Journal of Robotics and Control (JRC)
Volume 3, Issue 2, March 2022
ISSN: 2715-5072, DOI: 10.18196/jrc.v3i2.13082

Consensus of Multi-agent Reinforcement Learning Systems: The Effect of Immediate Rewards

Neshat Elhami Fard[1] and Rastko R. Selmic[2]
[1,2]Department of Electrical and Computer Engineering, Concordia University, Montreal, Canada
Email: [1]neshat.elhamifard@concordia.ca, [2]rastko.selmic@concordia.ca

***Abstract*—This paper studies the consensus problem of a leaderless, homogeneous, multi-agent reinforcement learning (MARL) system using actor-critic algorithms with and without malicious agents. The goal of each agent is to reach the consensus position with the maximum cumulative reward. Although the reward function converges in both scenarios, in the absence of the malicious agent, the cumulative reward is higher than with the malicious agent present. We consider here various immediate reward functions. First, we study the immediate reward function based on Manhattan distance. In addition to proposing three different immediate reward functions based on Euclidean, n-norm, and Chebyshev distances, we have rigorously shown which method has a better performance based on a cumulative reward for each agent and the entire team of agents. Finally, we present a combination of various immediate reward functions that yields a higher cumulative reward for each agent and the team of agents. By increasing the agents' cumulative reward using the combined immediate reward function, we have demonstrated that the cumulative team reward in the presence of a malicious agent is comparable with the cumulative team reward in the absence of the malicious agent. The claims have been proven theoretically, and the simulation confirms theoretical findings.**

***Keywords*—Multi-agent system; Malicious agent; Consensus control; Reinforcement learning; Immediate reward; Cumulative reward.**

I. Introduction

Applications of reinforcement learning (RL) algorithms have increased over the years and led to tremendous advancements in various fields of science and robotics [1], [2]. The RL algorithms have been used to solve numerous sequential decision-making problems and have encountered significant hurdles when dealing with high-dimensional environments [3]. To partially overcome high-dimensional problems and perform tasks that require policy control, deep reinforcement learning (DRL) algorithms were generated by combining deep learning, and RL algorithms [4], [5], [6]. In the combined algorithm, deep learning enables RL to address those challenges [7].

One of the DRL applications is in the multi-agent systems (MAS) control [8], [9]. The use of MAS stems from nature, where multiple agents have higher efficiency and competitiveness when acting together in groups. Individual agents collaborate and interact with the environment to achieve the best results [3]. The consensus control is one of the fundamental problems in MAS, where agents support a common decision or aim in the best interest of the entire system. The agents participate in a group decision-making process, called consensus decision-making [10], [11], [12]. In consensus control, the goal is to reach a global agreement on a value or state for all agents [12], [13].

To reach consensus, we use the RL actor-critic method for N homogeneous agents. An actor, as the policy structure, decides to choose the best action based on its perception of the environment [14]. A critic, as the value function structure, indicates what is right in the long term and evaluates the selected actions by the actor [14]. In this paper, the internal structures of agents' actors and critics are multi-layer neural networks (NN).

A. Contributions

We studied the behavior of a leaderless, homogeneous multi-agent reinforcement learning (MARL) system in reaching consensus using a decentralized actor-critic method with and without malicious agents. We defined and proposed various immediate reward functions based on different distance metrics. These immediate reward functions can be used to calculate the cumulative reward for each agent and the MARL system. This work examines whether changing the immediate reward can improve the system's overall performance even with the destructive effects of a malicious agent. Suppose one of the distance metrics (e.g., Manhattan, Euclidean, $n-$norm, or Chebyshev distances) provides a smaller value between the current position and the desired position compared with the existing distance metrics. Consequently, the extracted immediate reward from the discussed distance metric generates a higher return cumulative reward for each agent and the MARL system as a whole. Hence, the criterion for measuring the MARL system's behavior is based on various immediate reward functions. First, we studied the immediate reward function based on Manhattan distance proposed by [15].

The paper contributions are: (i) we proposed immediate reward functions based on Euclidean, n-norm, and Chebyshev distances; (ii) we provided an algorithm to combine various immediate reward functions and use them based on the maximum returned value during each episode to enhance the agents' cumulative reward with and without malicious agents within the MARL system; (iii) we proved the superiority of Euclidean

immediate reward function over Manhattan immediate reward function; (iv) we have shown the superiority of Chebyshev immediate reward function over the Euclidean immediate reward function; and (v) we have shown that the combined immediate reward function outperforms other immediate reward functions.

B. Related Research

Consensus control in MAS has been studied in various situations, e.g. distributed optimal consensus algorithm [16], distributed linear-quadratic regulator (LQR) consensus control for heterogeneous MAS [17] or consensus control under delayed information [18], and has been extensively investigated for different systems such as linear and nonlinear systems [19], [20], [21], [22], [23], [24], [25], [26]. In recent years, the consensus problem for MARL systems has also been researched, e.g. an optimal bipartite consensus control framework designed for MARL systems including model-free structure [27]. An integrated, resilient, model-free, off-policy, distributed state-feedback control protocol for leader-follower MARL system with adversarial inputs to reach consensus on the leader's state is proposed in [28], where only the leader can communicate with real information. The rest of the agents use a distributed observer to estimate the leader's state. The MARL system without malicious agents is implemented in [29], and the sum of the cumulative rewards is calculated and maximized. Inspired by [28], [29], we study a position consensus and we propose immediate reward functions that increase the cumulative reward with the presence of malicious agents in a leaderless MARL system. All agents of the on-policy system can communicate with the environment and receive the related states.

In order to use RL techniques in MAS, the authors of [30], [31], [32] have proposed actor-critic algorithms where the actor part is utilized for training of decentralized policies corresponding to each agent. Critic part is used for learning centralized value function including all agents information. We use the decentralized actor-critic method [29] and the multi-agent actor-critic algorithm under adversarial attacks proposed in [15]. The authors in [15] have shown that the algorithm introduced in [29] for the consensus of MARL systems is not robust to adversarial attacks. In this paper, we present results in the presence of malicious agents by changing the immediate reward function of the RL algorithm. We propose four different distance-based immediate reward functions to select the best one (based on the results) and study their effects on the MARL system's cumulative reward. Our analysis of distance-based immediate reward functions shows that if a distance metric (e.g., Manhattan distance, Euclidean distance, $n-$norm distance, or Chebyshev distance) provides a smaller value between the current position and the desired position compared with the existing distance metrics, then the defined immediate reward function based on that distance metric will improve the MARL system performance.

Though different immediate rewards have already been introduced for various RL algorithms, studies of their effects on the MARL systems, with and without a malicious agent, are lacking. Defining the reward function according to the overall system's objective is preferred, as noted in [33], [34]. As the objective of the presented system is to reach a position consensus, considering the distance between the current position and desired position plays a significant role in achieving the consensus. Therefore, the main advantage of the proposed method is that the defined immediate reward functions are formulated using various distance metrics, e.g., Manhattan, Euclidean, $n-$norm, and Chebyshev distances, and their superiority over the Manhattan immediate reward function (as already used in [15]) have been proven. The superiority of the proposed immediate reward functions leads to higher average cumulative rewards for each agent. In addition, there is a greater average cumulative team reward for the entire MARL system. The MARL system performance is improved by obtaining a higher cumulative team reward, causing a higher percentage of correct actions are executed over time to achieve consensus, resulting in a position consensus with a lower error rate.

II. Background

For decision-making, each agent applies the information received from the environment. The finite Markov decision process (MDP) is considered to represent the dynamics of the environment for decision-making in choosing the best action. An MDP for a MARL system can be defined by a 5-tuple $M = \langle S, A, T, R, \gamma \rangle$, where $S = S_1 \times S_2 \times ... \times S_N$ is a finite set and Cartesian product of environmental states, $A = A_1 \times A_2 \times ... \times A_N$ is a finite joint action set for all agents so that $A_i = a_1 \times a_2 \times ... \times a_K$, $i = 1, 2, ..., N$, is a set of actions of each agent, $T : S \times A \times S^{'} \rightarrow [0, 1]$, describes the environment's dynamics is the state-transition probability function that agents starts in state S, takes action A, and ends in state $S^{'}$. Further, $R : S \times A \times S^{'} \rightarrow \mathbb{R}^n$ is a reward function. For i^{th} agent $R^i_{t+1} = \mathbb{E}\left[r^i_{t+1} | s_t = s, a^i_t = a\right]$, where r^i_{t+1} indicates the immediate reward, s_t shows the state, and a^i_t is action at time t. In an MDP for a MARL system, the cumulative reward is expected to be maximized for all agents, as well as the team of agents [35]. The trade-off between an immediate reward and potential future reward is determined by the discount factor $\gamma \in [0, 1)$.

The MAS is considered as the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of all vertices (agents), and $\mathcal{E} \subseteq \{(i, j) | i \in \mathcal{V}, j \in \mathcal{V}\}$ is the set of all edges (communication links between agents). The agents i and j are neighbors if and only if $(i, j) \in \mathcal{E}$.

III. Method

This paper investigates increasing the agents' cumulative reward for two scenarios: with and without malicious agents in the MARL system.

