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) agents. 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-
wise trimmed mean to ensure Byzantine resilient reinforce-
ment learning. Third, we propose a communication-efficient
algorithm for resilience against a bounded fraction of Byzan-
tine 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 $(\mathcal{S}, \{A_i\}_{i \in \mathcal{N}}, P, \{R^i\}_{i \in \mathcal{N}})$ in which $\mathcal{S}$
is the state space shared by all the agents in $\mathcal{N}$, $A_i$ is the
action space of agent $i$, $P: \mathcal{S} \times A \times \mathcal{S} \rightarrow [0, 1]$ is the state
transition probability of the MDP, and $R^i: \mathcal{S} \times A \rightarrow \mathbb{R}$ is the
local reward function for agent $i$, where $A = \prod_{i=1}^{N} A_i$
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 $\pi^i: \mathcal{S} \times A_i \rightarrow [0, 1]$, 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 $\pi: \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$ 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 $r^i_{t+1}$. We assume that the local
policy for each agent $i$ is parameterized by $\pi^i_{\theta^i}$, 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 $\pi^i(s, a) = \prod_{i \in \mathcal{N}} \pi^i_{\theta^i}(s, a_i)$. 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 $a^i \in A_i$, the
policy function $\pi^i_{\theta^i}(s, a^i) > 0$ for any $\theta^i \in \Theta^i$ and is
continuously differentiable with respect to the parameter $\theta^i$
over $\Theta^i$. In addition, the Markov chain $\{s_t\}_{t \geq 0}$ is irreducible and aperiodic under any $\pi^i_{\theta^i}$, with the stationary distribution denoted by $d^i_{\theta^i}$.

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

The goal of the agents is to collaboratively find a policy $\pi^i_{\theta^i}$ that maximizes the averaged long-term return over the
network based on local information, i.e.,

$$\max_{\theta} J(\theta) = \lim_{T \to \infty} \frac{1}{T} \mathbb{E} \left( \sum_{t=0}^{T-1} \frac{1}{N} \sum_{i \in \mathcal{N}} r^i_{t+1} \right)$$

$$= \sum_{s \in \mathcal{S}, a \in A} d^i_{\theta}(s) \pi^i_{\theta^i}(s, a) \cdot \mathcal{R}(s, a),$$

(1)

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

$$Q^i(\theta) = \sum_{s} \mathbb{E}_{[\bar{r}_{t+1}]} J(\theta) | s_0 = s, a_0 = a, \pi^i_{\theta^i},$$

the global relative state-value function $V^i(\theta)$ is defined as
$V^i(\theta) = \sum_{s} \pi^i_{\theta^i}(s, a) Q^i(\theta, a)$, and the advantage function can be defined as $A^i(\theta, a) = Q^i(\theta, a) - V^i(\theta)$.

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
$A^i_b: \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ as

$$A^i_b(s, a) = Q^i(s, a) - V^i_b(s, a^{-i}),$$

(2)

where $V^i_b(s, a^{-i}) = \sum_{a^i \in A_i} \pi^i_{\theta^i}(s, a^i) \cdot Q^i(s, a^i, a^{-i})$ and $a^{-i}$ denotes the actions of all agents except for $i$. Then, the gradient of $J(\theta)$ with respect to $\theta^i$ is given by

$$\nabla_{\theta^i} J(\theta) = \mathbb{E}_{s \sim d^i_{\theta}, a \sim \pi^i_{\theta^i}} \left[ \nabla_{\theta^i} \log \pi^i_{\theta^i}(s, a^i) \cdot A^i_b(s, a) \right]$$

$$= \mathbb{E}_{s \sim d^i_{\theta}, a \sim \pi^i_{\theta^i}} \left[ \nabla_{\theta^i} \log \pi^i_{\theta^i}(s, a^i) \cdot A^i_b(s, a) \right].$$

(3)

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
$A^i_b$ defined in (2), which requires estimating the action-value
function $Q^i_b$ of policy $\pi^i_{\theta^i}$. Consider $Q^i(\cdot; \cdot; \omega): \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$, 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 $\omega^i$ and uses $Q^i(\cdot; \cdot; \omega)$ to be the local estimate of $Q^i_b$.

