Taming Byzantine Adversaries in Decentralized Multi-Agent Reinforcement Learning

Tamoghno Kandar[1] and Mayank Baranwal[1,2]

***Abstract*— In cooperative decentralized multi-agent reinforcement learning (MARL), the presence of even a single greedy adversarial agent can significantly disrupt the convergence towards optimality. This paper provides both theoretical insights and empirical evidence illustrating this phenomenon. Leveraging a variant of the `ClippedGossip` algorithm for consensus, we propose a novel approach to neutralize the disruptive influence of greedy adversaries. Through rigorous analysis, we establish the convergence of off-policy actor-critic decentralized MARL in environments containing non-cooperating agents. Experiments across diverse scenarios validate the efficacy of our approach, demonstrating its ability to maintain cooperation and achieve convergence even in the presence of adversarial behavior.**

I. Introduction

Multi-agent reinforcement learning (MARL) has been applied recently in many applications such as autonomous driving cars, intelligent systems [1], cyber-physical systems and sensor networks [2]. Specifically, MARL addresses the sequential decision-making problem of multiple autonomous agents that operate in a common environment, each of which aims to optimize its own long-term return by interacting with the environment and other agents. In this paper, we focus on MARL for cooperative agents.

In cooperative MARL [3], agents typically operate as independent decision-makers, refining their control policies through environmental interactions. At each state, each agent takes an action, and these actions together determine the next state of the environment and the reward of each agent. These cooperative agents aim at maximizing the long-term return corresponding to the team averaged reward. Although effective, these approaches hinge on the assumption of agents sharing their local rewards, which might not align with privacy concerns in certain scenarios. Recent advancements have sidestepped this assumption by pioneering entirely decentralized learning frameworks [4]. Here, both training and testing phases are decentralized, with agents exclusively receiving local rewards and exchanging information about team performance—such as local rewards or parameters of estimated team-average action-value functions—with neighboring agents, facilitated by a graph structure.

In this paper, we focus on decentralized learning using a consensus-based off policy actor-critic (AC) MARL algorithm [5]. Specifically we study the effect of a greedy Byzantine adversary on this consensus MARL algorithm. The Byzantine adversarial agent does not participate in the consensus process and instead aims to maximize its own reward. Even though other agents have their own goals, the adversarial agent's greedy actions can potentially push them all towards achieving the adversarial agent's own rewards instead. Our goal in this paper is to provide a robust algorithm and achieve resilience against adversarial attacks. We particularly focus on Byzantine adversarial agents and use a variant of the `ClippedGossip` algorithm [6] as the defensive strategy against these adversarial attacks. The main contributions of our paper are summarized as follows:

Cooperative decentralized MARL under adversarial setting Arguably for the first time, we undertake an examination of the off-policy decentralized AC-MARL algorithm in the context of an adversarial agent. Our investigation yields theoretical substantiation of the adversarial agent's influence, compelling cooperative agents to prioritize the adversarial agent's individual reward. We show that the presence of a single Byzantine adversary is sufficient to incite this opportunistic behavior.

Byzantine adversaries can be tamed Employing a variant of the recently introduced `ClippedGossip` consensus algorithm, we show that the opportunistic behavior of adversarial agents can be circumvented by the remaining cooperative agents. The central concept is to limit the consensus updates received from neighboring agents whose parameters significantly deviate from those of the receiving agent.

II. Related Work

Centralized Cooperative MARL: Recent advancements in centralized cooperative MARL algorithms have adopted the centralized training decentralized execution (CTDE) mechanism. These algorithms can be categorized into two main groups: value-based methods and actor-critic-based methods. Value-based approaches, such as those proposed by [7], achieve a separable joint Q-value function during centralized training. However, during execution, only the individual Q-value function for each agent is utilized. On the other hand, actor-critic methods typically feature a centralized critic, while the actor employs only local observations for decentralized execution by each agent [8].

Decentralized Cooperative MARL: Decentralized MARL has garnered significant attention in recent years due to privacy related concerns. Building upon the Gradient Temporal Difference (GTD-2) framework introduced by [9], fully decentralized extensions for MARL, applicable to both GTD-2 and linear Temporal Difference Control (TDC), have been proposed in [10].

[1]T. Kandar and M. Baranwal are with TCS Research, India.

[2]M. Baranwal is also a faculty with the Systems & Control Engineering group, Indian Institute of Technology, Bombay, India 400076. `baranwal.mayank@tcs.com`

Byzantine-Robust Distributed Learning: Byzantine-robust distributed learning has emerged as a significant area of interest in recent years. In decentralized settings, current Byzantine-resilient algorithms, such as those proposed in [11], are mainly applicable for supervised learning, either in i.i.d. or deterministic settings. In the consensus-based Multi-Agent Reinforcement Learning (MARL) setting, approaches like those presented in [12] utilize element-wise trimmed mean to aggregate neighboring messages.

III. Problem Formulation and Preliminaries

Our work focuses on fully decentralized multi-agent reinforcement learning (MARL), where both training and execution occur without centralized coordination. The agents, except for an adversary, are homogeneous and equipped with local objective functions. The primary goal is to collaboratively maximize the total reward.

Given the decentralized setting, each agent maintains its own local target policy function, parameterized by θ_i for the i^{th} agent. In the absence of centralized learning, agents can exchange their parameter vectors with neighboring agents while striving for cooperative reward maximization. The objective is to achieve consensus on a parameter vector that optimally enhances the total reward. However, the adversarial agent operates with a purely selfish objective, aiming solely to maximize its own reward while disregarding the cooperative goal, i.e., it does not participate in the consensus process.