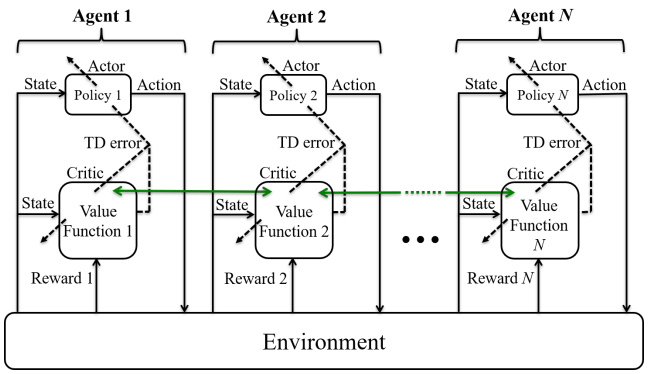


Fig. 1. Multi-agent actor-critic architecture with N agents. The green arrows indicate transferring correct data between neighboring agents.

A. Without Malicious Agents

The goal of each agent is to reach the position consensus in the environment with the maximum cumulative reward. The considered environment in this paper is a grid world. We consider a MAS with corresponding actor-critic architecture [29]. An actor-critic architecture is assigned to each agent in the MAS, where the neighboring agents communicate with each other via the critic unit as illustrated in Fig. 1. Each agent is trained to learn the local policy utilizing decentralized learning.

The set of independent local policies for N agents is described as $Policy = \{\pi_1, \pi_2, ..., \pi_N\}$. At time $t = 0$, all agents are assigned the initial state s_0. The actor unit of i^{th} agent uses the policy function $\pi_i(a_0^i|s_0)$ to perform the action a_0^i related to the initial state s_0. At time $t+1$, all agents receive state s_{t+1}, as well as a local immediate reward r_{t+1}^i from the environment according to the action a_t^i they performed at time t. Each agent keeps the immediate reward information r_{t+1}^i; however, they are permitted to estimate the immediate reward of the network. Based on the reward r_{t+1}^i and state s_{t+1}, the critic unit of i^{th} agent examines whether the actor unit has taken the appropriate action to improve the agent's selection in the following steps. For this purpose, at time $t+1$, the critic unit estimates the reward $\hat{r}_{t+1}^i$ and compares it with the environmental received reward r_{t+1}^i. The estimation of $\hat{r}_{t+1}^i$ is done using a four-layer NN including input layer (environmental received states are fed to this layer), two hidden dense layers, and a dense output layer to return the estimated reward $\hat{r}_{t+1}^i$.

The comparison between the estimated reward $\hat{r}_{t+1}^i$ and the environmental received reward r_{t+1}^i is carried out using the temporal difference (TD) error. The higher value of the TD error means the greater difference between the actual reward r_{t+1}^i and expected reward $\hat{r}_{t+1}^i$. The TD error for the i^{th} agent, δ_t^i, is given by

$$\delta_t^i = R_{t+1}^i + \gamma V_t^i(s_{t+1}) - V_t^i(s_t), \tag{1}$$

where $V_t^i(s_t)$ is the critic value function at time t defined by

$$V_t^i(s_t) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_{t+1}^i | s_t = s\right]. \tag{2}$$

Using (1) and (2), the TD error method yields

$$V_{t+1}^i(s_t) = V_t^i(s_t) + \alpha\delta_t^i, \tag{3}$$

where α is the learning rate. The TD error value of i^{th} agent is sent to the actor unit of the current agent to improve the following action selection, as well as to the critic units of the neighboring agents through the communication links using a consensus protocol.

In the utilized algorithm, the consensus step is as follows:

$$\begin{cases} \lambda_{t+1}^i \leftarrow \sum_{j\in\mathcal{N}} w_t(i,j)\cdot\tilde{\lambda}_t^j, \\ v_{t+1}^i \leftarrow \sum_{j\in\mathcal{N}} w_t(i,j)\cdot\tilde{v}_t^j, \end{cases} \tag{4}$$

where λ and $\tilde{\lambda}$ are the actual and estimated multi-agent reward function parameters, respectively. Furthermore, v and $\tilde{v}$ are the actual and estimated multi-agent value function parameters, respectively. According to [15], and [29] the initialization of the parameters is performed for λ, $\tilde{\lambda}$, v, and $\tilde{v}$ at time $t = 0$ for all N agents. The discussed parameters should be updated and added to the list of previous values at $t+1$. Moreover, v^i of each agent describes the network value function approximation $V_t^i(s_t; v_t^i)$. Hence,

$$\tilde{\lambda}_t^i \leftarrow \lambda_t^i + \alpha_{v,t}\left(r_{t+1}^i - \hat{r}_{t+1}\left(\lambda_t^i\right)\right)\nabla_\lambda \hat{r}_{t+1}\left(\lambda_t^i\right) \tag{5}$$

$$\tilde{v}_t^i \leftarrow v_t^i + \alpha_{v,t}\delta_t^i \nabla_v V_t^i\left(s_t; v_t^i\right). \tag{6}$$

Besides, $\mathcal{N}$ is the set of neighbors of i^{th} agent, and $W_t = [w_t(i,j)]_{N\times N}$ is Metropolis weight matrix specified by

$$W_t = \begin{cases} \dfrac{1}{1+\max\{d_t(i), d_t(j)\}} & \text{if } (i,j)\in\mathcal{E}, \\ 1 - \displaystyle\sum_{(i,k)\in\mathcal{E}} W_t(i,k) & \text{if } i = j, \\ 0 & \text{otherwise}, \end{cases} \tag{7}$$

with $d_t(i)$ and $d_t(j)$ being the degree of agents i and j, respectively. Therefore, the weight on the message transferred form agent i to agent j at time t is $w_t(i,j)$. The consensus step (4) must be done by all N agents in the MARL system to reach the position consensus. Updating the reward function parameter λ^i and value function parameter v^i enables the i^{th} agent to update its policy function $\pi_i(a_t^i|s_t)$. Note that the structure of weight matrix W_t depends on the communication graph topology [29].

B. With Malicious Agents

One of the problems that can occur with any MAS is an incompatibility of one or more agents with the other agents. These types of agents, termed malicious agents, may be internally disturbed and can have a negative effect on MAS

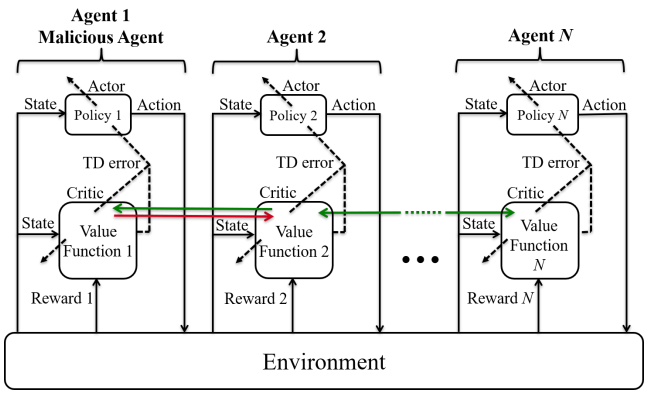


Fig. 2. Multi-agent actor-critic architecture with N agents. The green arrows indicate transferring correct data between neighboring agents, and the red arrow represents transmitting inaccurate data from malicious agent to neighboring agents.

performance. In this paper, the malicious agent does not apply the consensus updates and skips the consensus step of (4), which results in an unbalanced consensus throughout the entire MAS [15]. An actor-critic MARL system with a malicious agent is illustrated in Fig. 2. At the time t, the malicious agent receives correct data from the critic units of neighboring agents, including TD error. However, the malicious agent sends inaccurate information to neighboring agents' critic units via communication links or performs adverse actions, causing this agent to maximize its cumulative reward. Simultaneously, the cumulative rewards of neighboring agents are reduced due to improper information they receive from the malicious agent. MAS's cumulative team reward is reduced compared to the situation where there is no malicious agent in the system [15]. In the following, the immediate reward function and its effect on the system performance are analyzed.

C. Reward Functions

Choosing an appropriate reward function is a significant challenge in RL algorithms. There is no specific rule to select or define an immediate reward function. In general, one should select the immediate reward function based on the RL system's application. We consider five various immediate reward functions, based on multiple distance metrics, to reach the position consensus. For all the following distance metrics and immediate reward functions, (x, y) and (x_{des}, y_{des}) are the current position and the desired position, respectively. Moreover, (x^i, y^i) and (x^i_{des}, y^i_{des}) are the current position and the desired position of the i^{th} agent, sequentially.

1) Manhattan Immediate Reward: Each agent's immediate reward in $2D$ space is determined based on the Manhattan distance: $M_d = |x-x_{des}|+|y-y_{des}|$. The Manhattan immediate reward function for i^{th} agent is given by [15]:

$$Mr^i_{t+1} = -\left|x^i - x^i_{des}\right| - \left|y^i - y^i_{des}\right|. \tag{8}$$

2) Euclidean Immediate Reward: Based on the Euclidean distance E_d, we define the Euclidean immediate reward function for the i^{th} agent in $2D$ space

$$Er^i_{t+1} = -\left(\left|x^i - x^i_{des}\right|^2 + \left|y^i - y^i_{des}\right|^2\right)^{1/2}. \tag{9}$$

3) n-norm Immediate Reward: Using the n-norm metric $N_d = (|x - x_{des}|^n + |y - y_{des}|^n)^{1/n}$, the immediate reward function for the i^{th} agent in $2D$ space is given by

$$Nr^i_{t+1} = -\left(\left|x^i - x^i_{des}\right|^n + \left|y^i - y^i_{des}\right|^n\right)^{1/n}, \tag{10}$$

where $n \geq 3$.

