Proceedings of the 2023 International Conference on Machine Learning and Cybernetics, 9-11 July, 2023

HIDING IN PLAIN SIGHT: DIFFERENTIAL PRIVACY NOISE EXPLOITATION FOR EVASION-RESILIENT LOCALIZED POISONING ATTACKS IN MULTIAGENT REINFORCEMENT LEARNING

MD TAMJID HOSSAIN, HUNG LA

Advanced Robotics and Automation (ARA) Laboratory, University of Nevada, Reno, NV, USA
E-MAIL: mdtamjidh@nevada.unr.edu, hla@unr.edu

2023 International Conference on Machine Learning and Cybernetics (ICMLC) | 979-8-3503-0378-0/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICMLC58545.2023.10327979

Abstract:

Lately, differential privacy (DP) has been introduced in cooperative multiagent reinforcement learning (CMARL) to safeguard the agents' privacy against adversarial inference during knowledge sharing. Nevertheless, we argue that the noise introduced by DP mechanisms may inadvertently give rise to a novel poisoning threat, specifically in the context of private knowledge sharing during CMARL, which remains unexplored in the literature. To address this shortcoming, we present an adaptive, privacy-exploiting, and evasion-resilient localized poisoning attack (PeLPA) that capitalizes on the inherent DP-noise to circumvent anomaly detection systems and hinder the optimal convergence of the CMARL model. We rigorously evaluate our proposed PeLPA attack in diverse environments, encompassing both non-adversarial and multiple-adversarial contexts. Our findings reveal that, in a medium-scale environment, the PeLPA attack with attacker ratios of 20% and 40% can lead to an increase in average *steps to goal* by 50.69% and 64.41%, respectively. Furthermore, under similar conditions, PeLPA can result in a 1.4x and 1.6x computational time increase in optimal reward attainment and a 1.18x and 1.38x slower convergence for attacker ratios of 20% and 40%, respectively.

Keywords:

Differential Privacy; Adversarial Learning; Poisoning Attacks; Cooperative Multiagent Reinforcement Learning

1. Introduction

Cooperative multiagent reinforcement learning (CMARL) has been acknowledged for its proficiency in orchestrating complex tasks, such as automated robotic swarming and distributed power system optimization, through multi-agent collaboration [1–3]. However, the inherent nature of data sharing in CMARL can trigger potential privacy infringements, as the shared experiences often encompass sensitive data [4, 5]. To combat this, differential privacy (DP) mechanisms [6], which employ stochastic noise addition to obfuscate sensitive data, are posited as effective countermeasures [4, 7–9].

Yet, we conjecture that adversaries could exploit DP's noise-adding mechanism to craft their own malicious noise in CMARL, thereby degrading the learning efficacy while remaining undetected by hiding behind the DP-noise, leading to catastrophic implications in sectors like robotics, cyber-physical systems, automotive industries, etc. [10–13]. For example, false advising with DP-exploited misleading knowledge from advisor cars in autonomous driving may make lane-changing ambiguous and lead to severe road accidents. Contemporary state-of-the-art (SOTA) poisoning attacks typically focus on voluminous malicious data injection, which is prone to detection, leaving the creation of subtle, stealthy adversarial instances as a formidable challenge [14–17].

Addressing this challenge, our research proposes a novel adversarial model tailored for CMARL that exploits DP-induced noise to facilitate stealthy, localized poisoning attacks [18–20]. To our knowledge, this is the first investigation into DP-noise exploitation for conducting local poisoning attacks while evading detection in CMARL. Our contributions are:

- Uncovering the susceptibility of DP mechanisms to adversarial poisoning attacks, illustrating how adversaries can adaptively perturb knowledge to remain undetected.
- Proposing a novel privacy-exploiting local poisoning attack (PeLPA), contrasting general poisoning attacks that overlook the importance of attack stealthiness.
- Experimentally evaluating the potential ramifications of DP-exploited stealthy attacks in safety-critical sectors.

979-8-3503-0378-0/23/$31.00 ©2023 IEEE

The terms knowledge', experience', advice', and Q-value' are used interchangeably throughout the paper.

2. Related Works

In this section, we address the SOTA poisoning techniques.

2.1. Application of Differential Privacy for Knowledge Sharing

DP, a prominent method for privacy preservation, has been extensively employed in private knowledge sharing within the realm of CMARL [4, 7–9, 21, 22]. The scope of its application includes DP-guided Q-learning models to maintain the privacy of reward data [21], privacy-centric multi-agent frameworks leveraging federated learning (FL) and DP to obstruct illegitimate access to data statistics [4], and harnessing (β, ϕ)-DP to counteract offloading preference inference attacks in vehicular ad-hoc networks (VANET) [22]. Apart from protecting user information during private knowledge sharing, DP has also been proposed for differential advising. In particular, Ye et al. [7] propose a DP-based advising method for CMARL that enables agents to use the advice in a state even if the advice is created in a slightly different state. *Nevertheless, they overlook the susceptibility of DP to poisoning attacks during knowledge sharing [7–9].*