The following sections outline the mathematical foundations underlying information exchange in agent networks and the conditions necessary for parameter convergence.

A. Networked Markov Decision Process

We consider a Markov decision process (MDP) model on a time-varying communication network with $\mathcal{N} = \{1, \dots, N\}$ representing a set of N agents. Let $\mathcal{N}^+$ and $\mathcal{N}^-$ denote the set of cooperative agents and adversarial agents, respectively, and note that $\mathcal{N} = \mathcal{N}^+ \cup \mathcal{N}^-$. The graph $\{G_t\}_{t\in\mathbb{N}} = \{(\mathcal{N}, \mathcal{E}_t)\}_{t\in\mathbb{N}}$ is defined by the set of vertices $\mathcal{N}$ and the set of edges $\mathcal{E}_t$ which depicts the neighbor relationships among the agents. Specifically, (j, i) is an edge in G_t whenever agents j and i can communicate. Furthermore, $(S, A, P, \{r^i\}_{i\in\mathcal{N}}, \{G_t\}_{t\in\mathbb{N}}, \gamma)$ denotes a networked multi-agent discounted MDP, where S is the shared state space, $A = \prod_{i\in\mathcal{N}} A^i$ is the joint action space (A^i is the action space of agent i), $P : S\times S\times A \to [0, 1]$ is the transition probability function, $r^i : S \times A \to [0, 1]$ is the local reward function for each agent $i \in \mathcal{N}$, the sequence $\{G_t\}_{t\in\mathbb{N}}$ describes the communication network at each timestep, and $\gamma \in (0, 1)$ is an appropriately chosen discount factor.

We assume that the state and action spaces are finite. Let $\bar{r}_{t+1}$ denote the global reward generated at time $t + 1$, and let $\bar{r} : S \times A \to \mathbb{R}$ be given by $\bar{r}(s, a) = \frac{1}{N}\sum_{i\in\mathcal{N}} r^i(s, a) = E[\bar{r}_{t+1} \mid s_t = s, a_t = a]$. The policy function $\pi : A \times S \to [0, 1]$ represents a conditional probability distribution $\pi(\cdot|s)$ over A for each element $s \in S$. For a policy π, the state-value function is

$$v_\pi(s) = E_{s\sim\pi}\Big[\sum\nolimits_{k=1}^{\infty} \gamma^{k-1}\bar{r}_{t+k} \mid s_t = s\Big], \qquad (1)$$

which satisfies

$$v_\pi(s) = \sum_{a\in A} \nu(a|s) \sum_{s'\in S} P(s'|s, a)[\bar{r}(s, a) + \gamma v_\pi(s')].$$

The action-value function is

$$q_\pi(s, a) = \sum\nolimits_{s'\in S} P(s'|s, a)(\bar{r}(s, a) + \gamma v_\pi(s')).$$

Suppose each agent $i \in \mathcal{N}$ be equipped with its own local behavior policy $\mu^i : A^i \times S \to [0, 1]$. Now for each agent $i \in \mathcal{N}$, let $\pi^i_{\theta^i} : A^i \times S \to [0, 1]$ be some suitable set of local target policy functions parametrized by $\theta^i \in \Theta^i$, where $\Theta^i \subset \mathbb{R}^{m_i}$ is compact. We further assume that each $\pi^i_{\theta^i}$ is continuously differentiable with respect to θ^i. Set $\theta = [\theta_1^\intercal, \dots, \theta_N^\intercal]^\intercal$. Define

$$\mu = \prod_{i=1}^{N} \mu^i : A \times S \to [0, 1] \text{ and } \pi_\theta = \prod_{i=1}^{N} \pi^i_{\theta^i} : A \times S \to [0, 1].$$

These correspond to the global behavior function and global parametrized target policy function, respectively. Building upon previous works [13], our goal is to maximize the global policy function given by:

$$J_\mu(\theta) = \sum\nolimits_{s\in S} d_\mu(s) v_{\pi_\theta}(s). \qquad (2)$$

Here, $d_\mu(s) := \lim_{t\to\infty} \mathrm{P}(S_t = s|s_0, \mu)$ is the limiting distribution of states under μ and $\mathrm{P}(S_t = s|s_0, \mu)$ is the probability that $S_t = s$ when starting in state s_0 and executing μ. The gradient of $J_\mu(\theta)$ defined in (2) with respect to each θ^i as proved in [5] is given by:

$$\nabla_{\theta^i} J_\mu(\theta) = \sum_{s\in S} m(s) \sum_{a\in A} \pi_\theta(a|s) q_\theta(s, a) \nabla_{\theta^i} \log \pi_{\theta^i}(a^i|s).$$

where $m(s)$ is the emphatic weighting of $s \in S$, with vector form $\mathbf{m}^\intercal = \mathbf{d}_\mu^\intercal(\mathbf{I} - \mathbf{P}_{\theta,\gamma})^{-1}$; $\mathbf{P}_{\theta,\gamma} \in \mathbb{R}^{|S|\times|S|}$ has entries:

$$\mathbf{P}_{\theta,\gamma}(s, s') = \gamma \sum\nolimits_{a\in A} \pi_\theta(a|s) P(s'|s, a).$$

B. Assumptions and Network Structure

Before proceeding further, we introduce some common assumptions needed for obtaining the convergence of the consensus MARL algorithm which we introduce in the later section. Most of these assumptions are standard and have appeared in existing literature, such as [5].

