Data Poisoning Attacks and Defenses to LDP-Based Privacy-Preserving Crowdsensing

Zhirun Zheng, Zhetao Li, *Member, IEEE*, Cheng Huang, *Member, IEEE*, Saiqin Long, Mushu Li, *Member, IEEE*, and Xuemin Shen, *Fellow, IEEE*

***Abstract*—In this article, we explore data poisoning attacks and their defenses in local differential privacy (LDP)-based crowdsensing systems. First, we construct data poisoning attacks launched by corrupted workers to subvert crowdsensing results by tampering information reported. Specifically, the attacks are formulated as a bi-level optimization problem where attackers strive to conceal their malicious behavior by delicately exploiting noise perturbation introduced by LDP protocols. In this way, the attacks can not be detected, even with the weight-based truth discovery methods. Due to the NP-hard nature of the bi-level problem, we decompose it into upper-level and lower-level sub-problems and employ the augmented Lagrangian method to iteratively solve them, ultimately identifying optimal attack strategies. Second, we propose corresponding countermeasures to defend against the attacks. The countermeasures are formulated as a minimization problem, with the objective of minimizing disruptions caused by attacks through the identification and removal of corrupted workers from crowdsensing systems. To solve the problem, we utilize a differential evolution algorithm instead of gradient-based methods since the objective function of the problem is not differentiable. Extensive experiments on real-world datasets are conducted to evaluate the performance of the proposed attacks and defenses. The evaluation results demonstrate that LDP perturbation indeed facilitates the success of data poisoning attacks, and the proposed defenses can accurately distinguish malicious behaviors disguised.**

***Index Terms*—Data poisoning attacks, local differential privacy, crowdsensing, truth discovery, optimization-based defenses.**

Manuscript received 25 April 2023; revised 16 January 2024; accepted 5 February 2024. Date of publication 7 February 2024; date of current version 4 September 2024. This work was supported in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada, in part by the National Natural Science Foundation of China under Grant 62032020, Grant 62172350, Grant 62076214, and Grant U23B2027, in part by the National Key Research and Development Program of China under Grant 2021YFB3101200, in part by the Postgraduate Scientific Research Innovation Project of Hunan Province under Grant CX20200618, and in part by the Postgraduate Scientific Research Innovation Project of Xiangtan University under Grant XDCX2020B085. *(Corresponding author: Zhetao Li.)*

Zhirun Zheng is with the School of Mathematics and Computational Science, Xiangtan University, Xiangtan, Hunan 411105, China (e-mail: zhengzhirun2019@gmail.com).

Zhetao Li and Saiqin Long are with the National and Local Joint Engineering Research Center of Network Security Detection and Protection Technology, Guangdong Provincial Key Laboratory of Data Security and Privacy Protection, and College of Information Science and Technology, Jinan University, Guangzhou, Guangdong 510632, China (e-mail: liztchina@hotmail.com; xxgcxyxtu@sina.com).

Cheng Huang, Mushu Li, and Xuemin Shen are with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: cheng.huang@uwaterloo.ca; m475li@uwaterloo.ca; sshen@uwaterloo.ca).

Digital Object Identifier 10.1109/TDSC.2024.3363507

I. Introduction

CROWDSENSING has emerged as a popular and widely-adopted information collection paradigm in many applications such as healthcare and transportation, wherein mobile devices (e.g., wearable devices and smart vehicles) are recruited by a crowdsensing server and act as intelligent sensors (a.k.a. workers) to collect various environmental data, including temperature and humidity [1], [2], [3], [4]. Compared to traditional approaches that rely on fixed-deployed sensors, it utilizes mobile crowd wealth to offer several advantages and provide greater flexibility and accuracy via eliminating deployment costs and covering larger sensing areas.

However, there are two main obstacles that hinder the future development and advancement of crowdsensing. First, the quality of sensory data collected from different workers with varying sensing capabilities cannot be guaranteed, resulting in unreliable and conflicting data [5]. Furthermore, malicious workers may intentionally launch data poisoning attacks to benefit themselves [6], [7], [8]. In a smart transportation system, malicious drivers can report false traffic congestions and accidents to affect other vehicles' routes and reduce the traffic on their own routes. Second, considering that sensing tasks are performed by individual mobile users, collected data will carry sensitive personal information [9], [10], [11], [12], [13], [14], [15], [16], e.g., location and health data, raising serious privacy concerns and potentially deterring worker participation.

Many state-of-the-art works [5], [17], [18], [19], [20], [21], [22], [23] have been proposed to address the two major issues. To extract (or infer) truth information from the conflicting sensory data in crowdsensing, truth discovery methods [5], [17], [18], [19] are generally adopted. Most of the existing truth discovery methods are designed based on a class of reliability-based weighted aggregation algorithms. The rationale behind the methods is straightforward, that is, the worker will obtain a lower weight when it always disagrees with the majority of workers. In other words, a lower weight means that the worker has lower reliability and its sensory data will be counted less when the crowdsensing server aggregates all collected sensory data to find the truth. By doing so, the methods can mitigate corrupted (or unreliable) workers' negative effects. Furthermore, to protect workers' sensory data privacy in crowdsensing, Local Differential Privacy (LDP) [20], [21], [22] has been widely adopted. With LDP, each worker can first perturb its sensory data locally and then upload the perturbed data to the crowdsensing server. The scale of perturbation is controlled according to a pre-defined

privacy budget [24], and a lower privacy budget leads to more perturbation and stronger privacy protection, and vice versa.

Both truth discovery method and LDP protocol are effective in crowdsensing applications to some extent. However, when jointly applying them together, LDP perturbation makes it easier for corrupted workers to conduct data poisoning attacks even if an effective truth discovery method is deployed. This is due to the fact that LDP protocols make the aggregated truth discovery result highly sensitive to small changes in the collected sensory data. As a result, the attackers can overturn the inferred truth by covertly manipulating the distribution of collected sensory data. Similar studies [25], [26], [27] have demonstrated that LDP-based privacy-preserving heavy hitters and histograms are seriously vulnerable to data poisoning attacks. These studies focus on data statistic scenarios without the truth discovery methods, i.e., all workers' reports (sensory data) will be aggregated with the same weight. In addition, related defenses only work against simple and fixed attack strategies, which have their inherent limitations in scalability. Thus, attack and defense strategies should be studied jointly.

Different from existing works, we focus on more complex data poisoning attacks against LDP-based privacy-preserving crowdsensing. The LDP perturbation does not be identified as malicious behavior because the bias it introduces is calibrated by unbiased estimation in LDP-based schemes. As a result, the attackers can easily degrade the trustworthiness of inferred truth information by exploiting LDP perturbation. To make the attack feasible, there exist two technical challenges. The first challenge is *how to hide attack behavior in the LDP perturbation?* It is impossible to know the exact distribution that LDP perturbation obeys since the LDP protocols are randomized. Thus, it is challenging for the attackers to disguise attack behavior and embed their poisoned data in the reports using LDP perturbation. The second challenge is *how can attackers maximize the attack profits?* It is noted that the attack behavior can remain stealthy if the malicious behavior and LDP perturbation are indistinguishable, but it does not degrade the trustworthiness of inferred truth information. Therefore, it is nontrivial to maximize the deviation from the outcomes of truth discovery methods while remaining stealthy, namely, maximizing the attack profits.

To address the challenges, we propose optimal data poisoning attacks against LDP-based privacy-preserving crowdsensing. Specifically, the data poisoning attacks are formulated as an optimization problem, whose objective is to deviate from the truth information as much as possible and whose constraint is an LDP-based privacy-preserving truth discovery method. Through this problem, the attackers could maximize their profits while hiding behind the LDP perturbation. Since the constraint is also an optimization problem, the optimization problem is bi-level and NP-hard. To find the optimal attack strategies, the problem is decomposed into upper-level and lower-level sub-problems, which are alternately optimized by the augmented Lagrangian method. We evaluate the proposed data poisoning attacks on the weather sentiment dataset and the Duchenne smile dataset. The simulation results demonstrate that the LDP-based privacy-preserving crowdsensing are more vulnerable to data poisoning attacks when LDP with lower privacy budgets (i.e., stronger privacy preservation) is applied. For instance, on the Duchenne smile dataset, the attack effectiveness (measured by Average Utility Damage) can be increased by 86.4% when the privacy budget is decreased from 3.0 to 1.0.

We also propose optimal countermeasures to defend against the proposed data poisoning attacks. Defenders aim to detect all corrupted workers without violating LDP protocols. *Since the attack behavior and LDP perturbation are indistinguishable, it is extremely hard for defenders to detect corrupted workers without violating LDP protocols.* To address this challenge, the countermeasures are formulated as the optimization problem, whose objective is to minimize the deviation caused by our data poisoning attacks. In other words, defenders detect the workers with the largest influence on the deviation as corrupted. To obtain the optimal defense strategy, the minimization problem is solved by Differential Evolution (DE) algorithm. The countermeasures are evaluated on two real-world datasets, and the results validate that they can effectively defend against our proposed data poisoning attacks. For example, on the Duchenne smile dataset, the defense effectiveness (measured by AUC, a classic metric for evaluating the quality of classifiers) only decreases by 23.2% when the privacy budget is reduced from 2.8 to 1.0. To summarize, our main contributions are summarized as follows:

- To explore the hidden risks posed by LDP in crowdsensing, we propose optimization-based data poisoning attacks and defenses to LDP-based privacy-preserving crowdsensing systems.
- We formulate the data poisoning attacks against the LDP-based privacy-preserving crowdsensing as the bi-level optimization problem. Due to the problem being NP-hard, it is partitioned into upper-level and lower-level sub-problems, which can be iteratively solved using the augmented Lagrangian method. Through extensive evaluation on real-world datasets, we find that LDP perturbation makes data poisoning attacks easier.
- We formulate the countermeasures to defend against the data poisoning attacks as the minimization problem, which is solved by a differential evolution algorithm. Our experimental results demonstrate that the proposed countermeasures can significantly mitigate data poisoning attacks even if attackers perform their attack using LDP perturbation.

The remainder of the paper is organized as follows. We discuss related work in Section II. In Section III, we introduce truth discovery methods and local differential privacy, which are two basic concepts used in this paper. We give the system model and problem statement in Section IV. The main contributions are present in Sections V and VI. We detail the optimal data poisoning attacks against LDP-based privacy-preserving crowdsensing in Section V. Then, we give the optimal countermeasures to defend against the proposed data poisoning attacks in Section VI. The performance of the proposed optimal data poisoning attacks and defenses are evaluated in Section VII. We conclude this paper in Section VIII.