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 $i$
sends $\omega^i_t$, its estimate of $\omega$, to the master agent, which
then sends $\omega^0_t$, the average among all worker agents, back
to each of them. Specifically, the critic step iterates for each
$i \in \mathcal{N}$ as follows:

$$\mu^i_{t+1} = (1 - \beta_{\omega, t}) \cdot \mu^i_t + \beta_{\omega, t} \cdot r^i_{t+1},$$

$$\tilde{\omega}^i_t = \omega^i_t + \beta_{\omega, t} \cdot \delta^i_t \cdot \nabla_{\omega} Q^i_t(\omega^i_t),$$

$$\omega^0_t = \frac{1}{N} \sum_{j=1}^{N} \tilde{\omega}^i_t,$$

$$\omega^i_{t+1} = \omega^i_t,$$

(4)

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

$$\delta^i_t = r^i_{t+1} - \mu^i_t + Q^i_{t+1}(\omega^i_t) - Q^i_t(\omega^i_t).$$

(5)

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

$$\theta^i_{t+1} = \theta^i_t + \beta_{\theta, t} A^i_b^i \omega^i_{t+1},$$

(6)
where \( \beta_{\theta,t} > 0 \) is the stepsize,

\[
A_i^t = Q_i(\omega_i^t) - \sum_{a_i \in A_i^t} \pi_i^{t}(s_t, a_i) Q(s_t, a_i; \omega_t),
\]

and \( \psi_i = \nabla_{\theta_i} \log \pi_i^{t}(s_t, a_t^i) \).

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

\[
\begin{align*}
\mu_{t+1}^i &= (1 - \beta_{\omega,t}) \cdot \mu_t^i + \beta_{\omega,t} \cdot r_{t+1}^i, \\
\dot{\omega}_{t} &= \omega_{t} + \beta_{\omega,t} \cdot \delta_i \cdot \nabla \omega Q_i(\omega_t^i), \\
\omega_{t+1} &= C \omega_{t},
\end{align*}
\]

where \( \omega_t = [(\omega_s^T)^T \cdots (\omega_N^T)^T]^T \) and \( C = \frac{1}{N} (1_N 1_N^T) \otimes I_K \).

Note, this assumption is only for ease of exposition; agents are unaware of such labeling.

From (4), the result is an immediate consequence of Theorems 4.6 and 4.7.

In case the limit above is not unique, the globally averaged return as defined in (1), and \( \omega_0 \) is the unique solution to

\[
\Phi^T D_{\theta}^{\omega} \left( \sum_{i=1}^{N} R_i - 1_{|S| \times |A|} \sum_{i=1}^{N} \mu_i + (P^0 - I) \Phi \omega \right) = 0.
\]

C. Communication-Efficient Algorithm

In the algorithm described above, agents need to transmit entire vector \( \omega \) to the master agent, which can be expensive (communication cost) when the size of \( \omega \) 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 \( \omega \) 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:

\[
\begin{align*}
\mu_{t+1}^i &= (1 - \beta_{\omega,t}) \cdot \mu_t^i + \beta_{\omega,t} \cdot r_{t+1}^i, \\
\dot{\omega}_{t}^k &= \omega_{t}^k + \beta_{\omega,t} \cdot \delta_t^k \cdot \nabla \omega Q_i(\omega_t^i), \\
\omega_{t+1}^k &= \begin{cases} \omega_t^k & \text{if } l = k, \\
\tilde{\omega}_t^l & \text{if } l \neq k,
\end{cases}
\end{align*}
\]

where \( \delta_t^k \) 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 \( N_g \) be the agent set with non-faulty (normal) agents, and \( N_b \) be the agent set with Byzantine agents. Without loss of generality, we assume that the first \( |N_g| \) worker agents are normal, i.e., \( N_g = \{1, 2, \ldots, |N_g|\} \). 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 \( \{ x^{1k}, \ldots, x^{NK} \} \) 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 \( \tilde{\omega}^i_t \) with the master agent at each step, and the master agent sends back the coordinate-wise \( f \)-trimmed mean. Let \( \mathcal{V}^k_t \) be the subset of \( \{ \tilde{\omega}^{1k}_t, \ldots, \tilde{\omega}^{NK}_t \} \), obtained by removing the largest and smallest \( f \) elements, and \( \mathcal{U}^k_t = \{ i \in \mathcal{N} | \tilde{\omega}^{ik}_t \in \mathcal{V}^k_t \} \) be an agent set. Then, the critic step iterates for agent \( i \in \mathcal{N}_g \) as follows:

\[
\begin{align*}
\mu^i_{t+1} &= (1 - \beta^{i,t}) \cdot \mu^i_t + \beta^{i,t} \cdot r^i_{t+1}, \\
\tilde{\omega}^i_t &= \omega^i_t + \beta^{i,t} \cdot \delta^i_t \cdot \nabla Q_t(\omega^i_t), \\
\omega^i_{t0k} &= \frac{1}{N-2f} \sum_{j \in \mathcal{U}^k_t} \tilde{\omega}^{jk}_t, \\
\omega^i_{tk+1} &= \omega^i_{t0k},
\end{align*}
\]

where \( \omega^i_{tk} \) is the \( k \)-th entry of agent \( i \) at time \( t \) and the local action-value TD-error \( \delta^i_t \) is defined in (5). Besides, the actor step is:

\[
\theta^i_{t+1} = \theta^i_t + \beta^{n,t} A^i_t \psi^i_t.
\]

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 \([\min_{i \in \mathcal{N}_g} \tilde{\omega}^{ik}_{t0}, \max_{i \in \mathcal{N}_g} \tilde{\omega}^{ik}_{t0}]\).

