_t\sim d_\theta, a_t\sim\pi_\theta}\left(A_{t,\theta}^i\cdot\psi_{t,\theta}^i\right)\right], \quad i \in \mathcal{N}_g.$$

In the next section, we will modify the algorithm to significantly reduce the communication cost at each step. The above theorem is a special case of the theorem in the next section.

B. Communication-Efficient Resilient Algorithm

In this subsection, we propose an improved algorithm for communication efficient and Byzantine resilient MARL, where, workers share less entries (few coordinates) of the update at each iteration. Similar to the communication efficient update in Section II-C, we allow every worker to share the same one entry (coordinate) at each step, then the master agent returns the f-trimmed mean value of the received update to the workers. Workers only update this entry at this iteration.

At time t, if worker agents share the k-th entry, the critic step iterates as follows:

$$\begin{cases} \mu_{t+1}^i = (1-\beta_{\omega,t}) \cdot \mu_t^i + \beta_{\omega,t} \cdot r_{t+1}^i, \\ \tilde{\omega}_t^i = \omega_t^i + \beta_{\omega,t} \cdot \delta_t^i \cdot \nabla_\omega Q_t(\omega_t^i), \\ \omega_t^{0k} = \frac{1}{N-2f} \sum_{j \in \mathcal{U}_t^k} \tilde{\omega}_t^{jk}, \\ \omega_{t+1}^{il} = \begin{cases} \omega_t^{0k} & \text{if } l = k, \\ \tilde{\omega}_t^{il} & \text{if } l \neq k, \end{cases} \end{cases} \tag{11}$$

where δ_t^i is defined in (5). As we mentioned before, we can change the update from the distributed setting to decentralized setting. Then, there exists a matrix $\tilde{B}_t = \sum_{k=1}^K \tilde{B}_t^k \otimes (e_k e_k^\top)$, where $\{\tilde{B}_t^k\}, \forall k = 1, \cdots, K$ are scrambling matrices. Then we have the update for all normal agents as follows:

$$\omega_{t+1,g} = \tilde{B}_t \tilde{\omega}_{t,g}.$$

From Lemma 4, there exists a matrix $\tilde{B} = \lim_{T\to\infty} \Pi_{t=k}^T \tilde{B}_t$ with the form $\tilde{B} = \mathbf{1}_{|\mathcal{N}_g|} \otimes [\tilde{B}^1, \cdots, \tilde{B}^{|\mathcal{N}_g|}]$. As for the actor step,

$$\theta_{t+1}^i = \theta_t^i + \beta_{\theta,t} A_t^i \psi_t^i. \tag{12}$$

Theorem 3: Suppose that Assumptions 1-4 hold. Then, for any given policy π_θ, with the sequence $\{\mu_t^i\}$ generated from (11), we have $\lim_t \mu_t^i = \mu^i = \mathbb{E}_{s,a}[R^i(s,a)]$ and $\lim_t \omega_t^i = \omega_\theta$ almost surely for any $i \in \mathcal{N}_g$, where ω_θ is the unique solution to

$$\sum_{i\in\mathcal{N}_g} \tilde{B}^i \Phi^\top D_\theta^{s,a} (R^i - 1_{|\mathcal{S}||\mathcal{A}|} \mu^i) \\ + \Phi^\top D_\theta^{s,a} (P^\theta - I_{|\mathcal{S}||\mathcal{A}|}) \Phi \omega = 0.$$

Suppose further that Assumption 5 holds. Then, the sequence $\{\theta_t^i\}$ obtained from (12) converges almost surely to a point in the set of the asymptotically stable equilibria of

$$\dot{\theta}^i = \hat{\Gamma}^i \big[\mathbb{E}_{s_t \sim d_\theta, a_t \sim \pi_\theta} \big(A_{t,\theta}^i \cdot \psi_{t,\theta}^i \big) \big], \quad i \in \mathcal{N}_g.$$

To prove the above theorem, we need the following results.

Lemma 5: Under Assumptions 1 and 2, the sequence $\{\mu_t^i\}$ generated as in (11) is bounded almost surely.

Proof: The proof of the lemma is the same as that of Lemma 5.2 in [8]. ■

Let $\{\mathcal{F}_t\}$ be the filtration with $\mathcal{F}_t = \sigma(r_\tau, \mu_\tau, \omega_\tau, s_\tau, a_\tau, \tilde{B}_{\tau-1}, \tau < t)$.

Lemma 6: Under Assumptions 1-4, for the normal agent $i \in \mathcal{N}_g$, the sequence $\{\omega_t^i\}$ generated in (11) is bounded almost surely, i.e., $\sup_t \|\omega_t^i\| < \infty$.