TABLE I
RELATED WORKS ON CROWDSENSING SYSTEMS

Works	**Privacy**: Preventing the Real Sensory Data of Participating Workers Leaks	**Utility**: Trustworthiness of the Results Aggregated by the Truth Discovery Methods	**Poisoning**: Injecting Malicious (or Fake) Data into Crowdsensing Systems
[28]	✓	✓	×
	Protecting the privacy through DP (or LDP) while improving the utility.		
[8]	×	✓	✓
		Undermining the utility by stealthy injecting malicious sensory data.	
[9]	✓	✓	✓
	Disguising the poisoning as the DP-based noisy data to damage the utility while bypassing the truth discovery methods.		
Ours	✓	✓	✓
	Hide data poisoning attacks behind the LDP noise, remaining stealthy and degrading the utility as much as possible.		

II. RELATED WORK

We review state-of-the-art works related to privacy-preserving crowdsensing and data poisoning attacks and defenses to crowdsensing. These works focus on three dimensions: *Privacy* (preventing the real sensory data of participating workers leaks), *Utility* (trustworthiness of the results aggregated by the truth discovery methods), and *Poisoning* (injecting malicious (or fake) data into crowdsensing systems). As shown in Table I, previous works mainly fall into two topics: (1) privacy-preserving crowdsensing (which study privacy versus utility) and (2) data poisoning attacks and defenses to the crowdsensing without privacy protection (which study poisoning versus utility). Different from these works, we propose the data poisoning attacks and defenses in LDP-based privacy-preserving crowdsensing (which study privacy versus poisoning versus utility), exploring the hidden risks posed by LDP in crowdsensing.

A. Privacy-Preserving Crowdsensing

To protect workers' sensory data privacy, perturbation-based privacy-preserving protocols (such as DP [24], LDP [20], [21], [22], [29], and information-theoretic privacy [12], [30]) are widely deployed in crowdsensing. Because DP and LDP can provide strictly provable privacy protection, they are regarded as magic weapons to defend against various privacy attacks. However, DP and LDP gain privacy protection by sacrificing utility. To this end, existing studies about privacy-preserving crowdsensing have mainly focused on the trade-offs between sensory data privacy and the trustworthiness (or utility) of inferred truth information.

To prevent any party other than participating workers from observing the real sensory data, Li et al. [28] proposed an LDP-based one-layer privacy-preserving protocol. All sensory data is perturbed locally according to the same response probability before sharing. Li et al. [10] proposed an efficient differentially private truth discovery method for crowdsensing, in which each worker independently perturbs its sensory data. Furthermore, Xu et al. [31] considered a dishonest crowdsensing server and proposed a verifiable privacy-preserving protocol, namely V-PATD. V-PATD provides workers with privacy guarantees by LDP and enables requesters to verify that the truth discovery methods were performed correctly.

To improve the utility of inferred truth information, Li et al. [28] proposed an LDP-based two-layer privacy-preserving protocol, in which workers sample their own response probability from a hyper distribution and then perturbs its sensory data locally according to the sampled response probability. Li et al. [28] also proved that the two-layer protocol delivers better utility than the one-layer protocol when they provide the same privacy protection. Besides, personalization is widely adopted in privacy-preserving crowdsensing to reduce the scale of perturbation. For instance, Pang et al. [32] proposed a personalized privacy budget allocation mechanism according to the privacy requirements of workers. Then, based on the proposed privacy budget allocation mechanism, Pang et al. [32] designed a personalized privacy-preserving protocol for crowdsensing, namely PPPTD, that guarantees privacy while achieving high utility. To further improve the utility of inferred truth information, Sun et al. [33] proposed the personalized award mechanism for workers with different privacy preferences. In other words, workers can sacrifice privacy in exchange for more rewards. All those works fall into the topic of privacy versus utility.

B. Data Poisoning Attacks and Defenses to Crowdsensing

In recent years, the data poisoning attacks and defenses to crowdsensing without privacy protection have been widely studied [6], [7], [8], [34]. The truth discovery methods are widely deployed in crowdsensing as an anomaly detection mechanism, which makes data poisoning attacks against crowdsensing difficult to launch. To overcome this challenge, disguise-based data poisoning attacks have been proposed [6], [7], [8]. For example, Miao et al. [6] proposed an intelligent data poisoning attack mechanism against crowdsourcing with the Conflict Resolution on Heterogeneous data (CRH, a type of truth discovery methods) empowered. In this attack, the malicious behavior is more intelligently, that is, the corrupted workers try to improve their reliability degrees (i.e., worker weights) by agreeing with the normal workers on some objects whose truth information is unlikely to be overturned. However, the data poisoning attack proposed by [6] is only applicable to the CRH model and discrete data. To overcome this shortcoming, Fang et al. [7] proposed a data poisoning attack against crowdsourcing with the Dawid-Skene (a type of truth discovery methods) and Fang et al. [8] studied the data poisoning attacks and defenses to crowdsensing systems with continuous data. All of these works are focused on studying

poisoning versus utility. *However, with the increasing awareness of privacy protection, LDP is widely adopted in crowdsensing to protect workers' sensory data privacy. Thus, it is urgent to explore whether attackers can launch powerful data poisoning attacks by weaponizing LDP perturbation (which investigates the tradeoffs between privacy, poisoning, and utility).*

Li et al. [9] initially investigated the poisoning attacks against DP-based privacy-preserving crowdsensing and proposed a novel disguise-based data poisoning attack, named DDPA. DDPA attempts to evade the truth discovery methods by disguising itself as DP noise (which is added by the Laplace and Gaussian mechanisms), decreasing the trustworthiness of inferred truth information. Particularly, DDPA only applies to DP-based privacy-preserving crowdsensing with continuous data because the Laplace and Gaussian mechanisms are continuous probability distributions. *The randomized response mechanism is a discrete probability distribution, however, which is a classic LDP protocol. Besides, LDP perturbation is not taken from a specific distribution because the LDP protocols are randomized, which makes perturbation more difficult to weaponize. For these reasons, DDPA is not suitable for LDP-based privacy-preserving crowdsensing. To this end, we propose optimization-based data poisoning attacks against the LDP-based privacy-preserving crowdsensing systems and then propose corresponding countermeasures.*

III. Preliminaries

In this section, we detail some basic concepts about truth discovery methods and local differential privacy.

A. Truth Discovery Methods

Truth discovery methods [5], [18], [19] have been greatly deployed in crowdsensing to discover truth information from the conflicting and unreliable sensory data uploaded by workers. The methods can automatically estimate the worker's reliability (i.e., worker's weight) from its sensory data, and then infer reliable information (i.e., truth) from the collected sensory data power by the reliability. We note that the methods follow two general principles: workers obtain a relatively high weight if they always upload reliable sensory data, and the sensory data shared by high-weight workers is more likely to be the truth. In general, the methods mainly extract the truth from collected unreliable sensory data by weighted aggregation, which is summarized as an iterative algorithm: *Truth Inference* and *Weight Estimation* are alternately performed (as shown in Algorithm 1). Next, we detail *Truth Inference* and *Weight Estimation* respectively.

Truth Inference. This step infers the truth by conducting weighted aggregation according to the current worker weights $W = \{w_1, w_2, \ldots, w_{|U|}\}$, where the index of users is denoted by $m \in \{1, \ldots, |U|\}$. Then, the truth x_n^{trust} of each object $\forall o_n \in O$ (the index of objects is denoted by $n \in \{1, \ldots, |O|\}$) is calculated based on the following weighted aggregation:

$$x_n^{trust} = \frac{\sum_{m=1}^{|U|} w_m \cdot x_n^m}{\sum_{m=1}^{|U|} w_m}, \tag{1}$$

Algorithm 1: Truth Discovery Method.

Input : Sensory data of $|O|$ objects submitted by U workers $\{x_n^m \mid \forall o_n \in O, \forall u_m \in U\}$; Threshold for convergence $\phi > 0$

Output: Inferred truths $X^{truth} = \{x_1^{truth}, \cdots, x_{|O|}^{truth}\}$ for objects O

```
Initialize worker weights W = {w_m | ∀u_m ∈ U};
X^truth ← Calculate the truths using current weights W based on Eq. (1);
while Ture do
    W ← Calculate the weights using current truths X^truth based on Eq. (2);
    X^truth_new = {x^truth_1,new, ··· , x^truth_|O|,new} ← Calculate the truths using current weights W based on Eq. (1);
    if (1/|O|) · Σ_{n=1}^{|O|} d(x_n^truth, x_n,new^truth) ≤ φ then
        break;
    end
    X^truth ← X^truth_new;
end
```

where the truth x_n^{truth} is a continuous value and x_n^m is the sensory data uploaded by worker u_m for object o_n. We treat the integer closest to x_n^{truth} as the truth. For example, supposing x_n^{truth} equals 4.3, then it implies that the integer 4 is the truth of the object o_n.

Weight Estimation. In this step, the worker weights W are updated based on the current truths $X^{truth} = \{x_1^{truth}, x_2^{truth}, \ldots, x_{|O|}^{truth}\}$. A worker is assigned a higher weight when its sensory data is close to the aggregated result, which is formulated as follows:

$$w_m = h\left(\sum_{n=1}^{|O|} d(x_n^m, x_n^{truth})\right), \tag{2}$$

where $h(*)$ denotes a monotonically decreasing worker-weights-updating function, and $d(x_n^m, x_n^{truth})$ measures the difference between sensory data and truth.

We use a state-of-the-art truth discovery method CRH [5], [19] as a concrete example to illustrate the basic idea of the optimal data poisoning attacks and defenses to LDP-based privacy-preserving crowdsensing. CRH is a classic optimization-based truth discovery method, which is formulated as:

$$\underset{X^{truth}, W}{\text{minimize}} \sum_{m=1}^{|U|} w_m \cdot \sum_{n=1}^{|O|} d(x_n^m, x_n^{truth}), \tag{3}$$

$$\text{subject to} \sum_{m=1}^{|U|} \exp(-w_m) = 1, \ |U| > 1. \tag{3a}$$

According to the definition of $h(*)$ in CRH, the worker-weights-updating function (2) is formulated as:

$$w_m = \log\left(\frac{\sum_{m=1}^{|U|}\sum_{n=1}^{|O|} d(x_n^m, x_n^{truth})}{\sum_{n=1}^{|O|} d(x_n^m, x_n^{truth})}\right), \tag{4}$$

where distance function $d(x_n^m, x_n^{truth}) = (x_n^m - x_n^{truth})^2$.