**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 \( i, j \), 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 \( B^k_t \), for all coordinates \( k \in \{1, 2, \ldots, K\} \), at each time \( t \) such that \( [\omega^{ik}_{t0}, \ldots, \omega^{ik}_{tNK}]^\top = B^k_t [\omega^{1k}_t, \ldots, \omega^{NK}_t]^\top \).

**Proof:** For any entry \( k \), from Lemma 1, there exist two normal agents \( i, j \in \mathcal{N}_g \), so that \( \omega^{ik}_{t0} = \alpha_1 \tilde{\omega}^{ik}_t + \alpha_2 \tilde{\omega}^{jk}_t \), where \( \alpha_1 \geq 0, \alpha_2 \geq 0 \) and \( \alpha_1 + \alpha_2 = 1 \). With Lemma 2, the weight matrix for entry \( k \) at time \( t \) can be a scrambling matrix, with \( i \)-th column being \( a_1 \textbf{1}_K \), \( j \)-th column being \( a_2 \textbf{1}_K \) and the remaining elements being 0.

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

\[
\begin{align*}
\mu^i_{t+1} &= (1 - \beta^{i,t}) \cdot \mu^i_t + \beta^{i,t} \cdot r^i_{t+1}, \\
\tilde{\omega}^i_t &= \omega^i_t + \beta^{i,t} \cdot \delta^i_t \cdot \nabla Q_t(\omega^i_t), \\
\omega^i_{tk+1} &= \sum_{j \in \mathcal{N}_g} b^j_t(i, j) \omega^{jk}_t,
\end{align*}
\]

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

\[
\omega^i_{t+1} = B_t \tilde{\omega}^i_t + \theta^i_t \psi^i_t.
\]

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

**Theorem 2:** Suppose that Assumptions 1-4 hold. Then, for any given policy \( \pi \) with the sequence \( \{\mu^i_t\} \) generated from (9), \( \lim_{T \to \infty} E[\|r^i_t - 1_{|\mathcal{S}|} \cdot A^i_t \|_1] \to 0 \) and \( \lim_{T \to \infty} \omega^i_t = \omega^i_0 \) almost surely for any \( i \in \mathcal{N}_g \), where \( \omega^i_0 \) is the unique solution to

\[
\sum_{i \in \mathcal{N}_g} B^i \Phi^\top T_{\gamma} (P - 1_{|\mathcal{S}|} \cdot |A| \mu^i) + \Phi^\top T_{\gamma} (P - 1_{|\mathcal{S}|} \cdot |A|) \Phi = 0,
\]

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

\[
\hat{\theta}^i = \hat{\theta}^i \left[ E_{s_t \sim \pi_0} \psi^i_t (A^i_t \theta \psi^i_t) \right], \quad i \in \mathcal{N}_g.
\]

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:

$$
\begin{align*}
\mu_{i+1}^t &= (1 - \beta_{w,t}) \cdot \mu_i^t + \beta_{w,t} \cdot r_{i+1}^t, \\
\omega_{i}^t &= \omega_{i}^t + \beta_{w,t} \cdot \delta_i^t \cdot \nabla Q_i(\omega_{i}^t), \\
\omega_{i}^{0k} &= \frac{1}{N-2T} \sum_{j \neq i} \omega_{j}^t, \\
\omega_{i}^{j1} &= \left\{ \begin{array}{ll}
\omega_{i}^{0k} & \text{if } l = k, \\
\omega_{i}^{jt} & \text{if } l \neq k,
\end{array} \right.
\end{align*}
$$

(11)

where $\delta_i^t$ 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 $B_t = \sum_{k=1}^{K} B_k \otimes (e_k e_k^T)$, where $\{B_k\}, \forall k = 1, \ldots, K$ are scrambling matrices. Then we have the update for all normal agents as follows:

$$
\omega_{i+1,\theta} = B_t \omega_{i,\theta}.
$$

From Lemma 4, there exists a matrix $B_t = \lim_{T \to \infty} \Pi I_t B_t$ with the form $B = 1_{[N_i]} \otimes [B^1, \ldots, B^{N_i}]$. As for the actor step,

$$
\theta_{i+1}^t = \theta_i^t + \beta_{\theta,t} A_i^t \psi_i^t.
$$

(12)