4) Chebyshev Immediate Reward: Utilizing the Chebyshev distance metric $\check{C}_d = \max(|x-x_{des}|, |y-y_{des}|)$, the Chebyshev immediate reward function for i^{th} agent in $2D$ space is given by

$$\check{C}r^i_{t+1} = \max\left(-\left|x^i - x^i_{des}\right|, -\left|y^i - y^i_{des}\right|\right). \tag{11}$$

5) Combined Immediate Reward: Based on the immediate reward functions (8)-(11), the combined immediate reward function for i^{th} agent in $2D$ space is given by

$$Cr^i_{t+1} = \max\left(Mr^i_{t+1}, Er^i_{t+1}, Nr^i_{t+1}, \check{C}r^i_{t+1}\right). \tag{12}$$

Equation (12), in each episode and for all agents, calculates the various immediate rewards of (8)-(11) and selects the maximum reward based on the returned values. This method uses other immediate reward functions to get the largest cumulative reward during each episode.

Theorem 1. *Let Mr^i_{t+1} and Er^i_{t+1} be Manhattan and Euclidean immediate reward functions for i^{th} agent in $2D$ space, then the Euclidean cumulative team reward is greater than or equal to the Manhattan cumulative team reward for N agents in $2D$ space.*

Proof. For i^{th} agent in $2D$ space, $|\Delta x| = |x^i - x^i_{des}|$, and $|\Delta y| = |y^i - y^i_{des}|$. According to $|\Delta x|^2 + |\Delta y|^2 \leq (|\Delta x| + |\Delta y|)^2$ and by considering the positive roots of both sides of inequality, the following is valid

$$\begin{aligned} &-\left|x^i - x^i_{des}\right| - \left|y^i - y^i_{des}\right| \leq \\ &-\left(\left|x^i - x^i_{des}\right|^2 + \left|y^i - y^i_{des}\right|^2\right)^{1/2}. \end{aligned} \tag{13}$$

From (13), it is concluded that the Manhattan immediate reward is less than or equal to the Euclidean immediate reward. Therefore, from $Mr^i_{t+1} \leq Er^i_{t+1}$ it is obvious that

$$\mathbb{E}\left[Mr^i_{t+1} | s_t = s, a^i_t = a\right] \leq \mathbb{E}\left[Er^i_{t+1} | s_t = s, a^i_t = a\right] \tag{14}$$

$$MR^i_{t+1} \leq ER^i_{t+1}, \tag{15}$$

where MR^i_{t+1} and ER^i_{t+1} are Manhattan and Euclidean cumulative rewards for i^{th} agent in $2D$ space, respectively. Using (15) we have

$$\frac{1}{N}\sum_{i=1}^{N} MR^i_{t+1} \leq \frac{1}{N}\sum_{i=1}^{N} ER^i_{t+1}. \tag{16}$$

Hence, the Euclidean cumulative team reward is greater than or equal to the Manhattan cumulative team reward for N agents in $2D$ space. □

Remark. *Since Manhattan and Euclidean distances are called* 1*-norm and* 2*-norm distances, respectively, then the proof of Theorem 1 can be expanded to show that the* n*-norm cumulative team reward (*$n \geq 3$*) is greater than or equal to the Euclidean cumulative team reward for* N *agents in* $2D$ *space.*

Theorem 2. *Let* Er^i_{t+1} *and* $\check{C}r^i_{t+1}$ *be Euclidean and Chebyshev immediate reward functions for* i^{th} *agent in* $2D$ *space, then the Chebyshev cumulative team reward is greater than or equal to the Euclidean cumulative team reward for* N *agents in* $2D$ *space.*

Proof. Based on triangle inequality, the product of Chebyshev distance is always less than or equal to the outcome of Euclidean distance ($\check{C}_d \leq E_d$). Hence, the following is valid

$$-\left(|\Delta x|^2 + |\Delta y|^2\right)^{1/2} \leq -\max\left(|\Delta x|, |\Delta y|\right). \tag{17}$$

We know that

$$-\max\left(|\Delta x|, |\Delta y|\right) \leq \max\left(-|\Delta x|, -|\Delta y|\right). \tag{18}$$

Using (17) and (18) yields

$$\begin{aligned} &-\left(\left|x^i - x^i_{des}\right|^2 + \left|y^i - y^i_{des}\right|^2\right)^{1/2} \leq \\ &\max\left(-\left|x^i - x^i_{des}\right|, -\left|y^i - y^i_{des}\right|\right). \end{aligned} \tag{19}$$

From (19), it is derived that $Er^i_{t+1} \leq \check{C}r^i_{t+1}$. Afterward,

$$\mathbb{E}\left[Er^i_{t+1}|s_t = s, a^i_t = a\right] \leq \mathbb{E}\left[\check{C}r^i_{t+1}|s_t = s, a^i_t = a\right] \tag{20}$$

$$ER^i_{t+1} \leq \check{C}R^i_{t+1}, \tag{21}$$

where $\check{C}R^i_{t+1}$ is the Chebyshev cumulative reward for i^{th} agent in $2D$ space. Using (21) we have

$$\frac{1}{N}\sum_{i=1}^{N} ER^i_{t+1} \leq \frac{1}{N}\sum_{i=1}^{N} \check{C}R^i_{t+1}. \tag{22}$$

Therefore, the Chebyshev cumulative team reward is greater than or equal to the Euclidean cumulative team reward for N agents in $2D$ space. □

Theorem 3. *Let* Mr^i_{t+1}*,* Er^i_{t+1}*,* Nr^i_{t+1}*,* $\check{C}r^i_{t+1}$*, and* Cr^i_{t+1} *be Manhattan, Euclidean,* n*-norm, Chebyshev, and combined immediate reward functions for* i^{th} *agent in* $2D$ *space, respectively. Then, the combined cumulative team reward for* N *agents is greater than or equal to the maximum of the Manhattan, Euclidean,* n*-norm, and Chebyshev cumulative team rewards for the same* N *agents in* $2D$ *space during each episode.*

Proof. From (12) it follows

$$\begin{aligned} &\mathbb{E}\left[Cr^i_{t+1}|s_t = s, a^i_t = a\right] \geq \\ &\mathbb{E}\left[Mr^i_{t+1}, Er^i_{t+1}, Nr^i_{t+1}, \check{C}r^i_{t+1}|s_t = s, a^i_t = a\right]. \end{aligned} \tag{23}$$

Given that Mr^i_{t+1}, Er^i_{t+1}, Nr^i_{t+1}, and $\check{C}r^i_{t+1}$ are independent functions, also by taking the maximum function from both sides of inequality, one has

$$\begin{aligned} &\mathbb{E}\left[Cr^i_{t+1}|s_t = s, a^i_t = a\right] \geq \\ &\max\Big(\mathbb{E}\left[Mr^i_{t+1}|s_t = s, a^i_t = a\right], \mathbb{E}\left[Er^i_{t+1}|s_t = s, a^i_t = a\right], \\ &\mathbb{E}\left[Nr^i_{t+1}|s_t = s, a^i_t = a\right], \mathbb{E}\left[\check{C}r^i_{t+1}|s_t = s, a^i_t = a\right]\Big). \end{aligned} \tag{24}$$

As a consequence, we have

$$CR^i_{t+1} \geq \max\left(MR^i_{t+1}, ER^i_{t+1}, NR^i_{t+1}, \check{C}R^i_{t+1}\right), \tag{25}$$

where NR^i_{t+1} and CR^i_{t+1} are n-norm and combined cumulative rewards, respectively. From (25), it follows that the combined cumulative reward for i^{th} agent is greater than or equal to the maximum of Manhattan, Euclidean, n-norm, and Chebyshev cumulative rewards for the same agent in $2D$ space during each episode. Therefore,

$$\begin{aligned} &\frac{1}{N}\sum_{i=1}^{N} CR^i_{t+1} \geq \\ &\sum_{i=1}^{N} \max\left(\frac{1}{N}MR^i_{t+1}, \frac{1}{N}ER^i_{t+1}, \frac{1}{N}NR^i_{t+1}, \frac{1}{N}\check{C}R^i_{t+1}\right). \end{aligned} \tag{26}$$

We know that

$$\begin{aligned} &\sum_{i=1}^{N} \max\left(\frac{1}{N}MR^i_{t+1}, \frac{1}{N}ER^i_{t+1}, \frac{1}{N}NR^i_{t+1}, \frac{1}{N}\check{C}R^i_{t+1}\right) \geq \\ &\max \sum_{i=1}^{N}\left(\frac{1}{N}MR^i_{t+1}, \frac{1}{N}ER^i_{t+1}, \frac{1}{N}NR^i_{t+1}, \frac{1}{N}\check{C}R^i_{t+1}\right). \end{aligned} \tag{27}$$

Using (26) and (27) yields

$$\begin{aligned} &\frac{1}{N}\sum_{i=1}^{N} CR^i_{t+1} \geq \\ &\max\left(\frac{1}{N}\sum_{i=1}^{N} MR^i_{t+1}, \frac{1}{N}\sum_{i=1}^{N} ER^i_{t+1}, \right. \\ &\left.\frac{1}{N}\sum_{i=1}^{N} NR^i_{t+1}, \frac{1}{N}\sum_{i=1}^{N} \check{C}R^i_{t+1}\right). \end{aligned} \tag{28}$$

Hence, at time t, the combined cumulative team reward for N agents is greater than or equal to the maximum of the Manhattan, Euclidean, n-norm, and Chebyshev cumulative team rewards for the same N agents in $2D$ space during each episode. □

Theorem 4. *Let* Mr^i_{t+1}*,* Er^i_{t+1}*,* Nr^i_{t+1}*,* $\check{C}r^i_{t+1}$*, and* Cr^i_{t+1} *be Manhattan, Euclidean,* n*-norm, Chebyshev, and combined immediate reward functions for* i^{th} *agent in* $2D$ *space, respectively. Then, the combined critic value for* i^{th} *agent is greater*

than or equal to the maximum of Manhattan, Euclidean, n-norm, and Chebyshev critic values for the same agent in $2D$ space during each episode.