TABLE II
NOTATIONS USED IN THE PROPOSED DATA POISONING ATTACKS AND DEFENSES

Notation	Definition	Notation	Definition
$O/\|O\|$	Set/number of all objects	o_n	n-th object, $o_n \in O$
P_D	Set of corruption probabilities	p_m	Probability that m-th worker is corrupted
$\widehat{U}/\|\widehat{U}\|$	Set/number of all normal workers	$\widehat{u}_m$	m-th normal worker, $\widehat{u}_m \in \widehat{U}$
$\widetilde{U}/\|\widetilde{U}\|$	Set/number of all corrupted workers	$\widetilde{u}_m$	m-th corrupted worker, $\widetilde{u}_m \in \widetilde{U}$
$U/\|U\|$	Set/number of all participating workers, $\widehat{U} \cup \widetilde{U} = U$	u_m	m-th participating worker, $u_m \in U$
U^{known}	Set of known workers, $U^{known} \subset \widehat{U}$	φ	Percentage of corrupted workers
$\|U^{known}\|$	Number of known workers, $\|U^{known}\| < \|\widehat{U}\|$	ψ	Percentage of known workers
$\widehat{W}/\widetilde{W}$	Set of all normal/corrupted worker weights	$\widehat{w}_m/\widetilde{w}_m$	Weight of m-th normal/corrupted worker
W	Set of all participating worker weights, $\widehat{W} \cup \widetilde{W} = W$	w_m	Weight of m-th participating worker u_m
$\widehat{X}/\widehat{Y}$	Set of real/perturbed data sensed by all normal workers	$\widehat{x}_n^m/\widehat{y}_n^m$	Real/perturbed data sensed by $\widehat{u}_m$ for o_n
$\widetilde{X}$	Set of malicious data shared by all corrupted workers	$\widetilde{x}_n^m$	Malicious data provided by $\widetilde{u}_m$ for o_n
$\widetilde{X}^m$	Set of malicious data uploaded by m-th corrupted worker	$[k]$	Set of positive integers up to k, $[k] = \{1, \cdots, k\}$
X	Set of sensory data collected by the server, $\widehat{Y} \cup \widetilde{X} = X$	x_n^m	Sensory data uploaded by w_m for o_n
X^{truth}	Set of truths inferred from the perturbed data of known workers	x_n^{truth}	Inferred truth of n-th object o_n
X_a^{truth}	Set of inferred truths after attacks or defenses	$x_{a,n}^{truth}$	Inferred truth of o_n after attacks or defenses
X_g^{truth}	Set of ground truths	$x_{g,n}^{truth}$	Ground truth data of o_n

B. Local Differential Privacy

LDP [20], [21], [22] is a privacy metric to prevent sensory data leaks in the local model. In general, a protocol satisfies LDP if any two sensory data are perturbed to the same value with close probability. We give the definition of LDP as follows.

Definition 1 (Local Differential Privacy). Let $k \in \mathbb{N}$, $[k] = \{1, 2, \ldots, k\}$. A protocol $\mathcal{A}$ satisfies ε-local differential privacy (ε-LDP), if for any pair of sensory data $x_{n,1}^m, x_{n,2}^m \in [k]$ and any perturbed sensory data $y_n^m \in [k]$, we have

$$Pr\left\{\mathcal{A}(x_{n,1}^m) = y_n^m\right\} \leq e^{\varepsilon} \times Pr\left\{\mathcal{A}(x_{n,2}^m) = y_n^m\right\}, \quad (5)$$

where $\varepsilon > 0$ is called privacy budget and $\mathcal{A}(x_n^m)$ is the random perturbed value of sensory data x_n^m.

To achieve ε-LDP, a classic *randomized response* protocol, i.e., Warner's Randomized Response (Warner's RR) [35], is widely adopted. Next, we give the definition of Warner's RR.

Definition 2 (Warner's Randomized Response [35]). Let $k \in \mathbb{N}$, $[k] = \{1, 2, \ldots, k\}$. Given sensory data $x_n^m \in [k]$, Warner's RR outputs its perturbed sensory data $y_n^m \in [k]$. The sensory data x_n^m is perturbed to y_n^m, i.e., $y_n^m = x_n^m$, with probability p, or is perturbed to any sensory data exclude x_n^m, i.e., $\forall y_n^m \in [k] \backslash \{x_n^m\}$, with the same probability q. Therefore, we have

$$Pr(y_n^m = a) = \begin{cases} \frac{e^{\varepsilon}}{k-1+e^{\varepsilon}} := p, & \text{if } a = x_n^m, \\ \frac{1}{k-1+e^{\varepsilon}} := q, & \text{otherwise}, \end{cases} \quad (6)$$

where ε is the privacy budget.

The truth information directly inferred from all perturbed sensory data is always biased. Thus, crowdsensing needs to calibrate the bias through an unbiased estimate. According to Warner's RR, i.e., (6), we can get the expectation of perturbed sensory data y_n^m (also a random variable) as follows,

$$E[y_n^m] = (p - q) \cdot x_n^m + \frac{(1 + k)k}{2} \cdot q. \quad (7)$$

Then, we have $x_n^m = \frac{2y_n^m - (1+k)k \cdot q}{2(p-q)}$ since $E[y_n^m] = y_n^m$ (the random variable y_n^m is unbiased). Therefore, the unbiased estimate of truth x_n^{truth} is updated according to

$$x_n^{truth} = \frac{-(1 + k)k \cdot q + \sum_{m=1}^{|U|} 2w_m \cdot y_n^m}{2(p - q) \cdot \sum_{m=1}^{|U|} w_m}. \quad (8)$$

Equation (8) shows that the calibrated truth x_n^{truth} must be sensitive to small changes in perturbed sensory data. This is because $0 < p - q < 1$ and then amplifies the weight of worker u_m, i.e, $w_m/(p - q) > w_m$.

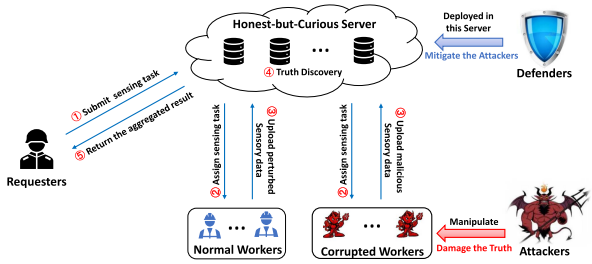


Fig. 1. Data poisoning attacks and defenses in LDP-based privacy-preserving crowdsensing systems.

IV. SYSTEM MODEL AND PROBLEM STATEMENT

Before diving into the details of optimal data poisoning attacks and defenses to LDP-based privacy-preserving crowdsensing, we first introduce the system model and then present the problem statement. Table II summarizes the key notations in our paper.

A. System Model

In this paper, we consider LDP-based privacy-preserving crowdsensing. It consists of a *server*, *requesters*, and multiple *privacy-conscious workers*, as shown in Fig. 1. Additionally, we consider the server to be *honest-but-curious*, i.e., the server honestly executes protocols such as the truth discovery but is curious about the workers' privacy, because the server may leak the workers' private information for profit. Particularly, the requesters submit the sensing tasks composed

of multiple objects, such as healthcare and travel records, to the honest-but-curious server. Then, the sensing tasks are assigned by the server to a group of participating workers and asked to upload sensory data. To prevent sensory data privacy leakage, the workers carry out the assigned sensing tasks and upload their perturbed sensory data instead of real sensory data to the server. After collecting the perturbed sensory data from all participating workers, the server estimates the truths by strictly performing the truth discovery method. Finally, the estimated truths are returned to the requesters.

Suppose there are *attackers* with the intent to attack the crowdsensing systems, and eventually overturn the truths estimated by the truth discovery methods. As depicted in Fig. 1, the attackers cannot manipulate the data uploaded by the *normal workers*, but they can create or recruit some *corrupted workers* and launch attacks by meticulously crafting their sensory data. In addition, we also consider *defenders* who deploy within the honest-but-curious server, intending to identify the corrupted workers by analyzing the sensory data collected by the server. In this system model, attackers and defenders do not transmit messages. That is, the defenders are unaware of the attack strategy employed by the attackers, and likewise, the attackers remain uninformed about the countermeasure adopted by the defenders. This paper focuses on researching data poisoning attacks and their countermeasures in the crowdsensing systems that protect workers' sensory data privacy through LDP. Although the worker weights may potentially leak sensitive information like their sensing capabilities and sensor quality, this is beyond the scope of this paper.

We consider each task composed of $|O|$ objects and those objects are completed by $|U|$ workers, where $O = \{o_1, o_2, \ldots, o_{|O|}\}$ is a set of objects and $U = \{u_1, u_2, \ldots, u_{|U|}\}$ is a set of workers. For $k \in \mathbb{N}$, let $x_n^m \in \{1, \ldots, k\}$ be the real sensory data of object o_n uploaded by worker u_m and its perturbed version be y_n^m, and $X = \{x_n^m \mid o_n \in O, u_m \in U\}$ is the sensory data shared by all participating workers. For convenience, let $[k] = \{1, \ldots, k\}$. The workers' reliability is denoted as worker weights $W = \{w_1, w_2, \ldots, w_{|U|}\}$, where w_m is the weight (or reliability) of worker u_m. As shown in Fig. 1, the participating workers U are composed of normal workers $\widehat{U} = \{\widehat{u}_1, \ldots, \widehat{u}_{|\widehat{U}|}\}$ and corrupted workers $\widetilde{U} = \{\widetilde{u}_1, \ldots, \widetilde{u}_{|\widetilde{U}|}\}$, i.e., $U = \widehat{U} \cup \widetilde{U}$. Similarly, we have $W = \widehat{W} \cup \widetilde{W}$, where $\widehat{W} = \{\widehat{w}_1, \ldots, \widehat{w}_{|\widehat{U}|}\}$ is the normal worker weights and $\widetilde{W} = \{\widetilde{w}_1, \ldots, \widetilde{w}_{|\widetilde{U}|}\}$ represents the corrupted worker weights. Besides, we denote the perturbed sensory data uploaded by privacy-conscious normal workers as $\widehat{Y} = \{\widehat{y}_n^m \mid o_n \in O, \widehat{u}_m \in \widehat{U}\}$ (i.e., the perturbed version of $\widehat{X} = \{\widehat{x}_n^m \mid o_n \in O, \widehat{u}_m \in \widehat{U}\}$, where real sensory data $\widehat{x}_n^m$ is perturbed to $\widehat{y}_n^m$). We also denote the malicious sensory data shared by corrupted workers as $\widetilde{X} = \{\widetilde{x}_n^m \mid o_n \in O, \widetilde{u}_m \in \widetilde{U}\}$, and the malicious sensory data is uploaded directly to the server without perturbation.

B. Problem Statement

The attackers who can craft malicious sensory data sent to the server will be able to disrupt the outcomes of the truth discovery methods, this type of attack is called a data poisoning attack. *Because the inferred truth is sensitive to small changes in the collected sensory data (detailed in (8)), attackers can exploit this sensitivity to overturn the inferred truth.* For instance, to disrupt the trustworthiness of inferred truth, the attackers only need to upload malicious sensory data that differs very little from the ground truth.