Theorem 3: Suppose that Assumptions 1-4 hold. Then, for any given policy $\pi_0$, with the sequence $\{\mu_i^t\}$ generated from (11), we have $\lim_t \mu_i^t = \mu^t = E_{s,a}[R(s, a)]$ and $\lim_t \omega_i^t = \omega_0$ almost surely for any $i \in N_g$, where $\omega_0$ is the unique solution to

$$
\sum_{i \in N_g} \hat{B} i \Phi^T D_y^{s,a}(R_i^t - 1_{[S,|A|]} \mu^t) + \Phi^T D_y^{s,a}(P^\theta - I_{[S,|A|]})\Phi \omega = 0.
$$

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

$$
\dot{\theta}_i = \hat{\Gamma}_i [E_{s_i \sim d_\theta a_i \sim s_0}[h_i(\omega_i^t, \mu_i^t, s_t, a_t)] = \Phi^T D_y^{s,a}[R_i^t - 1_{[S,|A|]} \mu_i^t + (P^\theta \Phi - \Phi) \omega_i^t].
$$

From Assumptions 2 and 4, and Lemma 5, we know that $\exists K_1, K_2 > 0$, s.t. $\|\phi_i^t\|_\infty \leq K_1$ and $\|r_{i+1}^t - \mu_i^t\| \leq K_2, \forall k, i$. Thus, $\exists K_3 > 0$ such that $\|h_i(\omega_i^t, \mu_i^t, s_t, a_t)\| \leq K_3 \cdot (1 + \|\omega_{i,t}\|_2^2)$. Moreover, we know $h_i(\omega_i^t, \mu_i^t, s_t, a_t)$ is Lipschitz continuous in $\omega_i^t$ and $M_i^{t+1}$ is martingale difference sequence. Since each $B_k^2$ is column stochastic, it has bounded norm. Thus, by Theorem A.2 in [8], it follows that for $i \in N_g$, $\omega_i^t$ is bounded almost surely.

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

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

where $\mu = [\mu_1, \ldots, \mu_{|N_g|}]^T$, and $R(s, a) = [R^1(s, a), \ldots, R^{|N_g|}(s, a)]^T$. Then, the equivalent point in the long run for $\mu$ is $\mu = E_{s,a}[R(s, a)]$.

**Proof:** The update for $\mu_i^t$ is

$$
\mu_{i+1}^t = \mu_i^t + \beta_{\theta,t} E[r_{i+1}^t - \mu_i^t | F_t] + \beta_{\theta,t} \xi_i^{t+1},
$$

where $\xi_i^{t+1} = r_{i+1}^t - E(r_{i+1}^t | F_t)$. Note that $E[r_{i+1}^t - \mu_i^t | F_t]$ is Lipschitz continuous in $\mu_i^t$, and $\xi_i^{t+1}$ is a martingale difference sequence. Based on Lemma 5, from Theorem B.2 in [8], $\mu_i^t$ will converge to a point $\mu^t$ almost surely in the long run, and the point satisfies the ODE: $\dot{\mu} = -\mu + E_{s,a}[R(s, a)]$ for all normal agents $i \in N_g$.

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 $\omega_i^t$ converges to $\omega_0$ almost surely for all normal agents $i \in N_g$, where $\omega_0$ is the unique equilibrium of the ODE

$$
\dot{\omega} = \Phi^T D_y^{s,a}(P^\theta - I_{[S,|A|]})\Phi \omega + \sum_{i=1}^{N_g} \hat{B} i \Phi^T D_y^{s,a}(R_i^t - 1_{[S,|A|]} \mu^t).
$$

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

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

(13)

$$
\dot{\omega} = \Phi^T D_y^{s,a}(P^\theta - I_{[S,|A|]})\Phi \omega + \sum_{i=1}^{N_g} \hat{B} i \Phi^T D_y^{s,a}(R_i^t - 1_{[S,|A|]} \mu^t).
$$

Note that from the Perron-Frobenius theorem and Assumption 1, the stochastic matrix $P^\theta$ has a simple eigenvalue of 1, and the remaining eigenvalues have real parts less than 1. Hence, since from Assumption 4 $\Phi$ is full column rank, $\Phi^T D_y^{s,a}(P^\theta - I\Phi)$ has all eigenvalues with negative real parts but one zero. Moreover, the eigenvalue of zero has eigen-vector $v$ when it satisfies $\Phi v = \alpha \Phi$ 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

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

Note that the solution for $\omega$ has the form $\omega = \omega_0 + lv$ with any $l \in \mathbb{R}$ and $v \in \mathbb{R}^v$ such that $\Phi v = 1_{|N_g|}$, where $\omega_0$ follows that $\Phi^\top D^{s,a}_\theta (P^\theta - I_{|S||A|})\Phi \omega_0 + \sum_{i=1}^{|N_g|} \tilde{B}_i \Phi^\top D^{s,a}_\theta (R^i - 1_{|S||A|}[\mu^i]) = 0$. By Assumption 4, $\omega_0$ 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