Proof. Since $\gamma \in [0,1)$ and $t \in [0,\infty)$, it is concluded that $\gamma^t \in [0,1]$. Therefore, from (12) we have

$$\sum_{t=0}^{\infty} \gamma^t C r_{t+1}^i = \sum_{t=0}^{\infty} \max\left(\gamma^t M r_{t+1}^i, \gamma^t E r_{t+1}^i, \gamma^t N r_{t+1}^i, \gamma^t \check{C} r_{t+1}^i\right). \tag{29}$$

We know that

$$\sum_{t=0}^{\infty} \max\left(\gamma^t M r_{t+1}^i, \gamma^t E r_{t+1}^i, \gamma^t N r_{t+1}^i, \gamma^t \check{C} r_{t+1}^i\right) \geq \max \sum_{t=0}^{\infty} \left(\gamma^t M r_{t+1}^i, \gamma^t E r_{t+1}^i, \gamma^t N r_{t+1}^i, \gamma^t \check{C} r_{t+1}^i\right). \tag{30}$$

After simplifying, using (29) and (30) yields

$$\sum_{t=0}^{\infty} \gamma^t C r_{t+1}^i \geq \sum_{t=0}^{\infty} \left(\gamma^t M r_{t+1}^i, \gamma^t E r_{t+1}^i, \gamma^t N r_{t+1}^i, \gamma^t \check{C} r_{t+1}^i\right). \tag{31}$$

By taking expectation with respect to the state from both sides of inequality, the following is achieved

$$\mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t C r_{t+1}^i | s_t = s\right] \geq \mathbb{E}\left[\left(\sum_{t=0}^{\infty} \gamma^t M r_{t+1}^i, \sum_{t=0}^{\infty} \gamma^t E r_{t+1}^i, \sum_{t=0}^{\infty} \gamma^t N r_{t+1}^i, \sum_{t=0}^{\infty} \gamma^t \check{C} r_{t+1}^i\right) \Big| s_t = s\right]. \tag{32}$$

Since $\sum_{t=0}^{\infty} \gamma^t M r_{t+1}^i$, $\sum_{t=0}^{\infty} \gamma^t E r_{t+1}^i$, $\sum_{t=0}^{\infty} \gamma^t N r_{t+1}^i$, and $\sum_{t=0}^{\infty} \gamma^t \check{C} r_{t+1}^i$ are statistically independent, after simplifying and taking the maximum function from both sides of (32), the following is obtained

$$\mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t C r_{t+1}^i | s_t = s\right] \geq \max\left(\mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t M r_{t+1}^i | s_t = s\right], \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t E r_{t+1}^i | s_t = s\right], \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t N r_{t+1}^i | s_t = s\right], \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \check{C} r_{t+1}^i | s_t = s\right]\right). \tag{33}$$

Therefore,

$$CV_t^i(s_t) \geq \max\left(MV_t^i(s_t), EV_t^i(s_t), NV_t^i(s_t), \check{C}V_t^i(s_t)\right), \tag{34}$$

where $MV_t^i(s_t)$, $EV_t^i(s_t)$, $NV_t^i(s_t)$, $\check{C}V_t^i(s_t)$, and $CV_t^i(s_t)$ are Manhattan, Euclidean, n-norm, Chebyshev, and combined critic value functions. Therefore, the combined critic value for i^{th} agent is greater than or equal to the maximum of Manhattan, Euclidean, n-norm, and Chebyshev critic values for the same agent in $2D$ space during each episode. □

IV. Results and Discussion

This section demonstrates results for consensus control of MAS with and without malicious agents, using the RL decentralized actor-decentralized critic method. To reach the position consensus, a fully connected graph $\mathcal{G}$ is considered, which is illustrated in Fig. 4. Each actor's internal structure consists of a fully-connected NN architecture for training, including three dense layers with Adam optimizer and categorical cross-entropy loss function. The first and second layers' activation functions are rectified linear unit (ReLU) functions, and the third layer has the softmax activation function.

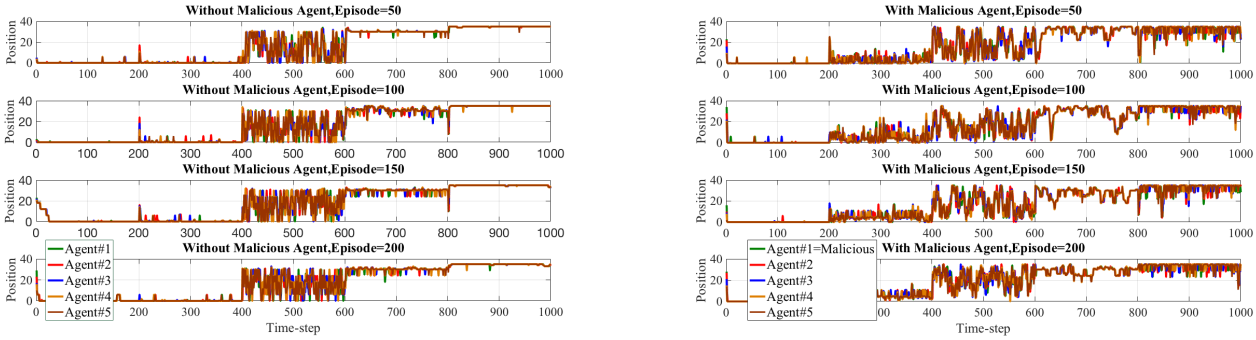
Similar to the actor, each agent's critic has a three-layer, fully-connected NN structure, including the ReLU activation functions for the first two layers, utilizing Adam optimizer and mean squared error (MSE) loss function. The NN structure for training the reward function is similar to the architecture used for training all agents critic. This section's results are derived from MAT-files, obtained by training the NN above for each agent. Each MAT-file is a cell of 200 structures (number of episodes to train); each structure contains state, action, reward, and predicted reward for five agents in 1000 time-steps. In this paper, evaluating the performance of the utilized RL algorithm is done by considering how much reward each agent and a team of agents receive while acting, and then showing the cumulative reward as a function of the episodes and number of steps.

First, reaching the position consensus on the X-axis is shown in the 1000 time-steps for five agents, using the Manhattan immediate reward function. Then, the average reward during 200 episodes is displayed using five immediate reward functions. Afterward, each agent's average cumulative reward and the average cumulative team reward using different immediate reward functions during 200 episodes with and without a malicious agent are compared. Note that action space consists of five distinctive actions, including waiting and also move to the right, left, up, and down. The actor and critic learning rates are $\alpha = 0.001$ and $\alpha = 0.01$, respectively, and the discount factor is $\gamma = 0.95$.

We have used and extended a part of the code provided in [36] for a part of our implementation. Moreover, the algorithm's execution is done using a system with 3.60 GHz Intel Core i7 − 7700 processor, 16 GB installed RAM, 64−bit operating system, and x64−based processor.

A. Reaching Consensus

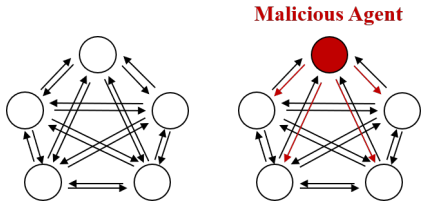
The position consensus of five agents on the X-axis with and without a malicious agent is illustrated in Fig. 3 at episodes 50,



(a) Position consensus of $N = 5$ agents on X-axis without malicious agents at episodes 50, 100, 150, 200.

(b) Position consensus of $N = 5$ agents on X-axis with a malicious agent at episodes 50, 100, 150, 200.

Fig. 3. The MARL system's performance in reaching the consensus without and with a malicious agent during 200 episodes.



(a) Topology of a fully connected graph $\mathcal{G}$ without a malicious agent.

(b) Topology of a fully connected graph $\mathcal{G}$ with a malicious agent.

Fig. 4. A fully connected graph $\mathcal{G}$ is considered as the MARL system, including $N = 5$ nodes. The malicious agent (red circle) refuses to update the parameters in the consensus step.