1) Attackers' Capability and Background Knowledge: As shown in Fig. 1, attackers can manipulate a group of corrupted workers and launch data poisoning attacks by asking them to submit malicious sensory data. The attackers' capability is realistic because of the openness of crowdsensing. As reported in [36], the attackers can easily access a large number of fake accounts in various services such as Twitter, Google, and Hotmail.

In our data poisoning attack, the attackers have full knowledge of the truth discovery method and the LDP protocol deployed in crowdsensing. Besides, the attackers also know the real sensory data of a subset of normal workers by stealing their credentials, compromising their computer systems, and even bribing them. Next, we detail two important definitions of the attackers' capability and background knowledge.

Definition 3 (Percentage of Corrupted Workers). Let workers $\widetilde{U} = \{\widetilde{u}_1, \widetilde{u}_2, \ldots, \widetilde{u}_{|\widetilde{U}|}\}$ be manipulated by attackers and all participating workers be $U = \{u_1, u_2, \ldots, u_{|U|}\}$, where $\widetilde{U} \subset U$. The percentage of corrupted workers is defined as

$$\varphi = |\widetilde{U}|/|U|, \tag{9}$$

where $|\widetilde{U}|$ and $|U|$ are the number of corrupted workers and all workers, respectively.

Definition 3 shows that a larger percentage of corrupted workers (i.e., parameter φ) means a stronger attackers' capability. We only consider $0 < \varphi \ll 0.5$, because the data poisoning attacks are guaranteed to succeed when $\varphi > 0.5$ (means that the number of corrupted workers is larger than the number of normal workers).

Definition 4 (Percentage of Known Workers). Let the attackers know the perturbed sensory data of workers U^{known} and the normal workers is denotes as $\widehat{U}$, where $U^{known} \subset \widehat{U}$. The percentage of known workers is defined as

$$\psi = |U^{known}|/|\widehat{U}|, \tag{10}$$

where $|U^{known}|$ is the number of known workers and $|\widehat{U}|$ denotes the number of normal workers.

Definition 4 shows that a larger percentage of known workers (i.e., parameter $0 < \psi \leq 1$) means a stronger background knowledge of the attackers.

2) Attackers' Goals: The attackers aim to damage the utility (or trustworthiness) of inferred truths as much as possible while remaining stealthy. These goals are well-motivated in real-world crowdsensing. For instance, attackers can manipulate traffic conditions displayed on Waze so that the victims travel as the attackers desire [37]. To this end, the attackers' goals can be formulated as a maximization problem:

$$\underset{\widetilde{X}}{\text{maximize}}\ \mathcal{U}\left(\mathcal{A}(\widehat{X}), \widetilde{X}, \mathcal{V}\right), \tag{11}$$

where $\mathcal{U}(*)$ is the utility loss of inferred truth (or attack gain), $\mathcal{A}$ denotes LDP protocol, $\widehat{X}$ is the real sensory data of normal workers, $\widetilde{X}$ represents the malicious sensory data submitted by corrupted workers, and $\mathcal{V}$ is the truth discovery method.

3) Defenders' Goals: We now define the defenders' objective: find an optimal way to defend against the proposed data poisoning attacks that are formulated in (11). The defenders need to design an optimal defense mechanism $\mathcal{T}$ so that (1) LDP is maintained; (2) the attackers' maximum achievable payoff is as low as possible subject to the privacy constraint. Therefore, when the attack strategy $\widetilde{X}$ is fixed, the optimal defense can be formulated as a minimization problem:

$$\underset{\mathcal{T}}{\text{minimize}}\ \mathcal{U}\left(\mathcal{A}(\widehat{X}), \widetilde{X}, \mathcal{V}, \mathcal{T}\right). \tag{12}$$

V. Designing Optimal Data Poisoning Attacks

The calibrating operation in LDP protocols, i.e., calibrating the bias arising from LDP noise, could amplify the influence of malicious sensory data on the inferred truths. Therefore, malicious sensory data with *subtle* deviations from real sensory data could overturn the inferred truths, which are hard to detect by the truth discovery methods. Powered by this intuition, we propose a novel attack "hiding behind the LDP noise", and subsequently design the optimization-based data poisoning attacks. Specifically, we give the LDP-based privacy-preserving truth discovery methods, which utilize the calibrating to mitigate the bias caused by LDP noise, inferring the truths from the collected data that contains both the malicious data and the privacy-preserving data added with LDP noise (detailed in Section V-A). Then, the attackers formulate the data poisoning attacks as a bi-level maximization problem, maximizing the deviation from the inferred truths under the constraint of privacy-preserving truth discovery methods (detailed in Section V-B). In this way, the attackers could maximize their attack profits while remaining stealthy. Finally, we employ the proposed optimal poisoning attack algorithm, which integrates the alternating optimization problem with the augmented Lagrangian method, to solve the formulated problem and obtain the optimal attack strategies (detailed in Section V-C).

A. LDP-Based Privacy-Preserving Truth Discovery Methods With Corrupted Workers

As shown in Fig. 1, the sensory data collected by the crowdsensing server consists of the perturbed sensory data (uploaded by normal workers) and the malicious sensory data (uploaded by corrupted workers). Therefore, to infer the truth information from the collected sensory data, CRH (introduced in Section III-A) is reformulated as

$$P_l: \underset{X_a^{truth}, \widehat{W}, \widetilde{W}}{\text{minimize}} \left\{ \sum_{m=1}^{|\widehat{U}|} \widehat{w}_m \cdot \sum_{n=1}^{|O|} d\left(\widehat{y}_n^m, x_{a,n}^{truth}\right) + \sum_{m=1}^{|\widetilde{U}|} \widetilde{w}_m \cdot \sum_{n=1}^{|O|} d\left(\widetilde{x}_n^m, x_{a,n}^{truth}\right) \right\}, \tag{13}$$

$$\text{subject to } \sum_{m=1}^{|\widehat{U}|} \exp(-\widehat{w}_m) + \sum_{m=1}^{|\widetilde{U}|} \exp(-\widetilde{w}_m) = 1, \tag{13a}$$

where $X_a^{truth} = \{x_{a,1}^{truth}, \ldots, x_{a,|O|}^{truth}\}$ is the inferred truths after attacks. The above problem P_l can be solved by Algorithm 1. Next, we reformulate the truth-updating function (i.e., (1) in Algorithm 1) and the worker-weight-updating function (i.e., (2) in Algorithm 1).

To satisfy ε-LDP, each normal worker strictly enforces Warner's RR (detailed in Definition 2) locally, i.e.,

$$Pr(\widehat{y}_n^m = a) = \begin{cases} \frac{e^\varepsilon}{k-1+e^\varepsilon} := p, & \text{if } a = \widehat{x}_n^m, \\ \frac{1}{k-1+e^\varepsilon} := q, & \text{otherwise.} \end{cases} \tag{14}$$

According to the (14), we have $E[\widehat{y}_n^m] = (p-q)\cdot \widehat{x}_n^m + \frac{(1+k)k}{2}\cdot q$. Due to $E[\widehat{y}_n^m] = \widehat{y}_n^m$, we have $\widehat{x}_n^m = \frac{2\widehat{y}_n^m - (1+k)k\cdot q}{2(p-q)}$. Thus, the unbiased estimation of $x_{a,n}^{truth}$ is updated based on

$$x_{a,n}^{truth} = T(\widehat{W}, \widetilde{W}, \widehat{Y}, \widetilde{X}), \tag{15}$$

$$:= \frac{-(1+d)k\cdot q + \sum_{m=1}^{|\widehat{U}|} 2\widehat{w}_m \widehat{y}_n^m + \sum_{m=1}^{|\widetilde{U}|} 2\widetilde{w}_m \widetilde{x}_n^m}{2(p-q)\cdot\left(\sum_{m=1}^{|\widehat{U}|} \widehat{w}_m + \sum_{m=1}^{|\widetilde{U}|} \widetilde{w}_m\right)}. \tag{16}$$

Equation (16) is the new truth-updating function, which shows that the inferred truth $x_{a,n}^{truth}$ is highly dependent on the sensory data uploaded by the higher-weight workers and is sensitive to small changes of malicious sensory data $\widetilde{x}_n^m$.

Similar to (4), we give the new worker-weight-updating function. That is, the normal worker weight $\forall \widehat{w}_m \in \widehat{W}$ and corrupted worker weight $\forall \widetilde{w}_m \in \widetilde{W}$ are calculated according to

$$\widehat{w}_m = \log \frac{\sum_{n=1}^{|O|}\left(\sum_{m=1}^{|\widehat{U}|} d(\widehat{y}_n^m, x_{a,n}^{truth}) + \sum_{m=1}^{|\widetilde{U}|} d(\widetilde{x}_n^m, x_{a,n}^{truth})\right)}{\sum_{n=1}^{|O|} d(\widehat{x}_n^m, x_{a,n}^{truth})}, \tag{17}$$

$$\widetilde{w}_m = \log \frac{\sum_{n=1}^{|O|}\left(\sum_{m=1}^{|\widehat{U}|} d(\widehat{y}_n^m, x_{a,n}^{truth}) + \sum_{m=1}^{|\widetilde{U}|} d(\widetilde{x}_n^m, x_{a,n}^{truth})\right)}{\sum_{n=1}^{|O|} d(\widetilde{x}_n^m, x_{a,n}^{truth})}. \tag{18}$$

Equation (18) shows that the corrupted workers obtain higher weights as long as their sensory data is closer to truths. In other words, the smaller $d(\widetilde{x}_n^m, x_{a,n}^{truth})$, the higher-weight of corrupted worker $\widetilde{u}_m$.

According to the new truth-updating function (i.e., (16)) and worker-weight-updating function (i.e., (17)–(18)), we can obtain latest Algorithm 1. Thus, the truths after attacks (i.e., X_a^{truth}) can be inferred by performing the latest Algorithm 1.