2.2. Poisoning Attacks in Cooperative Multiagent Learning

The infiltration of poisoning attacks in CMARL, which can alter training datasets and consequently disrupt learning outcomes, is a pertinent research concern [15,16,23,24]. Research has delved into scenarios where adversarial agents can manipulate network-wide policies [23], scrutinized targeted poisoning attacks in dual-agent frameworks where one agent's policy is modified [16], and investigated the implications of soft actor-critic algorithms in CMARL for executing poisoning attacks [24]. For instance, Figura et al. [23] demonstrate that an adversarial agent can persuade all other agents in the network to implement policies that optimize its desired objective. Another approach for performing poisoning attacks by any malicious advisor in multiagent Q-learning as demonstrated in [25], is to shuffle the Q-values for all actions corresponding to the requested state and inject false noise that is similar to the maximum reward using reward poisoning method. *However, the ramifications of these SOTA poisoning techniques against anomaly detection and privacy-preserving knowledge-sharing technologies remain largely unexplored. Our work endeavors to model a DP noise-exploiting poisoning attack that remains resilient to detection algorithms.*

2.3. Differential Privacy Exploitation Techniques

Another domain of interest focuses on the possible exploitation of DP in classification challenges, even though it does not necessarily concentrate on adversarial onslaughts on CMARL algorithms [14, 17–20, 26, 27]. This research trajectory involves the systemic degradation of utility by exploiting DP noise [19], gauging the impact of DP manipulation in smart grid networks [27], and designing stealthy model poisoning attacks on an FL model [18, 20]. Similarly, [27] investigates the impact of DP exploitation in a smart grid network and introduces a correlation among DP parameters to enable the system designer to calibrate the privacy level and reduce the attack surface. To examine the effect of DP-exploiting attacks on an FL model, [18] proposes a stealthy model poisoning attack leveraging DP noise added to ensure privacy. They improve their attack technique in [20], investigating how the degree of model poisoning can be adjusted dynamically through episodic loss memorization in FL and demonstrating how their attack can evade some SOTA defense techniques, such as norm, accuracy, and mix detection. *However, these attack models face constraints in multi-agent environments or decentralized CMARL platforms.* Contrarily, Cao et al. [14] propose an attack on the Local Differential Privacy (LDP) protocol by introducing fraudulent users. *Our research, however, targets legitimate yet compromised users infusing false noise into shared data, also aiming to dodge anomaly detectors - a critical objective for a successful attack.*

3. Local Differentially Private Cooperative Multiagent Reinforcement Learning

We present a local differentially private CMARL (LDP-CMARL) framework akin to the one adopted in [25]. However, for demonstration simplicity, instead of a generalized randomized response (GRR) technique, we leverage a Bounded Laplace (BLP) mechanism [28] to model our LDP framework that also achieves the same ε-LDP guarantee.

3.1. Cooperative Multiagent Reinforcement Learning (CMARL)

Environment model. Our research formalizes a cooperative reinforcement learning context with a Markov game $\mathcal{M} = (N, S, A, \Phi, \Gamma, T)$ incorporating N robots navigating an environment $\mathbb{E}$ of dimensions height (H) and width (W) towards

a goal G. It introduces obstacles O and freeway F with corresponding reward penalties and incentives, ϕ_O and ϕ_F. Dynamic obstacle positioning adds complexity to learning, which concludes when the first agent reaches G.

Learning objectives. Agent p_i's objective is to take the fewest steps, Π to reach G, collect ϕ_f, avoid hitting $o_x \in O$, and earn as much as rewards, $\phi_{F,G}$. In short, the objectives can be formalized as

$$\begin{aligned} &(a)\ \Pi p_i = \min_{\mathcal{M}} \Pi \\ &(b)\ \phi_{p_i} = \phi_F + \phi_G + [\phi_O = 0]\ \forall \phi_{G,F,O} \in \Phi \text{ and } \phi_G > \phi_F \\ &(c)\ \|(x_{p_i}, y_{p_i}) - (x_G, y_G)\| = 0 \end{aligned} \tag{1}$$

where (x_{p_i}, y_{p_i}), and (x_G, y_G) are p_i's and G's positions.

3.2. Integrating Local Differential Privacy (LDP) in CMARL

LDP protocols encapsulate two main stages: perturbation and aggregation. The Q-values domain, denoted as $\mathbb{Q} = [q]$, undergoes local perturbation before being relayed to the advisee, p_i, ensuring p_i's inability to infer the original Q-value of the advisor, p_k. The aggregation phase facilitates p_i's estimation of optimal advice utilizing the perturbed values received from all p_k, with perturbation function for Q-values of all actions, a in state s represented as $P(Q(s))$. Following the definition of ε-LDP [14], a protocol achieving LDP must ensure the probabilistic resemblance between any pair of perturbed Q-values.