Assumption 1: The policy $\pi^i(a^i|s; \theta^i) > 0$ for any $i \in \mathcal{N}$, $\theta^i \in \Theta^i$, $s \in \mathcal{S}$, $a^i \in \mathcal{A}^i$. Also, $\pi^i(a^i|s; \theta^i)$ is continuously differentiable with respect to θ^i. For any $\theta \in \Theta$, we let $P_\theta(s_{t+1}|s_t) = \sum_{a_t\in\mathcal{A}} P(s_{t+1}|s_t, a_t)\pi(a|s; \theta)$ denote the transition matrix of the Markov chain $\{s_t\}_{t\geq 0}$ induced by policy $\pi(a|s; \theta)$. The Markov chain $\{s_t\}_{t\geq 0}$ is irreducible and aperiodic under any $\pi(a|s; \theta)$.

Assumption 2: For each agent $i \in \mathcal{N}$, the local θ-update is carried out using the projection operator $\Gamma^i : \mathbb{R}^{m_i} \to \Theta^i \subset \mathbb{R}^{m_i}$. Furthermore, the set $\Theta = \prod_{i=1}^N \Theta^i$ contains at least one local optimum of $J_\mu(\theta)$.

Assumption 3: The instantaneous reward $r^i_{t+1}(s_t, a_t, s_{t+1})$ is uniformly bounded for any $i \in \mathcal{N}, t \geq 0$, i.e., $r^i_t \leq R$.

Recall that Assumptions 1-3 are trivially satisfied in most practical settings. Additionally, we require the underlying

network to be undirected and connected. More specifically, we impose the following structure on underlying network:

Assumption 4: The sequence of random matrices $\{C_t\}_{t\geq 0} \subseteq \mathbb{R}^{N\times N}$ satisfies

1) C_t is row stochastic, i.e., $C_t\mathbf{1} = \mathbf{1}$, and $c_t(i,j) = 1$ for $i = j \in \mathcal{N}^-$. There exists a constant $\eta \in (0,1)$ such that, for any $c_t(i,j) > 0$, we have $c_t(i,j) \geq \eta$.
2) If $(i,j) \notin \mathcal{E}_t$, then $c_t(i,j) = 0$. Moreover, for every $(i,j) \in \mathcal{N}$, we have $c_t(i,j) = c_t(j,i)$.
3) The spectral norm $\rho = \mathbb{E}[C_t^\intercal(I - \mathbf{1}\mathbf{1}^\intercal/N)C_t]$ satisfies $0 \leq \rho < 1$.
4) Given the σ-algebra generated by the random variables before time t, C_t is conditionally independent of r_{t+1}^i for each $i \in \mathcal{N}$.

Remark 1: Assumption 4 states that the edge weights $c_t(i,j)$ are non-negative whenever there exists an edge between any pair (i,j) of agents. A simple yet decentralized method for selecting edge weights is using Metropolis weights, defined as:

$$c_t(i,j) = \big(1 + \max[d_t(i), d_t(j)]\big)^{-1}, \quad \forall (i,j) \in \mathcal{E}_t,$$
$$c_t(i,i) = 1 - \sum\nolimits_{j\in\mathcal{N}_t(i)} c_t(i,j), \quad \forall i \in \mathcal{N}, \tag{3}$$

where $d_t(i)$ is the degree of the i^{th}-agent. Observe that the Metropolis consensus matrix is row-stochastic, with 1 as its dominant and simple eigenvalue. Consequently, the choice of Metropolis weights inherently satisfies Assumption 4.

Assumption 5: The feature matrix Φ has linearly independent columns, and the value function approximator $v_\omega(s) = \phi(s)^\intercal\omega$ is linear in ω.

Assumption 6: The step sizes $\beta_{\omega,t}$ and $\beta_{\theta,t}$ satisfy $\sum_t \beta_{\omega,t} = \sum_t \beta_{\theta,t} = \infty$, $\sum_t \beta_{\omega,t}^2 + \beta_{\theta,t}^2 < \infty$, $\beta_{\theta,t} = o(\beta_{\omega,t})$, and $\lim_{t\to\infty} \frac{\beta_{\omega,t+1}}{\beta_{\omega,t}} = 1$.

Remark 2: Assumption 6 is the standard Robbins-Monro type step size condition used in stochastic approximation algorithm, and is a crucial condition for ensuring the convergence of the algorithm.

Our primary contribution is the theoretical foundation for the observation that a single adversarial agent can exploit the network, inducing opportunistic behavior in other agents and driving them to maximize its local reward. Consequently, we make the following assumption for the sake of simplicity of analysis. However, our results readily generalize to multiple adversarial agents, provided no adversary is entirely surrounded by other adversaries, which could potentially obstruct information flow within the network.

Assumption 7: In adversarial scenario, we assume that only one malicious Byzantine agent is present.