Proof: Recall that the update of that $\omega_{t+1,g} = \tilde{B}_t \cdot (\omega_{t,g} + \beta_{\omega,t} \cdot U_{t,g})$ given in (11), where $U_{t,g} = [(u_t^1)^\top, \cdots, (u_t^{|\mathcal{N}_g|})^\top]^\top$ and $u_t^i = \delta_t^i \phi_t$. For $i \in \mathcal{N}_g$, let $h^i(\omega_t^i, \mu_t^i, s_t, a_t) = \mathbb{E}(u_t^i | \mathcal{F}_t)$, $M_{t+1}^i = u_t^i - \mathbb{E}(u_t^i | \mathcal{F}_t)$. Since the Markov chain $\{(s_t, a_t)\}_{t\geq 0}$ is irreducible and periodic given policy π_θ, we have $\bar{h}^i(\omega_t^i, \mu_t^i) = \mathbb{E}_{s_t \sim d_\theta, a_t \sim \pi_\theta} [h^i(\omega_t^i, \mu_t^i, s_t, a_t)] = \Phi^\top D_\theta^{s,a} [R^i - \mathbf{1}_{|\mathcal{S}||\mathcal{A}|} \otimes \mu_t^i + (P^\theta \Phi - \Phi) \omega_t^i]$.

From Assumptions 2 and 4, and Lemma 5, we know that $\exists K_1, K_2 > 0$, s.t. $\|\phi_t^k\|_\infty \leq K_1$ and $\|r_{t+1}^i - \mu_t^i\| \leq K_2, \forall k, i$. Thus, $\exists K_3 > 0$ such that $\|\bar{h}^i(\omega_t^i, \mu_t^i) - h^i(\omega_t^i, \mu_t^i, s_t, a_t)\|^2 \leq K_3 \cdot (1 + \|\omega_{t,g}\|^2)$. Moreover, we know $h^i(\omega_t^i, \mu_t^i, s_t, a_t)$ is Lipschitz continuous in ω_t^i, and M_{t+1}^i is martingale difference sequence. Since each B_t^k is column stochastic, it has bounded norm. Thus, by Theorem A.2 in [8], it follows that for $i \in \mathcal{N}_g$, ω_t^i is bounded almost surely. ■

Proposition 1: Under Assumptions 2-4, the following ODE captures the asymptotic behavior of (11):

$$\dot{\mu} = -\mu + \mathbb{E}_{s,a}[R(s,a)],$$

where $\mu = [\mu^1, \cdots, \mu^{|\mathcal{N}_g|}]^\top$, and $R(s,a) = [R^1(s,a), \cdots, R^{|\mathcal{N}_g|}(s,a)]^\top$. Then, the equivalent point in the long run for μ is $\mu = \mathbb{E}_{s,a}[R(s,a)]$.

Proof: The update for μ_t^i is

$$\mu_{t+1}^i = \mu_t^i + \beta_{\omega,t} \mathbb{E}[r_{t+1}^i - \mu_t^i | \mathcal{F}_t] + \beta_{\omega,t} \xi_{t+1}^i,$$

where $\xi_{t+1}^i = r_{t+1}^i - \mathbb{E}(r_{t+1}^i | \mathcal{F}_t)$. Note that $\mathbb{E}[r_{t+1}^i - \mu_t^i | \mathcal{F}_t]$ is Lipschitz continuous in μ_t^i, and ξ_t^i is a martingale difference sequence. Based on Lemma 5, from Theorem B.2 in [8], μ_t^i will converge to a point μ^i almost surely in the long run, and the point satisfies the ODE: $\dot{\mu}^i = -\mu^i + \mathbb{E}_{s,a}[R^i(s,a)]$ for all normal agents $i \in \mathcal{N}_g$. ■

We are now in a position to prove Theorem 3.

Proof of Theorem 3: With Assumptions 2–4 and Lemma 6, by using Theorem 3.2 in [34], we have that ω_t^i converges to ω_θ almost surely for all normal agents $i \in \mathcal{N}_g$, where ω_θ is the unique equilibrium of the ODE

$$\dot{\omega} = \Phi^\top D_\theta^{s,a} (P^\theta - I_{|\mathcal{S}||\mathcal{A}|}) \Phi \omega \\ + \sum_{i=1}^{|\mathcal{N}_g|} \tilde{B}^i \Phi^\top D_\theta^{s,a} (R^i - 1_{|\mathcal{S}||\mathcal{A}|} \mu^i]).$$