LDP offers plausible deniability to p_k, restraining p_i from determining the origin of the output confidently. This ambiguity is regulated by the privacy budget, ε [6]. To actualize $(\varepsilon, 0)$-DP, the Laplace mechanism, a noise-addition technique, is applied as follows [6]:

$$M(D) = f(D) + \eta \sim \mathcal{N}(0, b) \tag{2}$$

where the added noise, η is drawn from a zero-mean Laplace distribution with scale parameter, $b \geq \frac{\Delta}{\varepsilon}$. Here, Δ denotes the sensitivity of the query function. Nonetheless, the same Laplace mechanism that satisfies $(\varepsilon, 0)$-DP, can be deployed in a distributed fashion for achieving ε-LDP [28, 29], by integrating randomized Laplace noise into each state-action pair's Q-values of an advisor. We leverage the higher noise sensitivity offered by the Laplace mechanism to attain stronger privacy protection as compared to Gaussian or Exponential mechanism. The advisee, p_i computes the average value from all the noisy Q-values [29]. We utilize the following BLP technique for input perturbation [28]:

Definition 1 (Bounded Laplace Mechanism (BLP)) *Given an input $q \in [l, u] \subset \mathbb{R}$, and scale $b > 0$, the BLP technique, $M : \Omega \rightarrow [l, u]$ over output $\bar{q}$ can be represented by the following conditional probability density function (pdf):*

$$fM(\bar{q}) = \begin{cases} 0 & \text{if } \bar{q} \notin [l, u] \\ \frac{1}{C_q} \frac{1}{2b} e^{-\frac{|\bar{q} - q|}{b}} & \text{if } \bar{q} \in [l, u] \end{cases} \tag{3}$$

where l and u are the lower and upper range, and $C_q = \int_l^u \frac{1}{2b} e^{-\frac{|\bar{q} - q|}{b}} d\bar{q}$ is a normalization constant. The proof and further details can be found in [28]. BLP constrains noise sampling within a predefined range, avoiding values that may detriment learning performance. Hence, the sensitivity of the combined LDP mechanism is $\Delta = |u - l|$. Similar to [7], within our LDP-CMARL framework, the sensitivity Δ needs to be calculated carefully. The LDP-CMARL framework training stages utilizing the BLP mechanism are outlined in Algorithm 1. During advice request dispatch, p_i specifies a neighbor zone, Z, and sends advice requests only to advisors within Z. Both p_i and p_k calculate their advice requesting (ϱ_{p_i}) and advice giving (ϱ_{p_k}) probabilities as per [7]. After receiving advice from the neighbors, p_i aggregates all the advice following a weighted linear aggregation technique, controlled by a predefined weight parameter, w [25]. Then, p_i selects and executes an optimal action followed by a final Q-table update.

4. Privacy Exploited Localized Poisoning Attack

In this section, we dissect the DP noise exploitation mechanism, formulating adversarial noise profile challenges. We also articulate our threat model and proposed PeLPA algorithm.

4.1. How can LDP-noise be Exploited for Poisoning Attacks?

DP not included. Considering a non-LDP advising scenario, the agents exchange Q-value knowledge, facilitating learning. We formulate the knowledge as Q-values instead of the recommended actions since the Q-value advising, unlike the action advising, does not impair the performance of the agent's learning directly [30]. Let us assume an anomaly detector at p_i's end that monitors Q-values sequences from advisor agents for all actions in a specific state, s. Generally, for a received Q-value, $Q_{p_k}(s)$, from advisor p_k, the condition $|Q_{p_k}(s) - Q_0(s)| \leq \tau$ is consistently maintained, where τ is a detection threshold and $Q_0(s)$, a historical standard Q-value. Any deviation raises an alarm, implying a potential malicious advisor $p_a \in [p_k]$ with biased Q-values. Nonetheless, to evade

Algorithm 1: LDP-CMARL Framework

Input : $\mathbb{E}, N, A, S, \Phi \rightarrow (l, u)$
Output: Trained LDP-CMARL model

Initialize Q-table, set $\varepsilon, \alpha, \Gamma$, and compute $b = \frac{\alpha|u-l|}{\varepsilon}$
for *each agent,* $p_i \in N$ **do**
  for *each episode* **do**
    Initialize state, s **for** *each state* **do**
      Send advice request to p_k in Z with ϱ_{p_i}
      Receive LDP-advice,
      $\texttt{LDP}(s, \varepsilon, b) \rightarrow \bar{Q}_{p_i}(s) = \left[\bar{Q}_i(s)\right]_{i=1}^{k}$
      for *each action* $a \in A_i$ *in state,* s **do**
        Find weighted Q-value, $Q^*_{p_i}(s,a) = w \cdot Q_{p_i}(s,a) + (1-w)(\frac{1}{k}\sum_{i=1}^{k}\bar{Q}_i(s,a))$
        Append $Q^*_{p_i}(s,a)$ to $Q^*_{p_i}(s)$
      Update Q-table with $Q^*_{p_i}(s)$
      Choose $a^* \in A_i$ for s using ϵ-greedy policy
      Execute action, a^*, observe ϕ_{p_i}, s'
      Perform $Q_{p_i}(s,a) \leftarrow (1-\alpha)Q_{p_i}(s,a) + \alpha\left[\phi_{p_i} + \Gamma \max_{a'} Q(s',a')\right]$
      Set, $s \leftarrow s'$
    If $\|(x_{p_i}, y_{p_i}) - (x_{G,y_G}) > 0\|$ **then** continue
    else end episode and reset $\mathbb{E}$
return *Trained LDP-CMARL model*
Function `LDP` (s, ε, b)**:**
  for $i = 1, 2, ..., k$ *advisors* **do**
    Receive advice request for the state, s
    With ϱ_{p_k}, **for** *each action* $a \in A_i$ **do**
      find $Q_i(s,a)$ and generate $\eta_i \sim \mathcal{N}(0, b)$
      Add LDP-noise, $\bar{Q}_i(s,a) = Q_i(s,a) + \eta_i$
      if $\bar{Q}_i(s,a) \notin (l,u)$ **then**
        Repeat loop until $\bar{Q}_i(s,a) \in (l,u)$
      else
        Append $\bar{Q}_i(s,a)$ to $\bar{Q}_i(s)$
    return $\bar{Q}_i(s)$
  return $\left[\bar{Q}_i(s)\right]_{i=1}^{k}$