100, 150, and 200. This consensus is demonstrated during 1000 time-steps using Manhattan immediate reward function. The initial position for i^{th} agent is randomly selected. The desired position for i^{th} agent is $x^i_{des} = 35$. As shown in Fig. 3, the position convergence of the MARL system in the absence of a malicious agent is superior to the position convergence with a malicious agent's presence during 200 episodes. According to Fig. 3(a), the agents' convergence behavior is observed in the episode 50; however, according to Fig. 3(b), this behavior has not appeared during 200 episodes. The cumulative team reward of the system without malicious agents is greater than the system's cumulative team reward with a malicious agent (Fig. 5). Accordingly, the MARL system's performance in reaching the consensus without a malicious agent is superior to the network performance with a malicious agent during 200 episodes. Therefore, to improve the network performance in Fig. 3(b), the system's cumulative team reward with the malicious agent should increase, which we will examine in the following.

B. Increasing the Cumulative Reward

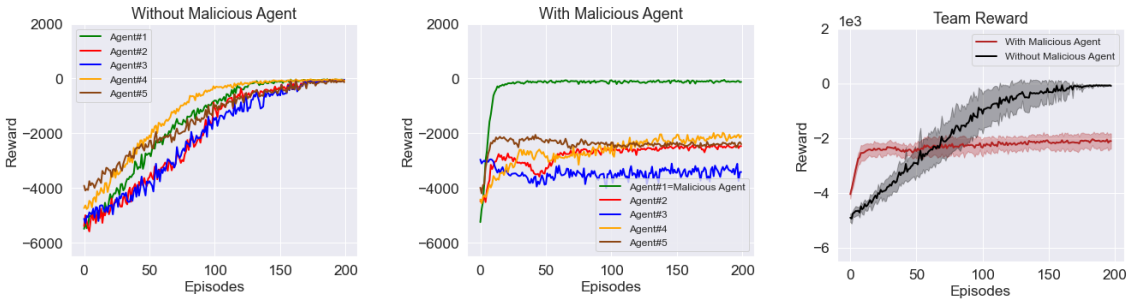
When no malicious agents exist in the MARL system, the agents' goal is to maximize the sum of all cumulative rewards. Fig. 5(a) shows the reward vs. episodes diagram of five agents without malicious agents. The cumulative reward of all agents reaches the maximum value during 200 episodes. As Fig. 5(a) shows, all agents have learned the optimal policy almost equally and have maximized their cumulative reward. We examine the agents' reaction of a MARL system if a malicious agent is detected within the system.

An agent is considered as a malicious agent when it seeks to maximize its own cumulative reward only. The reward vs. episodes diagram of five agents with the malicious agent's presence is illustrated in Fig. 5(b) during 200 episodes. Indeed, `Agent#1` is the malicious agent, and its cumulative reward is maximized. The other four agents are not able to maximize their cumulative reward as much as they did in the previous step and cannot learn the optimal policy precisely. However, they enhanced their cumulative reward. Thus, as shown in Fig. 5(c), the cumulative team reward of the MARL system converges without the presence of a malicious agent and is superior to the cumulative team reward of the MARL system with the presence of a malicious agent. The malicious agent has caused the cumulative team reward to converge to -2291.05. All diagrams of Fig. 5 are obtained using the Manhattan immediate reward function defined in (8).

C. Modifying the Immediate Reward Function

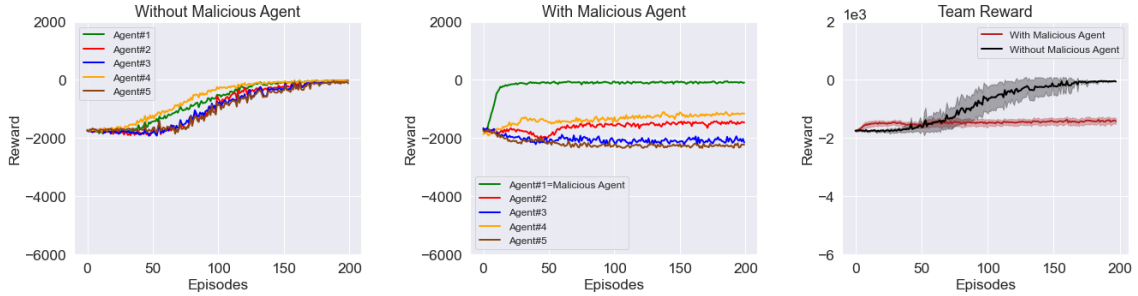
The experiment is repeated with the same conditions but using the proposed Euclidean, n-norm ($n = 5$), Chebyshev, and combined immediate reward functions, (9)-(12).

As shown in Fig. 6(a) and Fig. 6(b), the cumulative reward in both cases, without and with a malicious agent, have converged using the Euclidean immediate reward function in (9). The outcomes of using (9) is superior to the results of (8) because, as illustrated in Fig. 6(c), the MARL system's cumulative team



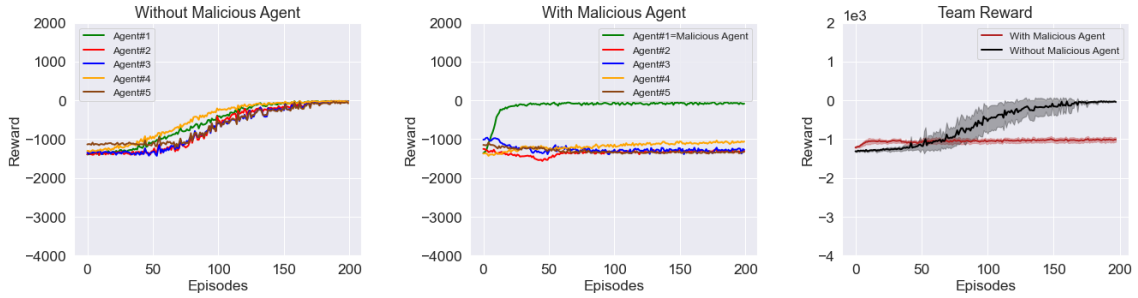
(a) Average reward without malicious agents. (b) Average reward with a malicious agent. (c) Cumulative team reward with and without a malicious agent.

Fig. 5. Reward convergence using the Manhattan immediate reward function during 200 episodes for $N = 5$ agents [15].



(a) Average reward without malicious agents. (b) Average reward with a malicious agent. (c) Cumulative team reward with and without a malicious agent.

Fig. 6. Reward convergence using the Euclidean immediate reward function during 200 episodes for $N = 5$ agents.

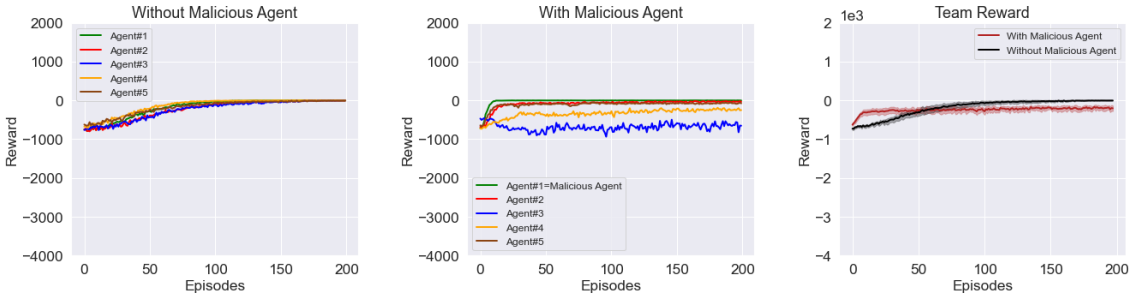


(a) Average reward without malicious agents. (b) Average reward with a malicious agent. (c) Cumulative team reward with and without a malicious agent.

Fig. 7. Reward convergence using the 5-norm immediate reward function during 200 episodes for $N = 5$ agents.

(a) Average reward without malicious agents. (b) Average reward with a malicious agent. (c) Cumulative team reward with and without a malicious agent.

Fig. 8. Reward convergence using the Chebyshev immediate reward function during 200 episodes for $N = 5$ agents.



(a) Average reward without malicious agents.

(b) Average reward with a malicious agent.

(c) Cumulative team reward with and without a malicious agent.

Fig. 9. Reward convergence using the combined immediate reward including Manhattan, Euclidean, 5-norm, and Chebyshev immediate reward functions during 200 episodes for $N = 5$ agents.

reward with a malicious agent is larger than demonstrated results in Fig. 5(c). The cumulative team reward with a malicious agent has converged to -1468.74 using (9). Hence, as shown in Fig. 5 and Fig. 6, the use of (9) yields better results compared to (8).

We repeated the experiment using (10) where $n = 5$ (5-norm immediate reward function). As demonstrated in Fig. 7, compared to the Fig. 5 and Fig. 6 the average received reward enhances for each agent and system by increasing n in the n-norm immediate reward function. For instance, the cumulative team reward with a malicious agent has converged to -1045.88 using (10), where $n = 5$.