C. Useful Definitions & Results for Byzantine agents

Definition 1: `ClippedGossip` is a gossip-based aggregator which uses its local reference model as center and clips all the received neighboring model weights. Formally, for $\textsc{Clip}(z,\tau) := \min(1, \tau/||z||)\cdot z$, we update information as:

$$\omega_i^{t+1} := \sum\nolimits_{j=1}^N W_{ij}(\omega_i^t + \textsc{Clip}(\omega_j^t - \omega_i^t, \tau_i)), \quad t = 0,1,\ldots \tag{ClippedGossip}$$

Algorithm 1 Byzantine-Robust Decentralized Optimization

Input: $\omega^0 \in \mathbb{R}^d$, α, η, $\{\tau_i^t\}$, $m_i^0 = g_i(\omega^0)$
for $t = 0,1,\ldots$ **do**
  for $i = 1,\ldots,N$ **in parallel**
    $m_i^{t+1} = (1-\alpha)m_i^t + \alpha g_i(\omega_i^t)$
    $\omega_i^{t+\frac{1}{2}} = \omega_i^t - \eta m_i^{t+1}$ if $i \in \mathcal{N}^+$ else $*$
    Exchange $\omega_i^{t+\frac{1}{2}}$ with $\mathcal{N}_i$
    $\omega_i^{t+1} = \texttt{ClippedGossip}(\omega_1^{t+\frac{1}{2}},..,\omega_n^{t+\frac{1}{2}};\tau_i^{t+1})$
  end for
end for

Definition 2: `Gossip averaging` is a simple consensus algorithm where we take the weighted average of the local node's weights and neighboring node's weights as:

$$\omega_i^{t+1} := \sum\nolimits_{j=1}^N W_{ij}\omega_j^t, \qquad t = 0,1,\ldots \tag{GossipAveraging}$$

We will be using Theorem 3 from [6] for our proofs. The assumptions stated in the following section is taken from [6]. Recall that $\mathcal{N}^+$ denotes the set of cooperative agents. Let $\mathcal{N}_i \in \mathcal{N}$ be the neighbors of node i and let $\bar{N}_i := \mathcal{N}_i \cup \{i\}$. Consider the general distributed optimization problem

$$\min_{\omega\in\mathbb{R}^d} f(\omega) := \frac{1}{|\mathcal{N}^+|}\sum\nolimits_{i\in\mathcal{N}^+}\{f_i(\omega) := \mathbb{E}_{\xi\sim\mathcal{D}_i}F_i(\omega;\xi)\}$$

on heterogeneous (non-i.i.d.) data, where f_i is the local objective of agent i with data distribution $\mathcal{D}_i$ and independent noise ξ_i. We require that the gradients computed over these data distributions satisfy standard Assumptions 8 and 9 in the stochastic optimization literature.

Assumption 8: **Bounded noise and heterogeneity.** Assume that for all $i \in \mathcal{N}^+$ and $\omega \in \mathbb{R}^d$, we have

$$\mathbb{E}_{\xi\sim\mathcal{D}_i}||\nabla F_i(\omega;\xi) - \nabla f_i(\omega)||^2 \leq \sigma^2$$
$$\mathbb{E}_{j\sim\mathcal{N}^+}||\nabla f_j(\omega) - \nabla f(\omega)||^2 \leq \zeta^2$$

Assumption 9: **L-smoothness.** For $i \in \mathcal{N}^+$, $f_i(x) : \mathbb{R}^d \to \mathbb{R}$ is differentiable and there exists a constant $L \geq 0$ such that for each $x, y \in \mathbb{R}^d$, $\|\nabla f_i(x) - \nabla f_i(y)\| \leq L\|x - y\|$.

Theorem 1 ([5]): Let the total weight of adjacent Byzantine edges around a regular agent i be defined as $\delta_i := \sum_{j\in\mathcal{N}^-} W_{ij}$, and let the maximum Byzantine weight be given by $\delta_{\max} := \max_{i\in\mathcal{N}^+}\delta_i$. Assume that the underlying graph is undirected and connected, and that Assumptions 8-9 hold. Furthermore, let $\delta_{\max} = \mathcal{O}(\gamma^2)$. Setting $\alpha := 3\eta L$, Algorithm 1 ensures that $\frac{1}{T+1}\sum_{t=0}^T \|\nabla f(\bar{\omega}_t)\|_2^2 \leq \frac{\delta_{\max}\zeta^2}{\gamma^2} + \epsilon$ with an iteration complexity bounded by $\mathcal{O}\left(\frac{\sigma^2}{N\epsilon^2}\left(\frac{1}{N} + \delta_{\max}\right) + \frac{\zeta}{\gamma\epsilon^{3/2}} + \frac{\sigma^{2/3}}{\gamma^{2/3}\epsilon^{4/3}} + \frac{1}{\gamma\epsilon}\right)$.

IV. Main Results: Byzantine Robust Multi Agent Off Policy Actor Critic Algorithm

A. Algorithm

We now introduce our off policy multi agent algorithm inspired from [5]. The proof of convergence of this particular

algorithm in the **absence** of any adversarial agent is already done in [5]. The primary difference in our work is that we use `ClippedGossip`-styled update [6] instead of traditional `GossipAveraging` to make our algorithm robust to Byzantine adversarial attack. In the subsections below, we establish the convergence of our algorithm in the **presence** of an adversarial byzantine agent both in `ClippedGossip` and `GossipAveraging` cases.