detection, the attacker can introduce a bias up to a maximum of τ relative to the standard value, i.e., $Q_{p_a}(s) \leq Q_0(s) + \tau$.

DP included. With an LDP mechanism safeguarding knowledge exchange, any received Q-value, $\bar{Q}p_k(s) = Qp_k(s) + \eta$, includes noise, η following a zero-mean Laplace distribution, $\mathcal{N}(0, b)$, where b is the distribution scale. To prevent false-positive alarms for benign differentially private Q-values, the detector adjusts the previous detection condition to $|\bar{Q}p_k(s) - Q_0(s)| \leq \tau'$ with $\tau' = \tau \times \kappa; \forall \kappa \in \mathbb{R}$, where κ is the tolerance multiplier. This adjustment creates a poisoning window of $|\tau(1-\kappa)|$ that an attacker can exploit, enabling a larger bias in knowledge (i.e., Q-values) without detection. Formally, the attacker shares malicious knowledge, $\bar{Q}p_a(s) = Qp_a(s) + \eta_a; \forall \eta_a \in |\tau(1-\kappa)|$, where η_a denotes the malicious noise drawn from an adversarial noise profile, $\mathcal{N}_a$. Hence, an increase in noise for privacy enhancement also expands the detection and the poisoning window.

4.2. Challenges in Formulating Adversarial Noise Profile

Crafting an adversarial noise profile, η_a, that optimizes attack gain while evading anomaly detection poses a technical conundrum. A previous methodology [15] attempted this by maximizing utility degradation, although this leads to a paradoxical situation in the face of an anomaly detector - more noise aids detection but less noise diminishes the attack gain. A sophisticated alternative, as proposed by [19], models this as a multi-objective optimization problem, i.e., $\max_{\mathcal{A}} G(\mathcal{A}, \mathcal{D}) \ni |\bar{Q}_{p_a}(s) - Q_0(s)| \leq \tau'$ where $\mathcal{A}, \mathcal{D}$, and G denote the attack, the detect, and the gain function, respectively. The solution of this multi-criteria optimization problem is derived in [19], where the authors presented an attack impact, μ^*_a, and an optimal adversarial distribution, $\mathcal{N}^*_a(\mu^*_a, b)$ having the probability density function, f^*_a as

$$f^*_a(x) = \frac{k^2 - b^2}{2bc^2} e^{-\frac{|x-\theta|}{b} + \frac{(x-\theta)}{c}} \text{ and } \mu^*_a = \frac{b^2(\theta - 2c) - \theta c^2}{b^2 - c^2} \tag{4}$$

where θ is the mean, b^2 is the variance, and c is the Lagrange multiplier. c can be solved numerically from [19]:

$$\frac{2b^2}{c^2 - b^2} + \ln\left(1 - \frac{b^2}{c^2}\right) = \gamma. \tag{5}$$

Here, γ is the degree of knowledge poisoning; a high γ implies a large malicious noise injection (i.e., a higher attack gain) and vice versa. In particular, choosing a high γ can lead to unrealistically large Q-values whereas choosing a minuscule γ can result in negligible to almost zero attack gain. Consequently, tuning γ for an optimal attack is non-trivial but challenging, which, unfortunately, overlooked by literature so far. We address this research gap in section 4.4. Figure 1(a) demonstrates the influence of κ and γ on detected outliers and RMSE. By adding LDP-noise to 100 uniform random values, non-DP Q-values detect a steady number of outliers for a fixed τ, whereas LDP implementation significantly increases outlier detection

due to benign DP Q-values flagged as false positives. This can be mitigated by setting $\tau' = \tau \times \kappa$. Moreover, an optimal attack approach as per (4) allows successful detection evasion, maintaining the baseline outlier count while inflating the system's RMSE, as shown in Fig. 1(b).