As shown in Fig. 8(a) and Fig. 8(b), the cumulative reward, without and with a malicious agent, have converged by applying the Chebyshev immediate reward function in (11). The results of using (11) are superior to (8)-(10), because, as illustrated in Fig. 8(c), the MARL system's cumulative team reward with a malicious agent is larger than demonstrated results in Figs. 5(c)-7(c). The cumulative team reward has converged to -369.11 using (11). Consequently, as displayed in Figs. 5-8, the use of (11) has yielded more reliable results compared to (8)-(10). Using (11), the average received reward is higher for each agent and MARL system.

The outcomes of using (12) are superior to the results of (8)-(11), because, as shown in Fig. 9(c), the MARL system's cumulative team reward in the presence of a malicious agent is larger than illustrated results in Figs. 5(c)-8(c). The cumulative team reward has converged to -244.78 using (12). Hence, as demonstrated in Fig. 9, the use of (12) has produced superior results compared to previously introduced immediate reward functions. Moreover, the average received reward is higher for each agent and system. The comparison of each agent's average cumulative reward as well as the average cumulative team reward using different immediate reward functions during 200 episodes without and with a malicious agent are indicated in Tables I and II, respectively. As highlighted in these tables, the combined reward performed superior than the other rewards for each agent and team of agents.

TABLE I
COMPARISON OF EACH AGENT'S AVERAGE CUMULATIVE REWARD AS WELL AS THE AVERAGE CUMULATIVE TEAM REWARD USING DIFFERENT IMMEDIATE REWARD FUNCTIONS DURING 200 EPISODES WITHOUT A MALICIOUS AGENT

	Agent 1	**Agent 2**	**Agent 3**	**Agent 4**	**Agent 5**	**Team**
Manhattan Reward	-1566.21	-2011.39	-2093.50	-1209.28	-1490.39	-1641.68
Euclidean Reward	-756.45	-939.41	-961.17	-640.07	-992.70	-849.07
5-norm Reward	-576.56	-703.18	-720.72	-491.55	-655.37	-622.58
Chebyshev Reward	-330.13	-423.84	-430.31	-250.89	-300.96	-335.76
Combined Reward	-176.21	-231.49	-238.25	-135.51	-174.27	-185.91

TABLE II
COMPARISON OF EACH AGENT'S AVERAGE CUMULATIVE REWARD AS WELL AS THE AVERAGE CUMULATIVE TEAM REWARD USING DIFFERENT IMMEDIATE REWARD FUNCTIONS DURING 200 EPISODES IN THE PRESENCE OF A MALICIOUS AGENT

	Agent 1	**Agent 2**	**Agent 3**	**Agent 4**	**Agent 5**	**Team**
Manhattan Reward	-273.01	-2781.51	-3454.21	-2638.68	-2403.99	-2291.05
Euclidean Reward	-164.16	-1606.92	-2047.06	-1349.11	-2190.22	-1468.74
5-norm Reward	-143.52	-1351.53	-1258.49	-1179.52	-1305.38	-1045.88
Chebyshev Reward	-44.15	-143.68	-1062.88	-465.00	-173.53	-369.11
Combined Reward	-21.84	-87.28	-686.82	-338.84	-109.43	-244.78

D. The Immediate Rewards' Comparison After Normalization

To have a valid comparison between the used and proposed immediate reward functions, we normalize the accumulated reward values into a range of $[-1, 0]$ for each agent using

$$R^i_{\text{normalized}} = \frac{R^i - R^i{}_{\min}}{R^i_{\max} - R^i{}_{\min}} - 1, \tag{35}$$

where R^i is the cumulative reward vector for i^{th} agent. Therefore, at this stage, the analysis is performed based on normalized data. Regarding Tables III and IV, as well as, Figs. 10-14 after normalization, still the combined reward performed superior to the other rewards for each agent and team of agents (with and without malicious agents). It is worth mentioning that the data of Tables III and IV are rounded to

TABLE III
COMPARISON OF EACH AGENT'S AVERAGE CUMULATIVE REWARD AS WELL AS THE AVERAGE CUMULATIVE TEAM REWARD AFTER NORMALIZATION USING DIFFERENT IMMEDIATE REWARD FUNCTIONS DURING 200 EPISODES WITHOUT A MALICIOUS AGENT

	Agent 1	**Agent 2**	**Agent 3**	**Agent 4**	**Agent 5**	**Team**
Manhattan Reward	−0.0496	−0.0524	−0.0523	−0.0486	−0.0546	−0.0515
Euclidean Reward	−0.0489	−0.0522	−0.0521	−0.0475	−0.0545	−0.0510
5-norm Reward	−0.0449	−0.0498	−0.0500	−0.0431	−0.0518	−0.0479
Chebyshev Reward	−0.0396	−0.0452	−0.0455	−0.0371	−0.0447	−0.0424
Combined Reward	−0.0381	−0.0427	−0.0431	−0.0364	−0.0444	−0.0409

TABLE IV
COMPARISON OF EACH AGENT'S AVERAGE CUMULATIVE REWARD AS WELL AS THE AVERAGE CUMULATIVE TEAM REWARD AFTER NORMALIZATION USING DIFFERENT IMMEDIATE REWARD FUNCTIONS DURING 200 EPISODES IN THE PRESENCE OF A MALICIOUS AGENT

	Agent 1	**Agent 2**	**Agent 3**	**Agent 4**	**Agent 5**	**Team**
Manhattan Reward	−0.0372	−0.0702	−0.0701	−0.0698	−0.0704	−0.0635
Euclidean Reward	−0.0334	−0.0688	−0.0702	−0.0686	−0.0702	−0.0623
5-norm Reward	−0.0249	−0.0659	−0.0702	−0.0645	−0.0697	−0.0590
Chebyshev Reward	−0.0153	−0.0366	−0.0686	−0.0561	−0.0406	−0.0435
Combined Reward	−0.0152	−0.0348	−0.0681	−0.0513	−0.0387	−0.0417

four decimal places. Furthermore, Fig. 15 depicts the values of Tables III and IV in two different charts. The performance of the malicious agent (`Agent#1`) in increasing its cumulative reward and decreasing the cumulative reward of the other agents is well illustrated in Fig. 15(b). In addition, Fig. 15(b) shows how changing the type of immediate reward function can reduce the negative effect of the malicious agent.

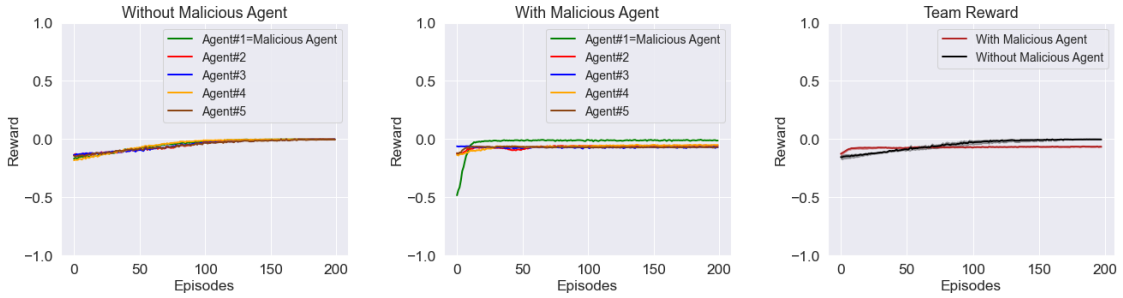
E. Reward Algorithm's Complexity and Execution Time

The comparison of the algorithm execution time of using different immediate reward functions during 200 episodes is presented in Table V. Lower algorithm execution time and higher cumulative team reward are crucial factors in determining the type of the immediate reward function for the MARL system. By comparing the results of using Mr^i_{t+1} provided by [15], and the outcomes of using the proposed immediate reward functions (Er^i_{t+1}, Nr^i_{t+1}, $\check{C}r^i_{t+1}$, and Cr^i_{t+1}) the following results are obtained. To calculate the percentage increase of team reward $\%\text{increase} = 100 \times \frac{\text{(final team reward - initial team reward)}}{|\text{initial team reward}|}$ is used, where initial team reward is the Manhattan team reward. In addition, the Euclidean, 5-norm, Chebyshev, and combined team rewards are considered as the final team reward, each time.

1) Before Normalization: By comparing the results of using Manhattan and Euclidean immediate rewards, it is concluded that after using the Euclidean immediate reward, the $+48.28\%$ and $+35.89\%$, increase in the assertiveness of team reward without and with a malicious agent, respectively. The algorithm execution time using the Euclidean immediate reward function is $+1.76$ times that of the Manhattan immediate reward function. Moreover, by comparing the outcomes of using Manhattan and 5-norm immediate rewards, it is realized that after using the 5-norm immediate reward, the $+62.08\%$ and $+54.35\%$, increase in the assertiveness of team reward without and with a malicious agent, respectively. The algorithm execution time using the 5-norm immediate reward function is $+1.28$ times that of the Manhattan immediate reward function. Furthermore, by comparing the outcomes of using Manhattan and Chebyshev immediate rewards, it is achieved that after using the Chebyshev immediate reward, the $+79.55\%$ and $+83.89\%$, increase in the assertiveness of team reward without and with a malicious agent, respectively. The algorithm execution time using the Chebyshev immediate reward function is $+1.08$ times that of the Manhattan immediate reward function. Besides, by comparing the results of using Manhattan and combined immediate rewards, it is concluded that after using the combined immediate reward, the $+88.68\%$ and $+89.32\%$, increase in the assertiveness of team reward without and with a malicious agent, respectively. The algorithm execution time using the combined immediate reward function is $+3.14$ times that of the Manhattan immediate reward function.