Combining Proposition 1, the following ODEs can capture the asymptotic behavior of (11),

$$\begin{cases} \dot{\mu} = -\mu + \mathbb{E}_{s,a}[R(s,a)], \\ \dot{\omega} = \Phi^\top D_\theta^{s,a} (P^\theta - I_{|\mathcal{S}||\mathcal{A}|}) \Phi \omega \\ \qquad + \sum_{i=1}^{|\mathcal{N}_g|} \tilde{B}^i \Phi^\top D_\theta^{s,a} (R^i - 1_{|\mathcal{S}||\mathcal{A}|} \mu^i]), \end{cases} \tag{13}$$

Note that from the Perron-Frobenius theorem and Assumption 1, the stochastic matrix P^θ has a simple eigenvalue of 1, and the remaining eigenvalues have real parts less than 1. Hence, since from Assumption 4 Φ is full column rank, $\Phi^\top D_\theta^{s,a} (P^\theta - I) \Phi$ has all eigenvalues with negative real parts but one zero. Moreover, the eigenvalue of zero has eigen-vector v when it satisfies $\Phi v = \alpha \mathbf{1}$ for some $\alpha \neq 0$. However, from Assumption 4 we know this will not happen. Hence, the ODE (13) is globally asymptotically stable and has its equilibrium satisfying $\mu = \mathbb{E}_{s,a}[R(s,a)]$ and

$$\Phi^\top D_\theta^{s,a}(P^\theta - I_{|\mathcal{S}||\mathcal{A}|})\Phi\omega + \sum_{i=1}^{|\mathcal{N}_g|} \tilde{B}^i \Phi^\top D_\theta^{s,a}(R^i - 1_{|\mathcal{S}||\mathcal{A}|}\mu^i]) = 0.$$

Note that the solution for ω has the form $\omega_\theta + lv$ with any $l \in \mathbb{R}$ and $v \in \mathbb{R}^K$ such that $\Phi v = \mathbf{1}_K$, where ω_θ follows that $\Phi^\top D_\theta^{s,a}(P^\theta - I_{|\mathcal{S}||\mathcal{A}|})\Phi\omega_\theta + \sum_{i=1}^{|\mathcal{N}_g|} \tilde{B}^i \Phi^\top D_\theta^{s,a}(R^i - 1_{|\mathcal{S}||\mathcal{A}|}\mu^i]) = 0$. By Assumption 4, ω_θ is the unique solution.

As for the actor step convergence, the proof is the same as that of Theorem 4.7 in [8]. ■

IV. Conclusions

In this paper, we propose an actor-critic algorithm for MARL in the distributed setting with resilience against bounded fraction of Byzantine worker agents. We show that this algorithm reduces the effect of Byzantine agents and guarantees existence of a limiting point for the policy parameters in the long run. Moreover, we have proposed a communication-efficient algorithm for Byzantine resilient MARL. Workers only share a coordinate (scalar value) of the parameter vector at each iteration. We show convergence (correctness) of our algorithm and resilience to Byzantine adversaries even under such stringent communication constraints. It is fairly straightforward to extend the algorithms and their convergence results to the case where each agent transmits more than one entry at each step. Future directions include characterizing the equilibrium point and extending our algorithms to the decentralized setting (peer-to-peer network) with the Byzantine agents. We intend to perform exhaustive numerical experiments with neural networks as function approximators in future (omitted here due to space limitations).

References

[1] S. Gade and N.H. Vaidya. Privacy-preserving distributed learning via obfuscated stochastic gradients. In *57th IEEE Conference on Decision and Control*, pages 184–191, 2018.

[2] S. Gade and N.H. Vaidya. Private optimization on networks. In *2018 American Control Conference*, pages 1402–1409, 2018.

[3] C. Boutilier. Planning, learning and coordination in multi-agent decision processes. In *Conference on Theoretical Aspects of Rationality and Knowledge*, pages 195–210, 1996.

[4] M. Lauer and M. Riedmiller. An algorithm for distributed reinforcement learning in cooperative multi-agent systems. In *International Conference on Machine Learning*, pages 535–542, 2000.

[5] M.L. Littman. Value-function reinforcement learning in Markov games. *Cognitive Systems Research*, 2(1):55–66, 2001.

[6] X. Wang and T. Sandholm. Reinforcement learning to play an optimal Nash equilibrium in team Markov games. In *Advances in Neural Information Processing Systems*, pages 1603–1610, 2003.

[7] S. Kar, J.M. Moura, and H.V. 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.

[8] K. Zhang, Z. Yang, H. Liu, T. Zhang, and T. Başar. Fully decentralized multi-agent reinforcement learning with networked agents. In *International Conference on Machine Learning*, pages 5872–5881, 2018.