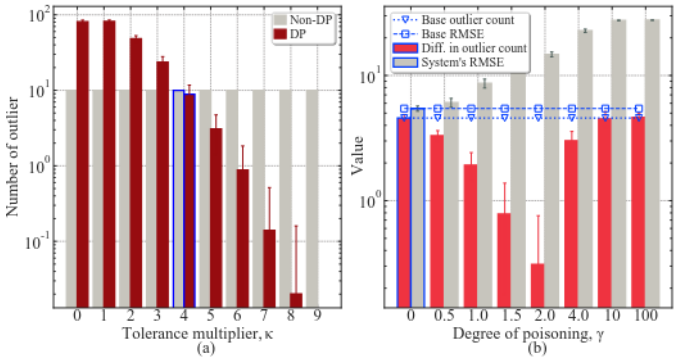

FIGURE 1. (a) Impact of tolerance multiplier, κ over detected outliers in both non-DP and DP settings, (b) Impact of degree of knowledge poisoning, γ over attack evasion (difference in outlier count between non-attack and attack scenario) and attack gain (System's RMSE).

4.3. Attacker's Capability and Knowledge

We contemplate an attacker manipulating knowledge submissions to an advisee, either by exploiting susceptible agents (internal threats) or by compromising communication channels (external threats) (Fig. 2a, b). The attacker, in line with SOTA research [31], is presumed to know the publicly available ε-value and noise distribution.

4.4. Proposed PeLPA Algorithm

A malevolent advisor, $p_a \in N$, could disrupt p_i's convergence by transmitting erroneous information during the knowledge-sharing phase. Having knowledge of $A, S, \Phi, (x_G, y_G)$ and p_i's state, s, p_a might manipulate larger Q-values for a misleading action a_m versus an ideal action a_h. This would steer p_i towards a malicious point. Yet, anomalous Q-values could either invite detection or result in an insignificant attack impact. The optimal attack method in section 4.2 addresses this trade-off. Our proposed PeLPA attack for LDP-CMARL is detailed in Algorithm 2. p_a continually injects adversarial noises (η_a) to its Q-values ($Q_a(s,a)$) until either the malicious Q-values drop below p_i's maximum Q-value for an action a, or γ exceeds a predetermined poisoning threshold

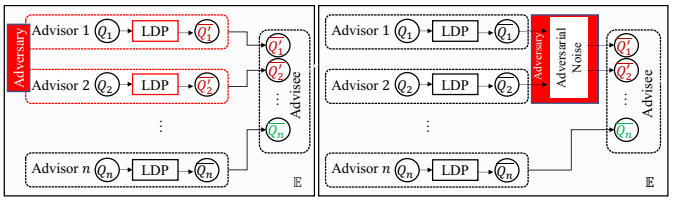

FIGURE 2. (a) Internal poisoning: Attacker compromises advisors and replaces benign LDP process with adversarial LDP process, (b) External poisoning: Attacker compromises the communication path and injects additional malicious noise.

(τ_γ). Additionally, p_a ensures malicious advice stays within the reward range $\bar{Q}_{p_a}(s,a) \in [l,u]$ to evade detection.

Algorithm 2: Proposed PeLPA Algorithm

Input : $\varepsilon, b, \alpha, Q_{p_i}(s), Q_{p_a}(s)$

Initialize $\bar{Q}_{p_a}(s) = [\,]$ and set $\gamma \leftarrow 0, \Psi \leftarrow True, \theta \leftarrow 0$

while Ψ *is True* **do**

  $\gamma = \gamma + 1$

  With b and γ, find c numerically from (5)

  Then, with c, θ and b, find $\mu^*_{p_a}$ from (4)

  for *each* $a \in A_i$ *in state,* s **do**

    while $\bar{Q}_{p_a}(s,a) \notin (l,u)$ **do**

      $\bar{Q}_{p_a}(s,a) = Q_{p_a}(s,a) + \eta_a \sim \mathcal{N}(\mu^*_{p_a}, b)$

    Append $\bar{Q}_{p_a}(s,a)$ to $\bar{Q}_{p_a}(s)$

  $\bar{Q}^*_{p_a}(s) =$

$$\begin{cases} \bar{Q}_{p_a}(s) \text{ and } \Psi \leftarrow False, & \text{if } \bar{Q}_{p_a}(s,a) < Q_{p_i}(s,a) \text{ s.t. } a \text{ for } {}_{max}Q_{p_i}(s) \text{ or } \gamma > \tau_\gamma \\ Continue, & \text{Otherwise until } \gamma \le \tau_\gamma \end{cases}$$

  Set $\bar{Q}_{p_a}(s) = [\,]$

return $\bar{Q}^*_{p_a}(s)$

5. Experimental Analysis

In this section, we implement our proposed PeLPA attack in a modified predator-prey domain, following the environmental specifications detailed in section 3.1 [3]. The environment consists of multiple predator agents and one prey. The environment is reset if the initial agent doesn't achieve the goal within a specified number of steps. Table 1 presents the experimental parameters. For comparative insight, we investigate three environment scales: *small-scale (5x5)*, *medium-scale (10x10)*, and *large-scale (15x15)*, exploring 0%, 20%, and 40% attacker