B. Formulating Optimal Data Poisoning Attacks

As explained in Section IV-B2, the attackers aim to find the optimal data poisoning attacks that *(1) maximize damage to inferred truth* and *(2) remain stealthy*. The damage caused by data poisoning attacks is defined as the amount of change in inferred truths. To achieve the first goal, for this reason, the attackers maximize the amount of change $\sum_{n=1}^{|O|} sgn(x_{a,n}^{truth} - x_n^{truth})$ (if the attackers aim to make $x_{a,n}^{truth} \gg x_n^{truth}$) or $\sum_{n=1}^{|O|} sgn(x_n^{truth} - x_{a,n}^{truth})$ (if the attackers aim to make $x_{a,n}^{truth} \ll x_n^{truth}$). The signum function $sgn(x_{a,n}^{truth} - x_n^{truth})$ is formulated as

$$sgn(x_{a,n}^{truth} - x_n^{truth}) = \begin{cases} 1, & \text{if } x_{a,n}^{truth} > x_n^{truth}, \\ 0, & \text{if } x_{a,n}^{truth} = x_n^{truth}, \\ -1, & \text{if } x_{a,n}^{truth} < x_n^{truth}. \end{cases} \quad (19)$$

The truth $X^{truth} = \{x_1^{truth}, x_2^{truth}, \ldots, x_{|O|}^{truth}\}$ is inferred by feeding the sensory data upload by known workers U^{known} to the Algorithm 1 (detailed in Section II-I-A). Therefore, the payoff of attackers is denoted as $\mathcal{U}(\mathcal{A}(\widehat{X}), \widetilde{X}, \mathcal{V}) = \sum_{n=1}^{|O|} sgn(x_{a,n}^{truth} - x_n^{truth})$ or $\mathcal{U}(\mathcal{A}(\widehat{X}), \widetilde{X}, \mathcal{V}) = \sum_{n=1}^{|O|} sgn(x_n^{truth} - x_{a,n}^{truth})$.

Corrupted workers detected by the truth discovery methods (such as CRH) have less impact over all objects in aggregation because they have lower weights. Thus, achieving the second goal, i.e., bypassing the truth discovery methods, is particularly important. To remain undetected by exploiting LDP perturbation, we treat the optimization problem P_l (i.e, (13) and detailed in Section V-A) as a constraint. Besides, to reduce the feasible region of attack strategy $\widetilde{X}$, we set the range of malicious sensory data to be the range of real sensory data known by the attackers. Formally, for $\forall o_n \in O, \forall \widetilde{u}_m \in \widetilde{U}$, the malicious sensory data $\widetilde{x}_n^m \in [k']$, where k' is the maximum sensory data known to attackers. After the above discussion, we formulated these two goals as the following optimization problem,

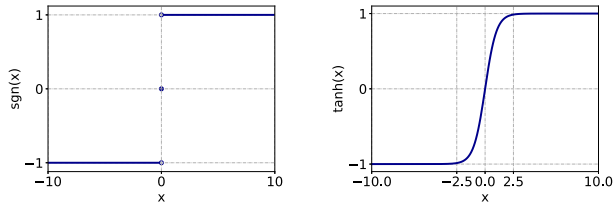
$$\begin{aligned} P: \underset{\widetilde{X}=\{\widetilde{x}_n^m | o_n \in O, \widetilde{u}_m \in \widetilde{U}\}}{\text{maximize}} & \sum_{n=1}^{|O|} sgn(x_{a,n}^{truth} - x_n^{truth}), \\ s.t.\, \widetilde{x}_n^m \in [k'],\ & \forall o_n \in O, \forall \widetilde{u}_m \in \widetilde{U}, \\ \{X_a^{truth}, \widehat{W}, \widetilde{W}\} := & \\ \underset{X_a^{truth}, \widehat{W}, \widetilde{W}}{\arg\min} & \left\{ \sum_{m=1}^{|\widehat{U}|} \widehat{w}_m \sum_{n=1}^{|O|} d\left(\widehat{y}_n^m, x_{a,n}^{truth}\right) \right. \\ & \left. + \sum_{m=1}^{|\widetilde{U}|} \widetilde{w}_m \cdot \sum_{n=1}^{|O|} d\left(\widetilde{x}_n^m, x_{a,n}^{truth}\right) \right\}, \\ \sum_{m=1}^{|\widehat{U}|} \exp(-\widehat{w}_m) + & \sum_{m=1}^{|\widetilde{U}|} \exp(-\widetilde{w}_m) = 1. \end{aligned} \quad (20)$$

The problem P (i.e., (20)) is a bi-level optimization problem [38] because its constraint is the problem P_l. Particularly, the optimization over $\widetilde{X}$ is the upper-level sub-problem, and the optimization over $(X_a^{truth}, \widehat{W}, \widetilde{W})$ given $\widetilde{X}$ is the lower-level sub-problem. We note that the attackers' goal (i.e., the problem P) is to increase the value of inferred truths. Conversely, if the attackers' goal is to reduce the value of inferred truths, we modify the object function (20) to

$$\underset{\widetilde{X}=\{\widetilde{x}_n^m | o_n \in O, \widetilde{u}_m \in \widetilde{U}\}}{\text{maximize}} \sum_{n=1}^{|O|} sgn(x_n^{truth} - x_{a,n}^{truth}). \quad (21)$$

Besides, the problem P is a general attack framework that can be applied to any optimization-based truth discovery method by changing the lower-level problem.

The data poisoning attacks are formulated as the bi-level maximization problem, allowing the attackers to obtain an effective attack strategy by solving this problem. Particularly, this problem aims to maximize the attack profits under the constraint of LDP-based privacy-preserving truth discovery methods. Therefore, the attackers can overturn the outputs of truth discovery as much as possible, which is supported by optimizing the formulated maximization problem.



(a) $sgn(x)$ (b) $\tanh(x)$

Fig. 2. Piecewise function $sgn(x)$ is relaxed into the continuous function $\tanh(x)$.

C. Finding Optimal Data Poisoning Attacks

The bi-level optimization problem is NP-hard [38], which makes it prohibitive to directly solve the formulated problem P. Thus, we utilize an alternating optimization algorithm [9], an iterative algorithm that alternately optimizes the upper-level sub-problem and the lower-level sub-problem, to solve problem P. The lower-level sub-problem P_l has been solved in Section V-A and then details how to solve the upper-level sub-problem P_u. When given worker weights $W = \widetilde{W} \cup \widehat{W}$, the upper-level sub-problem P_u is formulated as follows,

$$P_u: \underset{\widetilde{X}}{\text{maximize}} \sum_{n=1}^{|O|} sgn(x_{a,n}^{truth} - x_n^{truth}), \quad (22)$$

$$\text{subject to } \widetilde{x}_n^m \in [k'],\ \forall o_n \in O, \forall \widetilde{u}_m \in \widetilde{U}, \quad (22a)$$

$$x_{a,n}^{truth} = T(\widehat{W}, \widetilde{W}, \widehat{Y}, \widetilde{X}), \forall o_n \in O, \quad (22b)$$

where $T(\widehat{W}, \widetilde{W}, \widehat{Y}, \widetilde{X})$ is defined in (16) and $x_n^{truth}, \forall o_n \in O$ can be inferred by the attackers based on background knowledge.

The discreteness of (22) and (22a) brings great challenges to solving the problem P_u. To address these challenges, as shown in Fig. 2, we relax the signum function (19) to a continuous

function $\tanh(*)$, i.e.,

$$\tanh(x_{a,n}^{truth}-x_n^{truth})=\frac{e^{x_{a,n}^{truth}-x_n^{truth}}-e^{x_n^{truth}-x_{a,n}^{truth}}}{e^{x_{a,n}^{truth}-x_n^{truth}}+e^{x_n^{truth}-x_{a,n}^{truth}}}. \tag{23}$$

Then, we relax the constraint (22a) to $1\le\widetilde{x}_n^m\le k', \forall o_n\in O, \forall\widetilde{u}_m\in\widetilde{U}$. Thus far, we have relaxed the problem P_u into a continuous optimization problem P_u^1, i.e.,

$$P_u^1:\ \underset{\widetilde{X}}{\text{maximize}}\ \sum_{n=1}^{|O|}\tanh(x_{a,n}^{truth}-x_n^{truth}), \tag{24}$$

$$\text{subject to } 1\le\widetilde{x}_n^m\le k', \forall o_n\in O, \forall\widetilde{u}_m\in\widetilde{U}, \tag{24a}$$

$$x_{a,n}^{truth}=T(\widehat{W},\widetilde{W},\widehat{Y},\widetilde{X}), \forall o_n\in O. \tag{24b}$$

The problem P_u aims to overturn the inferred truth $x_n^{truth}, \forall o_n\in O$, while the problem P_u^1 aims to overturn and move away from the inferred truth $x_n^{truth}, \forall o_n\in O$. Therefore, relaxing (22) (the objective of problem P_u) to (24) (the objective of problem P_u^1) means enhancing the attack goal.

Besides, since the decision variable $\widetilde{X}=\{\widetilde{x}_n^m\mid\forall o_n\in O, \forall\widetilde{u}_m\in\widetilde{U}\}$ is an $|O|\times|\widetilde{U}|$ dimensional matrix, the computational complexity of solving problem P_u^1 grows rapidly with the number of corrupted workers $|\widetilde{U}|$. To overcome the curse of dimension, we let $\widetilde{X}^i=\widetilde{X}^j, \forall\widetilde{X}^i,\widetilde{X}^j\subset\widetilde{X}$, where $\widetilde{X}^i=\{\widetilde{x}_1^i,\ldots,\widetilde{x}_{|O|}^i\}$ is uploaded by corrupted worker $\widetilde{u}_i$ for all objects. This is more practical because the attackers have the most excellent chance of winning when all corrupted workers upload the same malicious sensory data, i.e., all corrupted workers exhibit identical malicious behaviors. Additionally, we have $x_{a,n}^{truth}\ge x_n^{truth}, \forall o_n\in O$ because the problem P_u^1 maximizes $\tanh(x_{a,n}^{truth}-x_n^{truth})$. To this end, we set $\widetilde{X}=[\widetilde{X}^i,\ldots,\widetilde{X}^i]_{|\widetilde{U}|}$, and then the problem P_u is relaxed to problem P_u^2, i.e.,

$$P_u^2:\ \underset{\widetilde{X}^i=\{\widetilde{x}_1^i,\ldots,\widetilde{x}_{|O|}^i\}}{\text{maximize}}\ \sum_{n=1}^{|O|}\tanh(x_{a,n}^{truth}-x_n^{truth}), \tag{25}$$

$$\text{subject to } 1\le\widetilde{x}_n^i\le k', \forall o_n\in O. \tag{25a}$$

$$x_{a,n}^{truth}=T(\widehat{W},\widetilde{W},\widehat{Y},\widetilde{X}), \forall o_n\in O, \tag{25b}$$

$$x_{a,n}^{truth}\ge x_n^{truth}, \forall o_n\in O. \tag{25c}$$

In what follows, we prove that the problem P_u^2 is a convex optimization problem.

Theorem 1. The problem P_u^2 is a convex optimization problem.