2) After Normalization: By comparing the results of using Manhattan and Euclidean immediate rewards, it is concluded that after using the Euclidean immediate reward, the $+0.97\%$ and $+1.89\%$, increase in the assertiveness of team reward without and with a malicious agent, respectively. The algorithm execution time using the Euclidean immediate reward function is $+1.75$ times that of the Manhattan immediate reward function. Moreover, by comparing the outcomes of using Manhattan and 5-norm immediate rewards, it is realized that after using the 5-norm immediate reward, the $+6.99\%$ and $+7.09\%$, increase in the assertiveness of team reward without and with a malicious agent, respectively. The algorithm execution time using the 5-norm immediate reward function is $+1.29$ times that of the Manhattan immediate reward function. Furthermore, by comparing the outcomes of using Manhattan and Chebyshev immediate rewards, it is noted that after using the Chebyshev immediate reward, the $+17.67\%$ and $+31.50\%$, increase in the assertiveness of team reward without and with a malicious agent, respectively. The algorithm execution time using the Chebyshev immediate reward function is $+1.08$ times that of the Manhattan immediate reward function. Besides, by comparing the results of using Manhattan and combined immediate rewards, it is concluded that after using the combined immediate reward, the $+20.58\%$ and $+34.33\%$, increase in the assertiveness of team reward without and with a malicious agent, respectively. The algorithm execution time using the combined immediate reward function is $+2.95$ times that of the Manhattan immediate reward function.

Table V lists the time complexity of various types of immediate reward algorithms. The time complexity of Manhattan, Euclidean, 5-norm, and Chebyshev immediate reward functions with n point pairs are $O(n)$, and they take linear time. Moreover, the time complexity of the combined immediate reward algorithm is $O(n)$ as well.

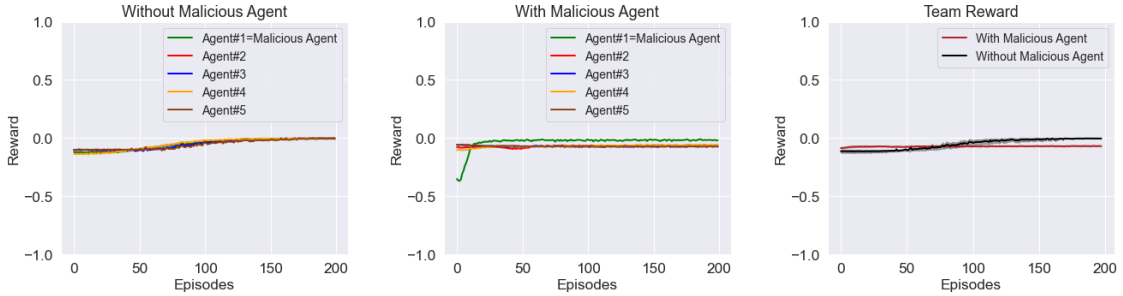


(a) Normalized average reward without malicious agents.

(b) Normalized average reward with a malicious agent.

(c) Normalized cumulative team reward with and without a malicious agent.

Fig. 10. Normalized reward convergence using the Manhattan immediate reward function during 200 episodes for $N = 5$ agents.

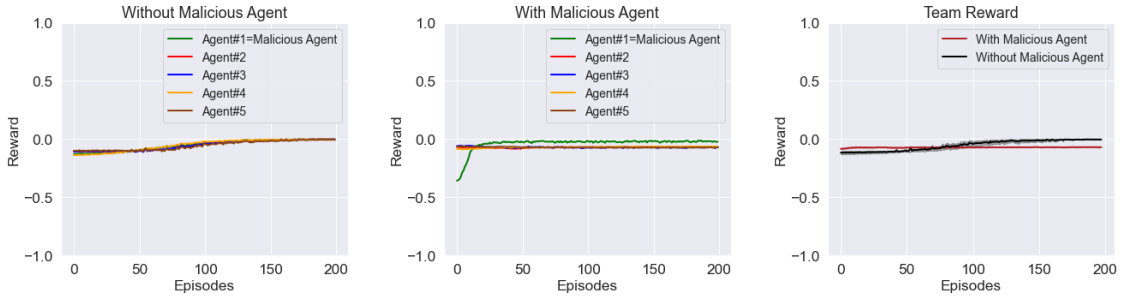


(a) Normalized average reward without malicious agents.

(b) Normalized average reward with a malicious agent.

(c) Normalized cumulative team reward with and without a malicious agent.

Fig. 11. Normalized reward convergence using the Euclidean immediate reward function during 200 episodes for $N = 5$ agents.

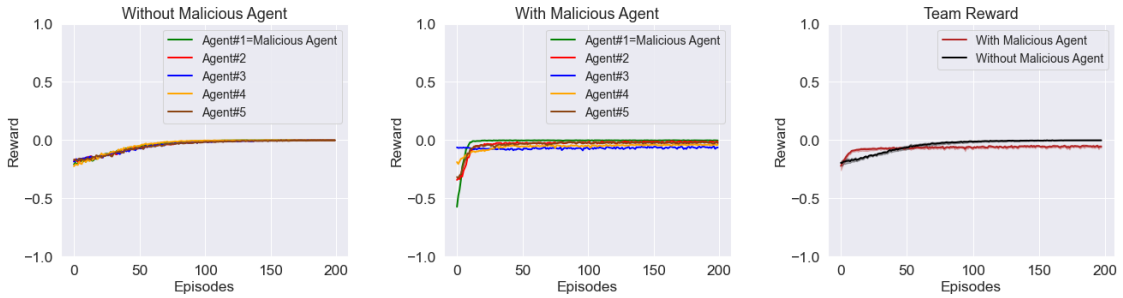


(a) Normalized average reward without malicious agents.

(b) Normalized average reward with a malicious agent.

(c) Normalized cumulative team reward with and without a malicious agent.

Fig. 12. Normalized reward convergence using the 5-norm immediate reward function during 200 episodes for $N = 5$ agents.

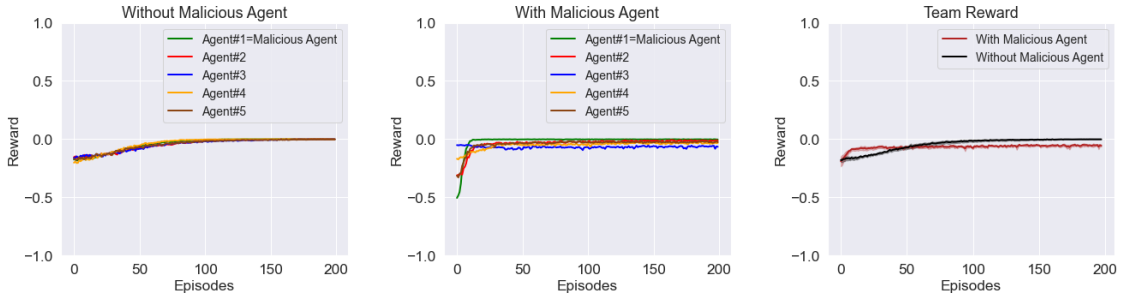


(a) Normalized average reward without malicious agents.

(b) Normalized average reward with a malicious agent.

(c) Normalized cumulative team reward with and without a malicious agent.

Fig. 13. Normalized reward convergence using the Chebyshev immediate reward function during 200 episodes for $N = 5$ agents.

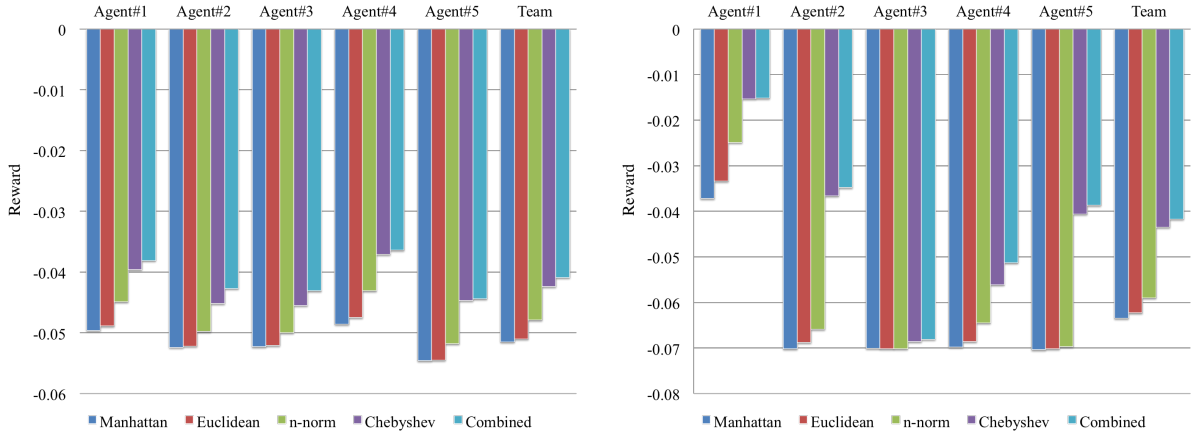


(a) Normalized average reward without malicious agents.