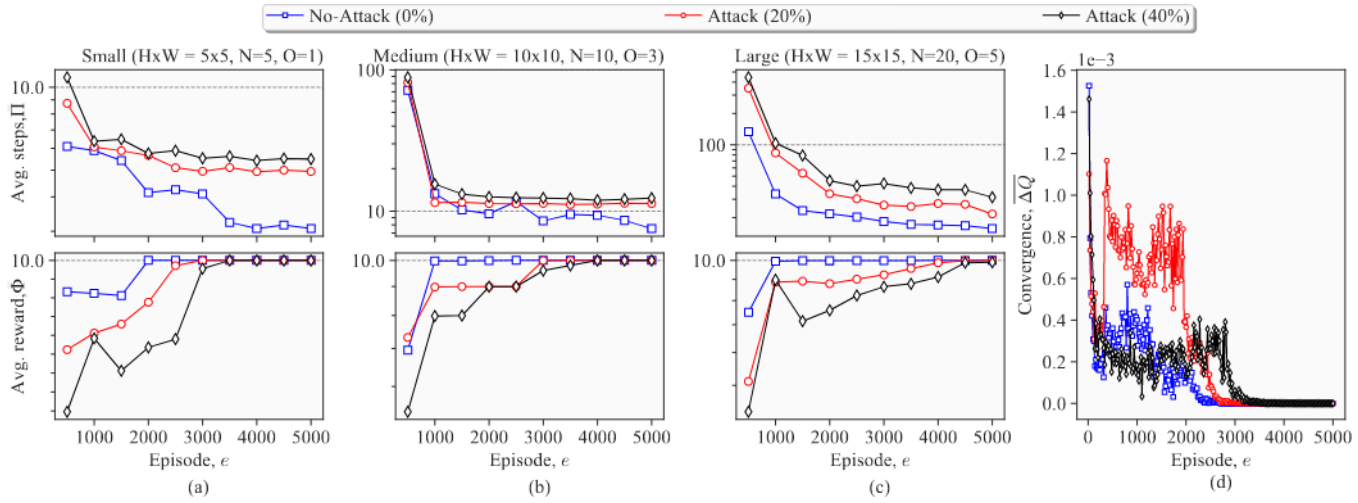

FIGURE 3. Average steps to goal ($\bar{\Pi}$) and obtained reward ($\bar{\Phi}$) analysis for (a) small ($H \times W = 5 \times 5, N = 5, O = 1$), (b) medium ($H \times W = 10 \times 10, N = 10, O = 3$), and (c) large-scale ($H \times W = 15 \times 15, N = 20, O = 5$) environments. The number of steps is increased as well as the maximum reward achievement is delayed with more attacks (large attacker ratio). Also, (d) convergence is delayed for both 20% and 40% attacks compared to the no-attack baseline.

percentages in each. Each experiment is repeated 10 times to average results. We use a privacy budget $\varepsilon = 1.0$ for all results presented, even though a smaller ε would indicate stronger privacy protection, albeit with larger attack gains.

Steps to Goal (Π) Analysis. The $\bar{\Pi}$-values represent the average steps an agent takes to achieve the goal, with lower values indicating efficient learning. The top three charts of Fig. 3(a-c) reveals an increase in the required step count to reach the goal as the attacker ratio rises and the environment expands. For example, after 5000 episodes in a medium-scale environment, $\bar{\Pi} = \{7.52, 11.332, 12.364\}$ for $\{0\%, 20\%, 40\%\}$ attackers, leading to a $\frac{(11.332-7.52)\times 100}{7.52} \approx 50.69\%$ and $\frac{(12.364-7.52)\times 100}{7.52} \approx 64.41\%$ increase in average *steps to goal* for 20% and 40% attackers, respectively.

Reward (Φ) Analysis. Similarly, the $\bar{\Phi}$-values represent average rewards obtained by agents as shown in the bottom three charts of Fig. 3(a-c). Our experiments exhibit a decrease in the speed of obtaining optimal rewards as the attacker ratio escalates. For instance, in a medium-scale environment, $\{2500, 3500, 4000\}$ episodes are requisite to attain the optimal $\bar{\Phi}$, for $\{0\%, 20\%, 40\%\}$ attackers, respectively. This leads to a $\frac{3500}{2500} \approx 1.4$x and $\frac{4000}{2500} \approx 1.6$x time increase in optimal $\bar{\Phi}$ acquisition for 20% and 40% attackers, respectively.

Convergence (ΔQ) Analysis. To gauge the effectiveness of our proposed attack, we conduct a convergence analysis based on $\overline{\Delta Q}$ values, i.e., the average of the deviation of Q-values from the optimal value (Q^*). An optimal learning process would have $\overline{\Delta Q}$ values tending to zero, and our analysis confirms this behavior is impeded as the attacker ratio increases. This delay in convergence correlates with the increase in attacker prevalence. Specifically, in a medium-scale environment, $\overline{\Delta Q}$ falls below $10e^{-6}$ following $\{2360, 2800, 3280\}$ episodes for $\{0\%, 20\%, 40\%\}$ attackers. Consequently, convergence is delayed by $\frac{2800}{2360} \approx 1.18$x and $\frac{3280}{2360} \approx 1.38$x for attacker ratios of 20% and 40%, respectively.