Algorithm 2 Byzantine Robust Multi-agent Off-policy Actor-critic

Initialize $\theta_0^i = 0, \omega_0 = e_{-1} = 0, F_{-1} = 0, \rho_{-1} = 1, \tau = 0.5$, for all $i \in \mathcal{N}$, the initial state s_0, and the stepsizes $\{\beta_{\omega,t}\}_{t\in\mathbb{N}}, \{\beta_{\theta,t}\}_{t\in\mathbb{N}}$.

repeat
  for all $i \in \mathcal{N}$ **do**
    receive $\widetilde{\omega}_{t-1}^j$ from neighbors $j \in \mathcal{N}_t(i)$
    if Using `ClippedGossip` **then**
      $\omega_t^i = \sum_{j\in\mathcal{N}} c_{t-1}(i,j)\Big(\widetilde{\omega}_{t-1}^i + \min\left(1, \frac{\tau}{||\widetilde{\omega}_{t-1}^j - \widetilde{\omega}_{t-1}^i||}\right)\cdot(\widetilde{\omega}_{t-1}^j - \widetilde{\omega}_{t-1}^i)\Big)$
    else
      $\omega_t^i = \sum_{j\in\mathcal{N}} c_{t-1}(i,j)\widetilde{\omega}_{t-1}^j$
    end if
    execute $a_t^i \sim \mu_i(\cdot|s_t)$
    $\rho_t^i = \frac{\pi_{\theta_t^i}^i(a_t^i|s_t)}{\mu_i(a_t^i|s_t)}$
    $p_t^i = \log \rho_t^i$
    observe r_{t+1}^i, s_{t+1}
    repeat ▷ begin inner consensus loop
      Share p_t^i, receive p_t^j from neighbors $j \in \mathcal{N}_t(i)$
      $p_t^i \leftarrow \sum_{j\in\mathcal{N}} c_t(i,j)p_t^j$
    until consensus is achieved ▷ end inner consensus loop
    $\rho_t = \exp(np_t^i)$
    $F_t = 1 + \gamma\rho_{t-1}F_{t-1}$ ▷ begin critic update
    $M_t = \lambda + (1-\lambda)F_t$
    $e_t = \gamma\lambda e_{t-1} + M_t\nabla_\omega v_{\omega_t^i}(s_t)$
    $\delta_t^i = r_{t+1}^i + \gamma v_{\omega_t^i}(s_{t+1}) - v_{\omega_t^i}(s_t)$
    $\widetilde{\omega}_t^i = \omega_t^i + \beta_{\omega,t}\rho_t\delta_t^i e_t$ ▷ end critic update
    $M_t^\theta = 1 + \lambda^\theta\gamma\rho_{t-1}F_{t-1}$ ▷ begin actor update
    $\theta_{t+1}^i = \theta_t^i + \beta_{\theta,t}\rho_t M_t^\theta \nabla_{\theta^i}\log\pi_{\theta_t^i}^i(a_t^i|s_t)\delta_t^i$ ▷ end actor update
    broadcast $\widetilde{\omega}_t^i$ to neighbors over network
  end for
until convergence

B. Theoretical Analysis of Adversarial attack under `GossipAveraging` Update

We show that the parameters of the critic converge to a fixed point both for the Byzantine adversarial agent and for the other non-Byzantine agents. Note that the convergence of actor parameters is unaffected by the addition of an adversarial agent, the proof of which has already been shown in [5], so we will not address it here.

Theorem 2: Let Assumptions 1 and 3-7 hold, then for any policy $\pi(a|s;\theta)$ the weight vector ω_t^i in Algorithm 2 converges to a unique value ω^* almost surely. That is, we have $\lim_{t\to\infty}\omega_t^i = \omega^*$ for $i \in \mathcal{N}$. ω^* is given by $\omega^* = -C^{-1}b$, where $C = -\Phi^T\overline{M}(I - P_{\pi,\gamma}^\lambda)\Phi$, $b = \Phi^T\overline{M}r_{\pi,\gamma}^\lambda$.

Proof: We let $\omega_t = [(\omega_t^1)^\intercal, \ldots, (\omega_t^N)^\intercal]^\intercal \in \mathbb{R}^{(M+L)N}$. To prove Theorem 2, we need to show the following

1) The parameter ω_t remains bounded for all $t \geq 0$,
2) The adversary's parameters asymptotically converge, i.e., $\omega_t^j \to \omega^*$, $j \in \mathcal{N}^-$,
3) The agents' parameters asymptotically converge to the consensus value $\langle\omega_t\rangle$.

We take advantage of the convergence analysis done in [5] to prove the key Lemmas.

Lemma 1: Let Assumptions 1 and 3-6 hold. Then the sequence $\{\omega_t\}$ satisfies $\sup_t ||\omega_t|| < \infty$ almost surely.

Proof: The main proof is given in the Lemma A.3 of [5, Appendix]. The only difference in our work is that in the absence of the consensus step the updates of ω_t^i, $i \in \mathcal{N}$, asymptotically follow the ODE $\dot{\omega}_t^i = C\omega_t^i + b_t^i$. The discount factor satisfies $\gamma \in [0,1)$ and the stochastic matrix C is negative definite as proved in [14] having eigenvalues with strictly negative real parts, which implies that the ODE $\dot{\omega}_t^i = C\omega_t^i + b_t^i$ has an asymptotically stable equilibrium. Therefore, $\sup_t ||\omega_t|| < \infty$ almost surely. ■

Lemma 2: Let Assumptions 1, 3, and 5-7 hold. Then $\lim_{t\to\infty}\omega_t^j = \omega_\theta$, $j \in \mathcal{N}^-$, almost surely. Furthermore, ω_θ is a unique solution to $C\omega + b = 0$, where C and b are defined in Theorem 2.

Proof: Since the adversarial agent bypasses the consensus step (as previously mentioned), we can use Lemma 1 to conclude that $\dot{\omega}_t^j = C\omega_t^j + b_t^j$ is the limiting ODE. The ODE possesses a unique asymptotically stable equilibrium point ω_θ that satisfies $C\omega + b = 0$. ■

Lemma 3: Let Assumptions 1 and 3-7 hold. Then the disagreement vector $\omega_{\perp,t}$ satisfies $\lim_{t\to\infty}\omega_{\perp,t} = 0$ a.s.