Proof. The problem P_u^2 is a convex optimization problem if and only if the following minimization problem is a convex optimization problem,

$$\underset{\widetilde{X}^i}{\text{minimize}}\ -\sum_{n=1}^{|O|}\tanh(x_{a,n}^{truth}-x_n^{truth}), \tag{26}$$

$$\text{subject to } \widetilde{x}_n^i\le k', -\widetilde{x}_n^i\le -1, \forall o_n\in O, \tag{26a}$$

$$x_{a,n}^{truth}=T(\widehat{W},\widetilde{W},\widehat{Y},\widetilde{X}), \forall o_n\in O, \tag{26b}$$

$$-x_{a,n}^{truth}\le -x_n^{truth}, \forall o_n\in O. \tag{26c}$$

Since $\widetilde{x}_n^i$ and $T(\widehat{W},\widetilde{W},\widehat{Y},\widetilde{X})$ (denoted in (16)) are the liner functions with respect to $\widetilde{x}_n^i, \forall o_n\in O$, the feasible region of the problem (26) is a convex set. Next, we focus on proving that the objective function is convex. Let $g(x)=-\tanh(x)$, we have $g'(x)=g^2(x)-1\le 0$ and $g''(x)=2g(x)\cdot g'(x)$. Since $g(x)\le 0$ when $x\ge 0$, we have $g''(x)\ge 0$ when $x\ge 0$, i.e., the function $g(x)$ is a monotonically decreasing convex function when $x\ge 0$. Since $\frac{\partial^2 g(x_{a,n}^{truth}-x_n^{truth})}{\partial(\widetilde{x}_n^i)^2}=g''(x_{a,n}^{truth}-x_n^{truth})\cdot((|\widetilde{U}|\cdot\widetilde{w}_i)/((p-q)\cdot(\sum_{m=1}^{|\widetilde{U}|}\widehat{w}_m+|\widehat{U}|\cdot\widetilde{w}_i)))^2$ and $g''(x)\ge 0$ when $x\ge 0$, we have $\frac{\partial^2 g(x_{a,n}^{truth}-x_n^{truth})}{\partial(\widetilde{x}_n^i)^2}\ge 0$ when $x_{a,n}^{truth}\ge x_n^{truth}$. That is, the object function $\sum_{n=1}^{|O|}g(x_{a,n}^{truth}-x_n^{truth})$ is a convex function with respect $\widetilde{X}^i$ when $x_{a,n}^{truth}\ge x_n^{truth}, \forall o_n\in O$. The constraint (26c) ensures that the condition $x_{a,n}^{truth}\ge x_n^{truth}, \forall o_n\in O$ is satisfy. In summary, the convex nature of both the feasible region and the object function constitute a convex optimization problem (26). As a result, the problem P_u^2 is a convex optimization problem.

Since problem P_u^2 is convex, it can be solved by the augmented Lagrangian method (a gradient-based optimization algorithm). By introducing the slack variables $S=\{s_1,s_2,\ldots,s_{|O|}\}$, $S'=\{s_1',s_2',\ldots,s_{|O|}'\}$, and $S''=\{s_1'',s_2'',\ldots,s_{|O|}''\}$, we can obtain an equivalent form of the problem P_u^2, i.e.,

$$\underset{\widetilde{X}^i}{\text{minimize}}\ \sum_{n=1}^{|O|}-\tanh(x_{a,n}^{truth}-x_n^{truth}), \tag{27}$$

$$\text{subject to } 1-\widetilde{x}_n^i+s_n=0, \forall o_n\in O, \tag{27a}$$

$$\widetilde{x}_n^i-k'+s_n'=0, \forall o_n\in O, \tag{27b}$$

$$x_{a,n}^{truth}-x_n^{truth}+s_n''=0, \forall o_n\in O, \tag{27c}$$

$$x_{a,n}^{truth}-T(\widehat{W},\widetilde{W},\widehat{Y},\widetilde{X})=0, \forall o_n\in O, \tag{27d}$$

$$s_n,s_n',s_n''\ge 0, \forall o_n\in O. \tag{27e}$$

The augmented Lagrangian function of the above optimization problem (27) is calculated as

$$\begin{aligned}&L_\sigma(\widetilde{X}^i,S,S',S'',\lambda,\mu,\nu)\\&=\sum_{n=1}^{|O|}-\tanh(x_{a,n}^{truth}-x_n^{truth})+\frac{\sigma}{2}\cdot\rho(\widetilde{X}^i,S,S',S'')\\&\quad+\sum_{n=1}^{|O|}\lambda_n\cdot(1-\widetilde{x}_n^i+s_n)+\sum_{n=1}^{|O|}\mu_n\cdot(\widetilde{x}_n^i-k'+s_n')\\&\quad+\sum_{n=1}^{|O|}\nu_n\cdot(x_{a,n}^{truth}-x_n^{truth}+s_n''),\end{aligned} \tag{28}$$

where $\lambda=\{\lambda_1,\ldots,\lambda_{|O|}\}$, $\mu=\{\mu_1,\ldots,\mu_{|O|}\}$, and $\nu=\{\nu_1,\ldots,\nu_{|O|}\}$ are Lagrange multipliers, σ is the penalty factor,

and the quadratic penalty function $\rho(\widetilde{X}^i, S, S', S'')$ is formulated as

$$\rho(\widetilde{X}^i, S, S', S'') = \sum_{n=1}^{|O|} \left(1 - \widetilde{x}_n^i + s_n\right)^2 + \sum_{n=1}^{|O|} \left(\widetilde{x}_n^i - k' + s_n'\right)^2 + \sum_{n=1}^{|O|} (x_{a,n}^{truth} - x_n^{truth} + s_n'')^2. \quad (29)$$

Fixing $\widetilde{X}^i$, λ, μ, and ν, the sub-problem of slack variables can be formulated as

$$\underset{S\geq 0}{\text{minimize}} \sum_{n=1}^{|O|} \lambda_n \cdot (1 - \widetilde{x}_n^i + s_n) + \frac{\sigma}{2} \cdot \sum_{n=1}^{|O|} (1 - \widetilde{x}_n^i + s_n)^2, \quad (30)$$

$$\underset{S'\geq 0}{\text{minimize}} \sum_{n=1}^{|O|} \mu_n \cdot (\widetilde{x}_n^i - k' + s_n') + \frac{\sigma}{2} \cdot \sum_{n=1}^{|O|} (\widetilde{x}_n^i - k' + s_n')^2, \quad (31)$$

$$\underset{S''\geq 0}{\text{minimize}} \sum_{n=1}^{|O|} \nu_n \cdot (x_{a,n}^{truth} - x_n^{truth} + s_n'') + \frac{\sigma}{2} \cdot \sum_{n=1}^{|O|} (x_{a,n}^{truth} - x_n^{truth} + s_n'')^2. \quad (32)$$

Therefore, we have

$$s_n = \max\left\{-\frac{\lambda_n}{\sigma} - 1 + \widetilde{x}_n^i, 0\right\}, \forall o_n \in O, \quad (33)$$

$$s_n' = \max\left\{-\frac{\mu_n}{\sigma} + k' - \widetilde{x}_n^i, 0\right\}, \forall o_n \in O, \quad (34)$$

$$s_n'' = \max\left\{-\frac{\nu_n}{\sigma} - x_{a,n}^{truth} + x_n^{truth}, 0\right\}, \forall o_n \in O. \quad (35)$$

The slack variables S, S', and S'' can be eliminated by bringing the (33)–(35) into the augmented Lagrangian function (28), and then we obtain a new augmented Lagrangian function $L_\sigma(\widetilde{X}^i, \lambda, \mu, \nu)$. Finally, the gradient of the augmented Lagrangian function $L_\sigma(\widetilde{X}^i, \lambda, \mu, \nu)$ with respect to $\widetilde{X}^i$, i.e., $\nabla_{\widetilde{X}^i} L_\sigma(\widetilde{X}^i, \lambda, \mu, \nu) = [\frac{\partial L_{\sigma_k}}{\partial \widetilde{x}_1^i}, \ldots, \frac{\partial L_{\sigma_k}}{\partial \widetilde{x}_n^i}]_{|O|}$, can be computed as follows,

$$\frac{\partial L_\sigma(\widetilde{X}^i, \lambda, \mu, \nu)}{\partial \widetilde{x}_n^i} = \left((\tanh(x_{a,n}^{truth} - x_n^{truth}))^2 - 1\right) \times \frac{\partial x_{a,n}^{truth}}{\partial \widetilde{x}_n^i} + \sigma \cdot \left((1 - \widetilde{x}_n^i + s_n) \times (-1 + \frac{\partial s_n}{\partial \widetilde{x}_n^i}) + (\widetilde{x}_n^i - k' + s_n') \times \left(1 + \frac{\partial s_n'}{\partial \widetilde{x}_n^i}\right) + (x_{a,n}^{truth} - x_n^{truth} + s_n'') \times \left(\frac{\partial x_{a,n}^{truth}}{\partial \widetilde{x}_n^i} + \frac{\partial s_n''}{\partial \widetilde{x}_n^i}\right)\right) + \lambda_n \cdot \left(-1 + \frac{\partial s_n}{\partial \widetilde{x}_n^i}\right) + \mu_n \cdot \left(1 + \frac{\partial s_n'}{\partial \widetilde{x}_n^i}\right) + \nu_n \cdot \left(\frac{x_{a,n}^{truth}}{\partial \widetilde{x}_n^i} + \frac{\partial s_n''}{\partial \widetilde{x}_n^i}\right), \forall o_n \in O, \quad (36)$$

where

$$\frac{\partial x_{a,n}^{truth}}{\partial \widetilde{x}_n^i} = \frac{\sum_{m=1}^{|\widetilde{U}|} \widetilde{w}_m}{(p-q) \cdot (\sum_{m=1}^{|\widetilde{U}|} \widehat{w}_m + \sum_{m=1}^{|\widehat{U}|} \widetilde{w}_m)}, \quad (37)$$

$$\frac{\partial s_n}{\partial \widetilde{x}_n^i} = \begin{cases} 1, & s_n > 0, \\ 0, & s_n = 0, \end{cases} \quad (38)$$

$$\frac{\partial s_n'}{\partial \widetilde{x}_n^i} = \begin{cases} -1, & s_n' > 0, \\ 0, & s_n' = 0, \end{cases} \quad (39)$$

$$\frac{\partial s_n''}{\partial \widetilde{x}_n^i} = \begin{cases} -\frac{\partial x_{a,n}^{truth}}{\partial \widetilde{x}_n^i}, & s_n'' > 0, \\ 0, & s_n'' = 0. \end{cases} \quad (40)$$

According to the above discussion, to solve the problem P, we propose an optimal poisoning attack algorithm that combines the alternating optimization algorithm with the augmented Lagrangian method as shown in Algorithm 2. The poisoning attack strategy $\widetilde{X} = [\widetilde{X}^i, \ldots, \widetilde{X}^i]_{|\widetilde{U}|}$ is initialized as a uniform distribution, i.e., $\forall \widetilde{x} \in \widetilde{X}^i$ and $\widetilde{x} \sim U(1, k')$. At each step in the iteration, we obtain the optimal solution $\widetilde{X}^{i,k+1}$ by alternately solving the lower-level sub-problem P_l (line 4) and the upper-level sub-problem P_u (lines 5 and 6) using the penalty factor σ_k and the Lagrange multipliers $\lambda^k = \{\lambda_1^k, \ldots, \lambda_{|O|}^k\}$, $\mu^k = \{\mu_1^k, \ldots, \mu_{|O|}^k\}$, and $\nu^k = \{\nu_1^k, \ldots, \nu_{|O|}^k\}$. Particularly, given current σ_k, λ^k, μ^k, and ν^k, we iterate the following three steps in sequence:

1) Fixing the attack strategy $\widetilde{X} = [\widetilde{X}^i, \ldots, \widetilde{X}^i]_{|\widetilde{U}|}$, the optimal solution $\{\widetilde{X}_a^{truth}, \widetilde{W}, \widehat{W}\}$ is updated by solving the lower-level sub-problem P_l (detailed in Section V-A);
2) Fixing $\{\widetilde{X}_a^{truth}, \widetilde{W}, \widehat{W}\}$, we calculate the gradient of augmented Lagrangian function $L_{\sigma_k}(\widetilde{X}^i, \lambda^k, \mu^k, \nu^k)$ with respect to $\widetilde{X}^i$, i.e., $\nabla_{\widetilde{X}^i} L_{\sigma_k}$, according to (36);
3) After obtaining the gradient $\nabla_{\widetilde{X}^i} L_{\sigma_k}$, we can update the attack strategy $\widetilde{X}^i$ as follows,

$$\widetilde{X}^i + \vartheta \cdot \nabla_{\widetilde{X}^i} L_{\sigma_k}(\widetilde{X}^i, \lambda^k, \mu^k, \nu^k), \quad (41)$$

where ϑ is the step size.