TABLE 1. Parameter value. α: learning rate, ϵ: exploration-exploitation probability, Γ: discount factor, B: communication budget, w: aggregation factor, τ, τ', τ_γ: predefined threshold, ϕ: reward, ε: privacy budget.

Parameter	α	ϵ	Γ	$B_{p_i}^{tot}$	$B_{p_a}^{tot}$	w	τ_γ
Value	0.10	0.08	0.80	100,000	10,000	0.90	12
Parameter	ϕ_G	ϕ_F	ϕ_O	ϕ_W	ε	τ	τ'
Value	10.0	0.50	-1.50	-0.50	1.0	100	100,000

Adaptive Degree of Knowledge Poisoning (γ). Finally, we consider the degree of knowledge poisoning, γ, demonstrating its distribution and symmetry in various scenarios as shown in Fig. 4. This parameter is adjusted following line 10 in Algorithm 2, showing varied instances of its manipulation across different episodes. We only present the episodes in which the attacker adjusted the γ value more than 20 times. For example,

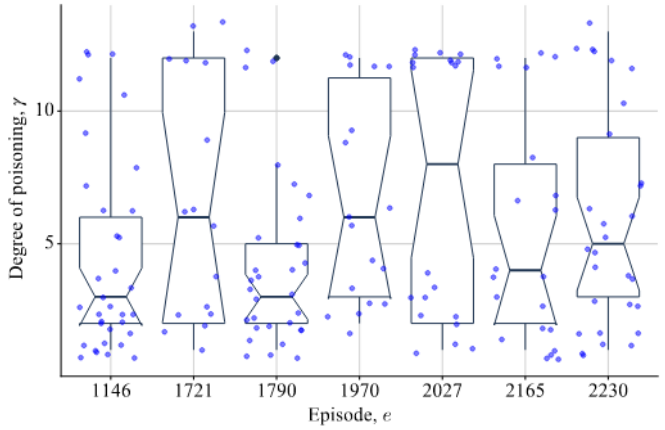

FIGURE 4. Distribution of the degree of knowledge poisoning, (γ) in some example episodes. For instance, in episode 1146, the attacker maintained the γ value under 5 for most of the steps but increased it to more than 10 for a few steps.

in episode 1146, the attacker maintained the γ value under 5 for most of the steps but increased it to more than 10 for a few steps. Contrarily, in episode 2027, the attacker never sets γ in the range of $[5, 10]$.

6. Conclusions

This paper highlights the potential security risks of using DP in CMARL algorithms and proposes a new adaptive and localized knowledge poisoning attack technique (PeLPA) to exploit DP-noise and prevent optimal convergence of the CMARL model. The proposed PeLPA technique is designed to evade SOTA anomaly detection techniques and degrade the multiagent learning performance. The effectiveness of the proposed attack technique is demonstrated through extensive experimental analysis in varying environment scales. The study fills a research gap in the literature and sheds light on the need for stronger security measures in LDP-CMARL systems.

References

[1] W. Zhou, D. Chen, J. Yan, Z. Li, H. Yin, and W. Ge, "Multi-agent reinforcement learning for cooperative lane changing of connected and autonomous vehicles in mixed traffic," *Autonomous Intelligent Systems*, vol. 2, 12 2022.

[2] F. L. Da Silva, R. Glatt, and A. H. R. Costa, "Simultaneously learning and advising in multiagent reinforcement learning," in *Proceedings of the 16th conference on autonomous agents and multiagent systems*, ser. AAMAS '17, 2017, pp. 1100–1108.

[3] H. M. Le, Y. Yue, P. Carr, and P. Lucey, "Coordinated multi-agent imitation learning," in *International Conference on Machine Learning*. PMLR, 2017, pp. 1995–2003.

[4] Q. Li, B. Guo, and Z. Wang, "A privacy-preserving multiagent updating framework for self-adaptive tree model," *Peer-to-Peer Networking and Applications*, vol. 15, no. 2, pp. 921–933, 2022.

[5] Y. Zou, Z. Zhang, M. Backes, and Y. Zhang, "Privacy analysis of deep learning in the wild: Membership inference attacks against transfer learning," *arXiv preprint arXiv:2009.04872*, 2020.

[6] C. Dwork, "Differential privacy," in *Automata, Languages and Programming*, M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 1–12.

[7] D. Ye, T. Zhu, Z. Cheng, W. Zhou, and P. S. Yu, "Differential advising in multiagent reinforcement learning," *IEEE Transactions on Cybernetics*, vol. 52, no. 6, pp. 5508–5521, 2022.

[8] S. Abahussein, T. Zhu, D. Ye, Z. Cheng, and W. Zhou, "Protect trajectory privacy in food delivery with differential privacy and multi-agent reinforcement learning," in *Advanced Information Networking and Applications*, L. Barolli, Ed. Cham: Springer International Publishing, 2023, pp. 48–59.

[9] D. Ye, S. Shen, T. Zhu, B. Liu, and W. Zhou, "One parameter defense—defending against data inference attacks via differential privacy," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 1466–1480, 2022.