Proof: The proof follows from Lemma A.1 in [5]. ■

In order to complete the proof of Theorem 2, we recall:

1) $\lim_{t\to\infty}(\omega_t^j - \omega_\theta) = 0$ for $j \in \mathcal{N}^-$ a.s. (Lemma 2)
2) $\lim_{t\to\infty}(\omega_t^i - \langle\omega_t\rangle) = 0$ for $i \in \mathcal{N}$ a.s. (Lemma 3).

Therefore, $\lim_{t\to\infty}(\langle\omega_t\rangle - \omega_\theta) = 0$ almost surely where ω_θ satisfies $C\omega + b = 0$. ■

C. Theoretical Analysis of Adversarial attack under `ClippedGossip` Update

In the analysis below, we establish that the consensus update steps of both Algorithm 1 and Algorithm 2 are fundamentally similar. Leveraging Theorem 1, which was used to prove the convergence of Algorithm 1, we extend the analysis to demonstrate the convergence of Algorithm 2.

Lemma 4: Consider the MARL setting described in Section 3.1. Under Assumption 3, δ_t^i is upper bounded.

Proof: Assuming infinite horizon and from Assumption 3:

$$v_\pi(s) = E_{s\sim\pi}\Big[\sum_{k=1}^{\infty}\gamma^{k-1}\bar{r}_{t+k} \mid s_t = s\Big] \leq R/1-\gamma$$

Here we used the definition of state value function given by (1). Hence now we can simplify for δ_t^i.

$$\delta_t^i = r_{t+1}^i + \gamma v_{\omega_t}^i(s_{t+1}) - v_{\omega_t}^i(s_t) \leq R + \frac{\gamma R}{1-\gamma} - \frac{R}{1-\gamma} = 0$$

■

Lemma 5: Consider the MARL setting described in Section 3.1. Under the assumption that $\rho_{t-1} \leq 1$ and ρ_t is constant, F_t is upper bounded.

Proof: Observe that

$$F_0 = 0, \quad F_t = 1 + \gamma\rho_{t-1}F_{t-1} = \sum_{i=0}^{t-1}(\gamma\rho_{t-1})^i = \frac{1-(\gamma\rho_{t-1})^t}{1-\gamma\rho_{t-1}}$$

Using the assumption that $\rho_{t-1} \leq 1$ we get $\gamma\rho_{t-1} < 1$. Therefore F_t is bounded. ■

We first simplify the update of critic parameters as follows:

$$\begin{aligned}
e_t &= \gamma\lambda e_{t-1} + M_t \nabla_\omega v_{\omega_t^i}(s_t) \\
M_t &= \lambda + (1-\lambda)F_t \\
\omega_t^i &= \omega_t^i + \beta_{\omega,t}\rho_t\delta_t^i e_t \\
\omega_t^i &= \omega_t^i + \beta_{\omega,t}\rho_t\delta_t^i(\gamma\lambda e_{t-1} + M_t\nabla_\omega v_{\omega_t^i}(s_t)) \\
\omega_t^i &= \omega_t^i + \beta_{\omega,t}\rho_t\delta_t^i(\gamma\lambda e_{t-1} + (\lambda + (1-\lambda)F_t)\nabla_\omega v_{\omega_t^i}(s_t)) \\
\omega_t^i &= \omega_t^i + \beta_{\omega,t}\rho_t\delta_t^i(\gamma\lambda e_{t-1} + \\
&\qquad (1 + (1-\lambda)\gamma\rho_{t-1}F_{t-1}))\nabla_\omega v_{\omega_t^i}(s_t))
\end{aligned}$$

Taking $\gamma\lambda = 1 - \alpha$, we get:

$$\begin{aligned}
\omega_t^i = \omega_t^i + \beta_{\omega,t}\rho_t\delta_t^i\big((1-\alpha)e_{t-1} + \\
(1 + (\gamma - 1 + \alpha)\rho_{t-1}F_{t-1})\nabla_\omega v_{\omega_t^i}(s_t)\big)
\end{aligned} \tag{4}$$

If we recall from Algorithm 1, the consensus update is:

$$\begin{aligned}
\omega_i^{t+\frac{1}{2}} &= \omega_i^t - \eta m_i^{t+1} \\
\omega_i^{t+\frac{1}{2}} &= \omega_i^t - \eta((1-\alpha)m_i^t + \alpha g_i(\omega_i^t))
\end{aligned} \tag{5}$$

We see that both the equations (4) and (5) are identical with:

$$\begin{aligned}
\eta &\coloneqq \beta_{\omega,t}\rho_t\delta_t^i \\
F_i(\omega_t) &\coloneqq v_{\omega_t^i}(s_t)(1 + (\gamma - 1 + \alpha)\rho_{t-1}F_{t-1})/\alpha \\
g_i(\omega_i^t) &\coloneqq \nabla F_i(\omega_i^t) \\
\min_{\omega_t \in \mathbb{R}^d} f(\omega_t) &\coloneqq \frac{1}{|\mathcal{N}^+|}\sum\nolimits_{i \in \mathcal{N}^+}\{f_i(\omega_t) \coloneqq \mathbb{E}_{\xi \sim \mathcal{D}_i}F_i(\omega_t;\xi)\}
\end{aligned}$$