(b) Normalized average reward with a malicious agent.

(c) Normalized cumulative team reward with and without a malicious agent.

Fig. 14. Normalized reward convergence using the combined immediate reward including Manhattan, Euclidean, 5-norm, and Chebyshev immediate reward functions during 200 episodes for $N = 5$ agents.



(a) Normalized average cumulative reward without malicious agents.

(b) Normalized average cumulative reward with a malicious agent (Agent#1).

Fig. 15. Normalized average cumulative reward for each agent and a team of agents, including $N = 5$ agents, using various immediate reward functions during 200 episodes.

Nevertheless, this time difference would not mean that the proposed immediate rewards are better or worse. Still, with longer episodes having more time-steps, this execution time difference may be significant. Note that the data of Table V are rounded to two decimal places.

V. Conclusions

We studied the consensus problem of a leaderless, homogeneous MARL system using the actor-critic algorithms in the absence and presence of malicious agents. Each agent's principal goal is to reach the position consensus with the maximum cumulative reward. We presented the immediate reward function based on Manhattan distance. Then, we proposed three other immediate reward functions based on various distance metrics to improve the MARL system's performance. We combined various immediate reward functions and used each of them based on the maximum returned value during each episode to enhance agents' cumulative reward in the presence of malicious agents within the MARL system. Finally, we compared different immediate reward functions within the MARL system and we found that the type of immediate reward

TABLE V
Comparing the results of algorithm's complexity and execution time using different immediate reward functions.

	Manhattan Reward	**Euclidean Reward**	**5-norm Reward**	**Chebyshev Reward**	**Combined Reward**
Algorithm Execution Time(seconds)	31.41 ±0.02	55.27 ±0.03	40.07 ±0.42	33.98 ±0.20	98.67 ±0.45
Algorithm Execution Time(seconds) (Normalized)	31.23 ±0.13	54.57 ±0.04	40.18 ±0.01	33.72 ±0.09	92.22 ±0.19
Algorithm Time Complexity	$O(n)$	$O(n)$	$O(n)$	$O(n)$	$O(n)$

function plays a significant role in efficiency of each agent in the network in reaching the consensus and obtaining further cumulative team reward.

Future work will include improved immediate reward functions such that the malicious agent's cumulative reward is reduced. In contrast, the cumulative team reward increases simultaneously. Moreover, we will study the heterogeneous MARL system in the team organization using the proposed immediate reward functions.

REFERENCES

[1] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton *et al.*, "Mastering the game of go without human knowledge," *nature*, vol. 550, no. 7676, pp. 354–359, 2017.

[2] M. AlQuraishi, "Alphafold at casp13," *Bioinformatics*, vol. 35, no. 22, pp. 4862–4865, 2019.

[3] T. T. Nguyen, N. D. Nguyen, and S. Nahavandi, "Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications," *IEEE Transactions on Cybernetics*, 2020.

[4] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[5] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[6] D. Zhang, X. Han, and C. Deng, "Review on the research and practice of deep learning and reinforcement learning in smart grids," *CSEE Journal of Power and Energy Systems*, vol. 4, no. 3, pp. 362–370, 2018.

[7] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Processing Magazine*, vol. 34, no. 6, pp. 26–38, 2017.

[8] A. Tampuu, T. Matiisen, D. Kodelja, I. Kuzovkin, K. Korjus, J. Aru, J. Aru, and R. Vicente, "Multiagent cooperation and competition with deep reinforcement learning," *PloS one*, vol. 12, no. 4, p. e0172395, 2017.

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

[10] R. Urena, F. Chiclana, G. Melancon, and E. Herrera-Viedma, "A social network based approach for consensus achievement in multiperson decision making," *Information Fusion*, vol. 47, pp. 72–87, 2019.

[11] R. Urena, G. Kou, Y. Dong, F. Chiclana, and E. Herrera-Viedma, "A review on trust propagation and opinion dynamics in social networks and group decision making frameworks," *Information Sciences*, vol. 478, pp. 461–475, 2019.

[12] G. De Pasquale and M. E. Valcher, "Consensus for clusters of agents with cooperative and antagonistic relationships," *Automatica*, vol. 135, p. 110002, 2022.

[13] D. Shen, C. Zhang, and J.-X. Xu, "Distributed learning consensus control based on neural networks for heterogeneous nonlinear multiagent systems," *International Journal of Robust and Nonlinear Control*, vol. 29, no. 13, pp. 4328–4347, 2019.

[14] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT Press, 2018.

[15] M. Figura, K. C. Kosaraju, and V. Gupta, "Adversarial attacks in consensus-based multi-agent reinforcement learning," *arXiv preprint arXiv:2103.06967*, 2021.

[16] A. Wang, T. Dong, and X. Liao, "Distributed optimal consensus algorithms in multi-agent systems," *Neurocomputing*, vol. 339, pp. 26–35, 2019.

[17] B. Mu and Y. Shi, "Distributed lqr consensus control for heterogeneous multiagent systems: Theory and experiments," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 1, pp. 434–443, 2018.

[18] Z. Wang, J. Xu, X. Song, and H. Zhang, "Consensus problem in multi-agent systems under delayed information," *Neurocomputing*, vol. 316, pp. 277–283, 2018.

[19] Y. Wang, Y. Song, D. J. Hill, and M. Krstic, "Prescribed-time consensus and containment control of networked multiagent systems," *IEEE Transactions on Cybernetics*, vol. 49, no. 4, pp. 1138–1147, 2018.

[20] B. Wang, W. Chen, and B. Zhang, "Semi-global robust tracking consensus for multi-agent uncertain systems with input saturation via metamorphic low-gain feedback," *Automatica*, vol. 103, pp. 363–373, 2019.

[21] L. Liu, H. Sun, L. Ma, J. Zhang, and Y. Bo, "Quasi-consensus control for a class of time-varying stochastic nonlinear time-delay multiagent systems subject to deception attacks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 11, pp. 6863–6873, 2020.

[22] F. Shamsi, H. A. Talebi, and F. Abdollahi, "Output consensus control of multi-agent systems with nonlinear non-minimum phase dynamics," *International Journal of Control*, vol. 91, no. 4, pp. 785–796, 2018.

[23] J. Zhang, H. Zhang, and T. Feng, "Distributed optimal consensus control for nonlinear multiagent system with unknown dynamic," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 8, pp. 3339–3348, 2017.

[24] L. Zha, J. Liu, and J. Cao, "Resilient event-triggered consensus control for nonlinear muti-agent systems with dos attacks," *Journal of the Franklin Institute*, vol. 356, no. 13, pp. 7071–7090, 2019.

[25] G. Cui, S. Xu, Q. Ma, Y. Li, and Z. Zhang, "Prescribed performance distributed consensus control for nonlinear multi-agent systems with unknown dead-zone input," *International Journal of Control*, vol. 91, no. 5, pp. 1053–1065, 2018.

[26] C. Gao, Z. Wang, X. He, and Q.-L. Han, "Consensus control of linear multiagent systems under actuator imperfection: When saturation meets fault," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2021.

[27] Z. Peng, J. Hu, K. Shi, R. Luo, R. Huang, B. K. Ghosh, and J. Huang, "A novel optimal bipartite consensus control scheme for unknown multi-agent systems via model-free reinforcement learning," *Applied Mathematics and Computation*, vol. 369, p. 124821, 2020.

[28] R. Moghadam and H. Modares, "Resilient adaptive optimal control of distributed multi-agent systems using reinforcement learning," *IET Control Theory & Applications*, vol. 12, no. 16, pp. 2165–2174, 2018.

[29] K. Zhang, Z. Yang, H. Liu, T. Zhang, and T. Başar, "Fully decentralized multi-agent reinforcement learning with networked agents," *arXiv preprint arXiv:1802.08757v2*, 2018.

[30] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," *arXiv preprint arXiv:1706.02275v4*, 2020.

[31] S. Iqbal and F. Sha, "Actor-attention-critic for multi-agent reinforcement learning," *arXiv preprint arXiv:1810.02912v2*, 2019.

[32] J. Foerster, G. Farquhar, T. Afouras, N. Nardelli, and S. Whiteson, "Counterfactual multi-agent policy gradients," *arXiv preprint arXiv:1705.08926v2*, 2017.

[33] A. Nair, V. Pong, M. Dalal, S. Bahl, S. Lin, and S. Levine, "Visual reinforcement learning with imagined goals," *arXiv preprint arXiv:1807.04742v2*, 2018.

[34] Q.-K. Hu and Y.-P. Zhao, "Aero-engine acceleration control using deep reinforcement learning with phase-based reward function," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, p. 09544100211046225, 2021.

[35] S. Zhou, Z. Hu, W. Gu, M. Jiang, M. Chen, Q. Hong, and C. Booth, "Combined heat and power system intelligent economic dispatch: A deep reinforcement learning approach," *International Journal of Electrical Power & Energy Systems*, vol. 120, p. 106016, 2020.

[36] K. C. Kosaraju, M. Figura, and V. Gupta, "Adversarial multi-agent reinforcement learning (adv-marl)," https://github.com/asokraju/adv-marl, 2020.