[10] P. M. Scheikl, B. Gyenes, T. Davitashvili, R. Younis, A. Schulze, B. P. Müller-Stich, G. Neumann, M. Wagner, and F. Mathis-Ullrich, "Cooperative assistance in robotic surgery through multi-agent reinforcement learning," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 1859–1864.

[11] H. M. La, R. Lim, and W. Sheng, "Multirobot cooperative learning for predator avoidance," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 1, pp. 52–63, 2014.

[12] A. Prasad and I. Dusparic, "Multi-agent deep reinforcement learning for zero energy communities," in *2019 IEEE PES innovative smart grid technologies Europe (ISGT-Europe)*. IEEE, 2019, pp. 1–5.

[13] B. R. Kiran, I. Sobh, V. Talpaert, P. Mannion, A. A. A. Sallab, S. Yogamani, and P. Pérez, "Deep reinforcement learning for autonomous driving: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 4909–4926, 2022.

[14] X. Cao, J. Jia, and N. Z. Gong, "Data poisoning attacks to local differential privacy protocols," in *30th {USENIX} Security Symposium ({USENIX} Security 21)*, 2021.

[15] M. Fang, X. Cao, J. Jia, and N. Gong, "Local model poisoning attacks to byzantine-robust federated learning," in *29th {USENIX} Security Symposium ({USENIX} Security 20)*, 2020, pp. 1605–1622.

[16] M. Mohammadi, J. Nöther, D. Mandal, A. Singla, and G. Radanovic, "Implicit poisoning attacks in two-agent reinforcement learning: Adversarial policies for training-time attacks," *arXiv preprint arXiv:2302.13851*, 2023.

[17] A. Cheu, A. Smith, and J. Ullman, "Manipulation attacks in local differential privacy," in *2021 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2021, pp. 883–900.

[18] M. T. Hossain, S. Islam, S. Badsha, and H. Shen, "Desmp: Differential privacy-exploited stealthy model poisoning attacks in federated learning," in *2021 17th International Conference on Mobility, Sensing and Networking (MSN)*, 2021, pp. 167–174.

[19] J. Giraldo, A. Cardenas, M. Kantarcioglu, and J. Katz, "Adversarial classification under differential privacy," in *Network and Distributed Systems Security (NDSS) Symposium 2020*, 2020.

[20] M. T. Hossain, S. Badsha, H. La, H. Shen, S. Islam, I. Khalil, and X. Yi, "Adversarial analysis of the differentially-private federated learning in cyber-physical critical infrastructures," 2022.

[21] B. Wang and N. Hegde, "Privacy-preserving q-learning with functional noise in continuous spaces," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[22] D. Wei, J. Zhang, M. Shojafar, S. Kumari, N. Xi, and J. Ma, "Privacy-aware multiagent deep reinforcement learning for task offloading in vanet," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–15, 2022.

[23] M. Figura, K. C. Kosaraju, and V. Gupta, "Adversarial attacks in consensus-based multi-agent reinforcement learning," in *2021 American Control Conference (ACC)*. IEEE, 2021, pp. 3050–3055.

[24] Z. Xie, Y. Xiang, Y. Li, S. Zhao, E. Tong, W. Niu, J. Liu, and J. Wang, "Security analysis of poisoning attacks against multi-agent reinforcement learning," in *Algorithms and Architectures for Parallel Processing*, Y. Lai, T. Wang, M. Jiang, G. Xu, W. Liang, and A. Castiglione, Eds. Cham: Springer International Publishing, 2022, pp. 660–675.

[25] M. T. Hossain, H. M. La, and S. Badsha, "Brnes: Enabling security and privacy-aware experience sharing in multiagent robotic and autonomous systems [manuscript submitted for publication]," *Department of Computer Science and Engineering, University of Nevada, Reno*, 2023.

[26] J. Giraldo, A. A. Cardenas, and M. Kantarcioglu, "Security vs. privacy: How integrity attacks can be masked by the noise of differential privacy," in *2017 American Control Conference (ACC)*. IEEE, 2017, pp. 1679–1684.

[27] M. T. Hossain, S. Badsha, and H. Shen, "Privacy, security, and utility analysis of differentially private cpes data," in *2021 IEEE Conference on Communications and Network Security (CNS)*. IEEE, 2021, pp. 65–73.

[28] J. Neera, X. Chen, N. Aslam, K. Wang, and Z. Shu, "Private and utility enhanced recommendations with local differential privacy and gaussian mixture model," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 4, pp. 4151–4163, 2023.

[29] T. Wang, X. Zhang, J. Feng, and X. Yang, "A comprehensive survey on local differential privacy toward data statistics and analysis," *Sensors*, vol. 20, no. 24, p. 7030, 2020.

[30] C. Zhu, H.-F. Leung, S. Hu, and Y. Cai, "A q-values sharing framework for multi-agent reinforcement learning under budget constraint," *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, vol. 15, no. 2, pp. 1–28, 2021.

[31] C. Dwork, N. Kohli, and D. Mulligan, "Differential privacy in practice: Expose your epsilons!" *Journal of Privacy and Confidentiality*, vol. 9, no. 2, 2019.