Theorem 3: Consider the multi-agent reinforcement learning (MARL) setting described in Section III-A, and let $\delta_{\max} = \mathcal{O}(\gamma^2)$. Define L as in Theorem 1. Then, setting $\alpha := 3\beta_{\omega,t}L$, Algorithm 2 guarantees $\frac{1}{T+1}\sum_{t=0}^{T}\|\nabla f(\bar{\omega}_t)\|_2^2 \leq \frac{\delta_{\max}\zeta^2}{\gamma^2} + \epsilon$ with an iteration complexity of $\mathcal{O}\bigg(\frac{\sigma^2}{N\epsilon^2}\bigg(\frac{1}{N} + \delta_{\max}\bigg) + \frac{\zeta}{\gamma\epsilon^{3/2}} + \frac{\sigma^{2/3}}{\gamma^{2/3}\epsilon^{4/3}} + \frac{1}{\gamma\epsilon}\bigg)$.

Proof: We make the following observations:

- We first note that $\rho_{t-1} \leq 1$.
- We assume $\lambda = 0.9$ as suggested in [13].
- $\gamma < 1$ is the discount factor.

Based on Lemma 4 and the observation that ρ_{t-1} is bounded, we can conclude that η is bounded as well. In Algorithm 2, our communication network is assumed to be a connected, undirected graph satisfying Assumption 4. Additionally, we utilize Metropolis weights 3 in the consensus matrix which adheres to Assumption 4. Since $v_{\omega_t^i}(s_t)$ is linear by Assumptions 5, $F_i(\omega_t)$ is consequently differentiable. This ensures that Assumptions 8 and 9 are also satisfied. Therefore, with Assumptions 8-9 met, we can directly apply Theorem 1, which guarantees convergence. ■

V. A Multi-Agent Gridworld Experiment

We now validate our theoretical findings through a simple multi-agent grid-world experiment.

A. Description of Environment

GridWorld [15]: This environment is a grid-world of dimension 6×6. Let the number of agents be $N = 4$. The position of agent i is described by the tuple $(x_i, y_i), i \in [0 \cdots 5]^2$. The state of this grid-world is given as $s = [(x^i,\ y^i),\ i \in \mathcal{N}] \in \mathcal{S}$ where $\mathcal{S} = \mathcal{S}^1 \times \ldots \times \mathcal{S}^N$. The cardinality of S is $|\mathcal{S}| = 36^N$. The agents can move up, down, left, right, or stay put. If a move takes them off the grid, they just stay where they are. The set of actions of the network is given as $\mathcal{A} = \mathcal{A}^1 \times \ldots \times \mathcal{A}^N$, whose cardinality is $|\mathcal{A}| = 5^N$. The reward of agent $i \in \mathcal{N}$ is given as:

$$r^i(s^i) = -|x^i - x^i_{\text{des}}| - |y^i - y^i_{\text{des}}| - q^i,$$

where q^i denotes the number of neighboring agents that agent i collides at the current time step. $(x^i_{\text{des}}, y^i_{\text{des}})$ and (x^i, y^i) denote the desired and the initial position of agent i.

B. Implementation Details

Choice of consensus matrix: For undirected graphs, a useful choice of the weights $c_t(i,j)$ that rely on only local information of the agents is the Metropolis weights.
We consider the following experimental settings:

- Cooperative case where all agents participate in consensus process truthfully.
- Adversarial setting: We introduce a single Byzantine adversarial agent (Agent-1) that does not participate in consensus while other agents follow gossip averaging.
- Adversarial setting with `ClippedGossip`-styled update: We introduce a single Byzantine adversarial agent (Agent-1) that does not participate in consensus while other agents follow resilient `ClippedGossip`-styled update (see Algorithm 2).

C. Results and Inferences

We train 4 agents for 1900 episodes in all the three scenarios. Within each episode, agents are initialized at random locations in the environment and continue for a maximum of 100 steps, or until they reach their desired positions. For each agent, we use the same multi-layer perceptron with two hidden layers to approximate both the actor and critic functions. We plot the true cumulative rewards obtained in each episode for each agent and the team averaged returns versus the number of episodes in Figure 1.