Then, we judge whether the constraint violation degree $v_k(\widetilde{X}^{i,k+1})$ meets the accuracy requirement (line 9). If so, update the Lagrange multipliers λ^{k+1}, μ^{k+1}, and ν^{k+1} (lines 13, 14, and 15), and improve the accuracy of solving the sub-problem (line 17), at this time, the penalty factor σ_{k+1} remains unchanged (line 16); If not, the Lagrange multipliers λ^{k+1}, μ^{k+1}, and ν^{k+1} are not changed (line 20) and increase the penalty factor σ_{k+1} (line 21) appropriately to obtain a solution with a smaller constraint violation degree (line 22). Finally, the constraint violation degree is formulated as follows

$$v_k(\widetilde{X}^{i,k+1}) = \left(\sum_{n=1}^{|O|} (1 - x_n^{i,k+1} + s_n)^2 + \sum_{n+1}^{|O|} (x_n^{i,k+1} - k' + s_n')^2 + \sum_{n=1}^{|O|} (x_{a,n}^{truth} - x_n^{truth} + s_n'')^2\right)^{\frac{1}{2}}, \quad (42)$$

Algorithm 2: Optimal Poisoning Attack Algorithm.

Input : Lagrange multipliers λ^0, μ^0, and ν^0; Penalty factor $\sigma_0 > 0$; Threshold for convergence $\eta > 0$; Constraint violation constant $\tau > 0$; Constants $0 < \alpha \leq \beta \leq 1$ and $\rho > 1$; Step size ϑ

Output: Optimal poisoning attack $\widetilde{X} = [\widetilde{X}^i, \cdots, \widetilde{X}^i]_{|\widetilde{U}|}$

Initialize the poisoning attack strategy $\widetilde{x} \sim U(1, k'), \forall \widetilde{x} \in \widetilde{X}^{i,0}$, $\eta_0 = \frac{1}{\sigma_0}$, and $\tau_0 = \frac{1}{\sigma_0^\alpha}$;
for $k = 0, 1, 2, 3, \cdots$ **do**
  while $\|\nabla_{\widetilde{X}^i} L_{\sigma_k}(\widetilde{X}^{i,k}, \lambda^k, \mu^k, \nu^k)\|_2 \leq \eta_k$ **do**
    $\{X_a^{truth}, \widehat{W}^k, \widetilde{W}^k\} \leftarrow$ Solving the lower-level problem P_l using $[\widetilde{X}^{i,k}, \cdots, \widetilde{X}^{i,k}]_{|\widetilde{U}|}$ by Algorithm 1;
    $\nabla_{\widetilde{X}^i} L_{\sigma_k}(\widetilde{X}^{i,k}, \lambda^k, \mu^k, \nu^k) = [\frac{\partial L_\sigma}{\partial \widetilde{x}_1^i}, \cdots, \frac{\partial L_\sigma}{\partial \widetilde{x}_{|\mathcal{O}|}^i}] \leftarrow$ Computing the gradient according to Eq. (36) using $X_g^{truth}, \widehat{W}^k, \widetilde{W}^k$;
    $\widetilde{X}^{i,k+1} \leftarrow \widetilde{X}^{i,k} + \vartheta \cdot \nabla_{\widetilde{X}^i} L_{\sigma_k}(\widetilde{X}^{i,k}, \lambda^k, \mu^k, \nu^k)$;
    $\widetilde{X}^{i,k} \leftarrow \widetilde{X}^{i,k+1}$;
  end
  if $v_k(\widetilde{X}^{i,k+1}) \leq \tau_k$ **then**
    if $v_k(\widetilde{X}^{i,k+1}) \leq \tau$ *and* $\|\nabla_{\widetilde{X}^i} L_{\sigma_k}(\widetilde{X}^{i,k+1}, \lambda^k, \mu^k, \nu^k)\|_2 \leq \eta$ **then**
      return $\widetilde{X}^{i,k+1}, \lambda^k, \mu^k, \nu^k$;
    end
    Update Lagrange multiplier λ_n^{k+1} based on $\max\{\lambda_n^k + \sigma_k \cdot (1 - \widetilde{x}_n^{i,k+1}), 0\}$, $\forall o_n \in O$;
    Update Lagrange multiplier μ_n^{k+1} based on $\max\{\mu_n^k + \sigma_k \cdot (\widetilde{x}_n^{i,k+1} - k'), 0\}$, $\forall o_n \in O$;
    Update Lagrange multiplier ν_n^{k+1} based on $\max\{\nu_n^k + \sigma_k \cdot (x_{a,n}^{truth} - x_n^{truth}), 0\}$, $\forall o_n \in O$;
    Keep penalty factor unchanged: $\sigma_{k+1} \leftarrow \sigma_k$;
    Reduce the threshold for convergence and the constraint violation constant of sub-problem based on $\eta_{k+1} \leftarrow \frac{\eta_k}{\sigma_{k=1}}$, $\tau_{k+1} \leftarrow \frac{\tau_k}{\sigma_{k+1}^\beta}$;
  end
  else
    Keep Lagrange multipliers unchanged: $\lambda^{k+1} \leftarrow \lambda^k$, $\mu^{k+1} \leftarrow \mu^k$, and $\nu^{k+1} \leftarrow \nu^k$;
    Update penalty factor σ_{k+1} based on $\rho \cdot \sigma_k$;
    Adjust the threshold for convergence and the constraint violation constant of the sub-problem based on $\eta_{k+1} \leftarrow \frac{1}{\sigma_{k+1}}$, $\tau_{k+1} \leftarrow \frac{1}{\sigma_{k+1}^\alpha}$;
  end
end

where the values of s_n, s_n', and s_n'' are calculated according to (33)–(35).

D. Computational Complexity and Limitations of the Attacks

The attackers could find the optimal attack strategy through Algorithm 2. In the subsequent discussion, we will conduct a detailed analysis of its computational complexity using the $\mathcal{O}$ notation (order of approximation). First, the computational complexity of line 4, line 5, and line 6 are $\mathcal{O}(|O| \cdot |U|)$, $\mathcal{O}(|O|)$, and $\mathcal{O}(|O|)$, respectively. Thereby, the computational complexity of lines 3 to 8 is $\mathcal{O}(c \cdot |O| \cdot (|U| + 2))$, where c represents the number of sequentially iterating lines 4 to 6. Second, the computational complexity of lines 9 to 23 is $\mathcal{O}(3 \cdot |O|)$. To this end, the computational complexity of finding the optimal data poisoning attacks is $\mathcal{O}(k_{max} \cdot c \cdot (|O| \cdot (|U| + 5))) \approx \mathcal{O}(k_{max} \cdot c \cdot |O| \cdot |U|)$, where k_{max} denotes the maximum iteration number of lines 3 to 23.

We demonstrate that LDP increases the vulnerability of crowdsensing systems to data poisoning attacks. However, there are still two limitations that could be addressed in future research. The first limitation of attacks is that they are built on the truth discovery methods and the LDP protocols, wherein the truth discovery methods are difficult for attackers to be aware of because they are deployed on the crowdsensing server. Second, the attacks rely on a subset of real sensory data, acquired by compromising some normal workers. For the first limitation, we will consider how to remain stealthy without background knowledge of the truth discovery methods. For the first limitation, we will consider how to remain stealthy without background knowledge of the truth discovery methods.

VI. Countermeasures: Designing the Optimal Defenses

A. Formulating Optimal Countermeasures

As explained in Section IV-B3, the goal of defenders is to defend against the proposed data poisoning attacks. The attackers' goal is to damage the inferred truth as much as possible, which is achieved by manipulating some corrupted workers to upload malicious sensory data. Therefore, defenders can identify workers who have the largest influence on damaging the inferred truth as corrupted by analyzing the sensory data collected by the server. To this end, we take the corruption probabilities $P_D = \{p_1, p_2, \ldots, p_{|U|}\}$ as decision variables, where $\forall p_m \in P_D, 0 \leq p_m \leq 1$ represents the probability that the worker u_m is corrupted. Let's assume that the defenders know the number of corrupted workers the attackers can manipulate. The corrupted workers detected by defenders can be formulated as,

$$U_D = \{u_m | \forall p_m \in sort(P_D, N)\}, \tag{43}$$

where N is the number of corrupted workers known by the defenders, and $sort(P_D, N)$ represents the top N workers most likely to be corrupted, i.e., the N-th highest p_m.

According to the above discussion, the optimal countermeasures are formulated as the minimization problem Q, i.e.,

$$Q: \underset{P_D}{\text{minimize}} \sum_{n=1}^{|O|} sgn(x_{a,n}^{truth} - x_n^{truth}), \tag{44}$$

$$\text{subject to } U_D = \{u_m | \forall p_m \in sort(P_D, N)\}, \tag{44a}$$

$$\widetilde{Z} = \widetilde{X} \backslash \{\widetilde{x}_n^m \mid \widetilde{u}_m \in \widetilde{U} \cap U_D\}, \tag{44b}$$

$$\widehat{Z} = \widehat{X} \backslash \{\widehat{y}_n^m \mid \widehat{u}_m \in \widehat{U} \cap U_D\}, \tag{44c}$$

Algorithm 3: Optimal Poisoning Attack Algorithm.