[9] W. Suttle, Z. Yang, K. Zhang, Z. Wang, T. Başar, and J. Liu. A multi-agent off-policy actor-critic algorithm for distributed reinforcement learning. In *21st IFAC World Congress*, 2020. to appear.

[10] D.H. Lee, H.J. Yoon, and N. Hovakimyan. Primal-dual algorithm for distributed reinforcement learning: Distributed GTD. *IEEE Conference on Decision and Control*, pages 1967–1972, 2018.

[11] T.T. Doan, S.T. Maguluri, and J.K. Romberg. Finite-time analysis of distributed TD(0) with linear function approximation on multi-agent reinforcement learning. In *36th International Conference on Machine Learning*, pages 1626–1635, 2019.

[12] A. Nayyar, A. Mahajan, and D. Teneketzis. Decentralized stochastic control with partial history sharing: A common information approach. *IEEE Transactions on Automatic Control*, 58(7):1644–1658, 2013.

[13] C. Amato, G. Chowdhary, A. Geramifard, N. Üre, and M.J. Kochenderfer. Decentralized control of partially observable Markov decision processes. In *52nd IEEE Conference on Decision and Control*, pages 2398–2405, 2013.

[14] K. Hsu and S. Marcus. Decentralized control of finite state Markov processes. *IEEE Transactions on Automatic Control*, 27(2):426–431, 1982.

[15] J. Wu and S. Lall. Sufficient statistics for multi-agent decision problems. In *52nd Annual Allerton Conference on Communication, Control, and Computing*, pages 467–474, 2014.

[16] J. Hu and M.P. Wellman. Nash Q-learning for general-sum stochastic games. *Journal of Machine Learning Research*, 4(11):1039–1069, 2003.

[17] J. Foerster, Y.M. Assael, N. Freitas, and S. Whiteson. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 2137–2145, 2016.

[18] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Advances in Neural Information Processing systems*, pages 6379–6390, 2017.

[19] S. Omidshafiei, J. Pazis, C. Amato, J. P. How, and J. Vian. Deep decentralized multi-task multi-agent reinforcement learning under partial observability. In *International Conference on Machine Learning*, pages 2681–2690, 2017.

[20] T. Chen, K. Zhang, G.B. Giannakis, and T. Başar. Communication-efficient distributed reinforcement learning. *arXiv preprint arXiv:1812.03239*, 2018.

[21] L. Lamport, R. Shostak, and M. Pease. The Byzantine generals problem. *ACM Transactions on Programming Languages and System*, 4(3):382–401, 1982.

[22] C. Xie, O. Koyejo, and I. Gupta. Generalized Byzantine tolerant SGD. *arXiv preprint arXiv:1802.10116*, 2018.

[23] D. Yin, Y. Chen, K. Ramchandran, and P. Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. pages 5650–5659, 2018.

[24] C. Xie, S. Koyejo, and I. Gupta. Fall of empires: Breaking Byzantine-tolerant SGD by inner product manipulation. *arXiv preprint arXiv:1903.03936*, 2019.

[25] L. Bottou. Large-scale machine learning with stochastic gradient descent. In *19th International Conference on Computational Statistics*, pages 177–186, 2010.

[26] P. Blanchard, E. El Mhamdi, R. Guerraoui, and I. Stainer. Machine learning with adversaries: Byzantine tolerant gradient descent. In *Advances in Neural Information Processing Systems*, pages 119–129. 2017.

[27] P. Blanchard, E. El Mhamdi, R. Guerraoui, and J. Stainer. Byzantine-tolerant machine learning. *arXiv preprint arXiv:1703.02757*, 2017.

[28] E. El Mhamdi and R. Guerraoui. Fast and secure distributed learning in high dimension. *arXiv preprint arXiv:1905.04374*, 2019.

[29] E. El Mhamdi, R. Guerraoui, and S. Rouault. The hidden vulnerability of distributed learning in Byzantium. In *35th International Conference on Machine Learning*, pages 3521–3530, 2018.

[30] V.R. Konda and J.N. Tsitsiklis. Actor-critic algorithms. In *Advances in Neural Information Processing Systems*, pages 1008–1014, 2000.

[31] S. Bhatnagar, R. Sutton, M. Ghavamzadeh, and M. Lee. Natural actor-critic algorithms. *Automatica*, 45(11):2471–2482, 2009.

[32] Y. Lin, K. Zhang, Z. Yang, Z. Wang, T. Başar, R. Sandhu, and J. Liu. A communication-efficient multi-agent actor-critic algorithm for distributed reinforcement learning. In *58th IEEE Conference on Decision and Control*, pages 5562–5567, 2019.

[33] E. Seneta. *Non-negative Matrices and Markov Chains*. Springer, 1981.

[34] H.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.