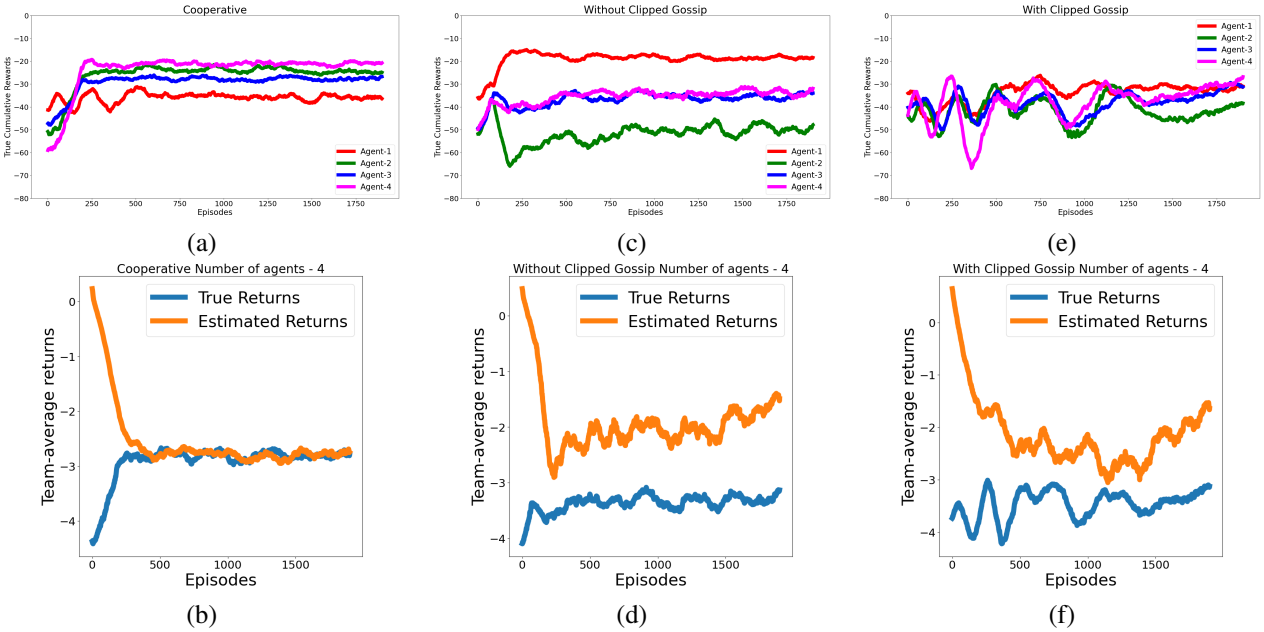

Fig. 1: GridWorld environment: (a)-(b) Multi-agent cooperative setting, (c)-(d) Agent-1 is Byzantine, while gossip clipping is turned off, (e)-(f) Agent-1 is Byzantine with robust averaging

As expected, all agents achieve near-optimal performance in environments in the cooperative case. However, the presence of a single Byzantine adversarial agent in the case where agents follow an ordinary consensus update, disrupts the learning process. This is evident from Figure 1c, where the adversarial agent maximizes its rewards while the performance of other agents deteriorates compared to the adversary-free scenario. When the agents adopt a more resilient `ClippedGossip`-styled update, the non-adversarial agents can be observed to successfully negate the presence of Byzantine adversary (see Figure 1e).

VI. Conclusion

In this study, we first demonstrate the vulnerability of the algorithm proposed in [5] to adversarial attacks. Subsequently, we introduce the `ClippedGossip` update mechanism, which exhibits proven resilience against Byzantine adversaries. Our analysis includes a convergence assessment of our algorithmic variant amidst adversarial influence, employing linear function approximation. Through experiments utilizing non-linear function approximation, we remark that the presence of a single adversarial agent prioritizes its own rewards while compromising rewards for others, which can be circumvented by employing a `ClippedGossip` update.

References

[1] Weijia Zhang, Hao Liu, Fan Wang, Tong Xu, Haoran Xin, Dejing Dou, and Hui Xiong. Intelligent electric vehicle charging recommendation based on multi-agent reinforcement learning. In *Proceedings of the Web Conference 2021*, WWW '21. ACM, April 2021.

[2] Chongjie Zhang and Victor Lesser. Coordinated multi-agent reinforcement learning in networked distributed pomdps. *Proceedings of the AAAI Conference on Artificial Intelligence*, 25(1):764–770, Aug. 2011.

[3] Afshin Oroojlooy and Davood Hajinezhad. A review of cooperative multi-agent deep reinforcement learning. *Applied Intelligence*, 53(11):13677–13722, 2023.

[4] Kaiqing Zhang, Zhuoran Yang, Han Liu, Tong Zhang, and Tamer Basar. Fully decentralized multi-agent reinforcement learning with networked agents. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 5872–5881. PMLR, 2018.

[5] 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.

[6] Lie He, Sai Praneeth Karimireddy, and Martin Jaggi. Byzantine-robust decentralized learning via clippedgossip. *arXiv preprint arXiv:2202.01545*, 2022.

[7] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder De Witt, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. Monotonic value function factorisation for deep multi-agent reinforcement learning. *Journal of Machine Learning Research*, 21(178):1–51, 2020.

[8] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, page 6382–6393. Curran Associates Inc., 2017.

[9] Richard S Sutton, Hamid Reza Maei, Doina Precup, Shalabh Bhatnagar, David Silver, Csaba Szepesvári, and Eric Wiewiora. Fast gradient-descent methods for temporal-difference learning with linear function approximation. In *Proceedings of the 26th annual international conference on machine learning*, pages 993–1000, 2009.

[10] Miloš S. Stanković, Marko Beko, and Srdjan S. Stanković. Distributed consensus-based multi-agent temporal-difference learning. *Automatica*, 151:110922, 2023.

[11] Jie Peng, Weiyu Li, and Qing Ling. Byzantine-robust decentralized stochastic optimization over static and time-varying networks. *Signal Processing*, 183:108020, 2021.

[12] Yijing Xie, Shaoshuai Mou, and Shreyas Sundaram. Towards resilience for multi-agent *qd*-learning. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 1250–1255. IEEE, 2021.

[13] Ehsan Imani, Eric Graves, and Martha White. An off-policy policy gradient theorem using emphatic weightings. *Advances in neural information processing systems*, 31, 2018.

[14] Huizhen Yu. On convergence of emphatic temporal-difference learning. In *Conference on learning theory*, pages 1724–1751. PMLR, 2015.

[15] Martin Figura, Krishna Chaitanya Kosaraju, and Vijay Gupta. Adversarial attacks in consensus-based multi-agent reinforcement learning. In *2021 American control conference (ACC)*, pages 3050–3055. IEEE, 2021.