Input : Size of population SP; Maximum number of iterations gen_{max}; Amplification factor $F \in [0,2]$; Crossover factor $CR \in [0,1]$
Output: Corrupted workers U_D detected by defenders

1 Initialize a population of SP individuals $\mathbf{P}_D^0 = \{P_{D,0}^0, P_{D,1}^0, \cdots, P_{D,SP}^0\}$ with $\forall j \in [0, SP]$, $P_{D,j}^0 = \{p_{j,0}^0, p_{j,1}^0, \cdots, p_{j,|U|}^0\}$ uniformly distribution in range $[0,1]$;
2 $P_{best}^0 \leftarrow P_{D,0}^0$;
3 **for** $j = 0, 1, 2, \cdots, SP$ **do**
4 **if** $H(P_{best}^0) > H(P_{D,j}^0)$ **then**
5 $P_{best}^0 \leftarrow P_{D,j}^0$;
6 **end**
7 **end**
8 **for** $G = 1, 2, 3, \cdots, gen_{max}$ **do**
9 **for** $j = 0, 1, 2, \cdots, SP$ **do**
10 $r_1, r_2, r_3 \leftarrow rand([0, SP])$;
11 $B_j^G \leftarrow P_{D,r_1}^{G-1} + F \cdot (P_{D,r_2}^{G-1} - P_{D,r_3}^{G-1})$;
12 **end**
13 **for** $j = 0, 1, 2, \cdots, SP$ **do**
14 **for** $i = 0, 1, 2, \cdots, |U|$ **do**
15 **if** $rand(0,1) > CR$ **then**
16 $c_{j,i}^G \leftarrow p_{j,i}^{G-1}$;
17 **end**
18 **else**
19 $c_{j,i}^G \leftarrow b_{j,i}^G$;
20 **end**
21 **end**
22 **end**
23 **for** $j = 0, 1, 2, \cdots, SP$ **do**
24 **if** $H(C_j^G) < H(P_{D,j}^{G-1})$ **then**
25 $P_{D,j}^G \leftarrow C_j^G$;
26 **end**
27 **if** $H(P_{best}^{G-1}) > H(C_j^G)$ **then**
28 $P_{best}^G \leftarrow C_j^G$;
29 **end**
30 **end**
31 **end**
32 $U_D \leftarrow \{u_m | p_m \in sort(P_{best}^G, N)\}$;

where $x_{a,n}^{truth}, \forall o_n \in O$ is the optimal solution of problem P_l given sensory data $Z = \widetilde{Z} \cup \widehat{Z}$ (detailed in Section V-A). Constraints (44b)–(44c) show that the sensory data uploaded by the detected corrupted workers will be deleted in $X = \widetilde{X} \cup \widehat{X}$.

The countermeasures are formulated as minimization problems, minimizing the damage arising from attacks by identifying and deleting corrupted workers. Therefore, the defenders can detect the workers who have a great impact on the attack profits by solving the formulated minimization problem. As a result, we can ensure those countermeasures are effective, which is supported by the optimization theory.

B. Finding Optimal Countermeasures

The formulated problem Q cannot be solved by gradient-based optimization algorithms, because its objective (44) is not differentiable with respect to P_D. Thus, we adopt the Differential Evolution algorithm [39], [40], an efficient and powerful swarm-based stochastic search technique, to solve the problem Q. The details of finding optimal defenses are presented in Algorithm 3, i.e., Optimal Defense Algorithm (ODA). The ODA first performs *initialization* and then iteratively performs *mutation*, *crossover*, and *selection*, which are described in detail below.

In the initialization phase (lines 1 to 7), the ODA generates a uniform distribution initial population of SP individuals, (line 1), and then the optimal individual in the initialization population is selected according to the fitness function $H(*)$ (lines 2 to 7). Particularly, the 0th population, i.e., $\mathbf{P}_D^0 = \{P_{D,0}^0, \ldots, P_{D,SP}^0\}$, contains SP individuals. Each individual $\forall P_{D,j}^0 = [p_{j,0}^0, \ldots, p_{j,|U|}^0]_{|U|} \in \mathbf{P}_D^0$ is a solution candidate of problem Q, where $\forall p_{j,i}^0 \in P_{D,j}$ drawn from uniformly distribution $U(0,1)$. Then, the fitness value of individual $P_{D,j}^0$ (j-th individual in 0th population), i.e., $H(P_{D,j}^0)$, is calculated according to the following steps:

1) Given any individual $P_{D,j}^0 \in \mathbf{P}_D^0$, we obtain the corresponding corrupted workers U_D according to (43);
2) We delete the sensory data uploaded by U_D from the sensory data set X, and then obtain the new sensory data set Z (detailed in (44b)–(44c));
3) After obtain the sensory data set Z, we obtain the truths X_a^{truth} by solving the problem P_l (detailed in Section V-A) and then calculate the fitness value of individual $P_{D,j}^0$ by (44).

In the mutation phase (lines 9 to 12), we generate a mutant individual $B_j^G = [b_{j,0}^G, \ldots, b_{j,|U|}^G]_{|U|}$ (mutant individual of j-th individual in $(G-1)$-th population) for each individual by

$$B_j^G = P_{D,r_1}^{G-1} + F \cdot (P_{D,r_2}^{G-1} - P_{D,r_3}^{G-1}), \tag{45}$$

where F is amplification factor, P_{D,r_1}^{G-1} is the r_1-th individual in $(G-1)$-th population, and r_1, r_2, and r_3 are random numbers drawn from uniformly distribution $U(0, SP)$.

To increase the diversity of the populations, *the crossover phase (lines 13 to 22)* is introduced. According to the $P_{D,j}^{G-1} = [p_{j,0}^{G-1}, \ldots, p_{j,|U|}^{G-1}]_{|U|}$ (j-th individual in $(G-1)$-th population) and the $B_j^G = [b_{j,0}^G, \ldots, b_{j,|U|}^G]_{|U|}$ (mutant individual of $P_{D,j}^{G-1}$), we obtain the trial individual $C_j^G = [c_{j,0}^G, \ldots, c_{j,|U|}^G]_{|U|}$ (trial individual of $P_{D,j}^{G-1}$) by

$$c_{j,i}^G = \begin{cases} p_{j,i}^{G-1}, & \text{if } rand(0,1) > CR, \\ b_{j,i}^G, & \text{else}, \end{cases} \tag{46}$$

where $CR \in [0,1]$ represents crossover factor.

To obtain new (or next generation) population $\mathbf{P}_D^G$ (G-th population), we adopt greedy criterion in *the selection phase (lines 23 to 31)*. That is, trial individual C_j^G is selected when the fitness value of $H(C_j^G)$ smaller than the fitness value of $P_{D,j}^{G-1}$ (i.e., $H(C_j^G) < H(P_{D,j}^{G-1})$); otherwise, the individual $P_{D,j}^{G-1}$ is retained. Through iterating *mutation*, *crossover*, and *selection*, we obtain the optimal population $\mathbf{P}_D^{gen_{max}}$ (gen_{max}-th population). Finally, we select the individual with the smallest fitness value from population $\mathbf{P}_D^{gen_{max}}$ as the optimal solution of problem Q, which minimizes (44) and so does the attack gain.

TABLE III
SAMPLE RECORDS IN THE WEATHER SENTIMENT DATASET

WorkerID	ObjectID	Response	Ground-Truth
81991168	A2N7I0331X72Z7	5	5
79196168	A2HBI8BAE2C5QH	3	2

TABLE IV
SAMPLE RECORDS IN THE DUCHENNE SMILE DATASET

WorkerID	ObjectID	Response	Ground-Truth
AODPVTKUSNBUT	2162	1	1
A31R9OP1C70KJF	2162	1	1

C. Computational Complexity and Limitations of the Countermeasures

The optimal countermeasures are given in Algorithm 3. Specifically, ODA begins with initialization and then executes mutation, crossover, and selection during each iteration. The computational complexity of these four phases, i.e., initialization, mutation, crossover, and selection, are $\mathcal{O}(SP \cdot (|U| \cdot \log|U| + 2 \cdot |U| \cdot |O| + |U|))$, $\mathcal{O}(SP)$, $\mathcal{O}(SP \cdot |U|)$, and $\mathcal{O}(SP \cdot (|U| \cdot \log|U| + 2 \cdot |U| \cdot |O|))$, respectively. Thus, the computational complexity of the proposed countermeasures is $\mathcal{O}(gen_{max} \cdot SP \cdot |U| \cdot (\log|U| + |O|))$.

We propose optimization-based defenses to mitigate the vulnerability caused by LDP, verifying through experiments that they can detect corrupted workers (see Section VII-C for details). A possible limitation of the proposed defense methods is that the defenders are aware of the attackers' intent to overturn the truth data. To address this limitation, we will study how to identify corrupted workers without any background knowledge in the future.

VII. EXPERIMENTAL EVALUATION

In this section, we validate the effectiveness of our proposed data poisoning attacks and corresponding countermeasures.

A. Experiment Setup

In this part, we give details of the experiment setup containing the datasets, the comparisons of data poisoning attacks, the metrics, and the parameter setting. All experiments are implemented in Python and performed on a laptop with an Intel i7 CPU and 16 GB memory.

1) Datasets: We adopt two real-world datasets, i.e., Weather Sentiment Dataset [41] and Duchenne Smile Dataset [42], for experimental evaluation. Next, we overview these two real-world datasets respectively.

Weather Sentiment Dataset [41]. In this dataset, the task is to judge the sentiment of tweets (objects). The sentiment judgments are provided in the following categories: Neutral (label: 1), Positive (label: 2), Tweet not Related to Weather (label: 3), Can't Tell (label: 4), and Negative (label: 5). This dataset contains 6,000 records provided by 110 workers for 300 tweets. Each record includes the *WorkerID*, *ObjectID*, *Response*, and *Ground-Truth*. Table III shows two sampled records in the weather sentiment dataset.

Duchenne Smile Dataset [42]. In this dataset, the task is to judge whether the smile in a face image (an object) is Duchenne (label: 1) or Non-Duchenne (label: 2). The Duchenne smile dataset contains 17,729 records provided by 64 workers for 2,134 face images. Each record includes the *WorkerID*, *ObjectID*, *Response*, and *Ground-Truth*. Table IV shows two sampled records in the Duchenne smile dataset.

2) Comparisons of Data Poisoning Attacks: We evaluate our proposed optimal data poisoning attack by comparing it with four classical data poisoning attacks: RA, MA, OA, and AD. The effectiveness of our proposed optimal data poisoning attack can be demonstrated by comparing it with these four data poisoning attacks. Next, we detail RA, MA, OA, and AD [6] respectively.

Random Attack (RA). In RA, each corrupted worker randomly uploads sensory data from $[k']$, i.e., $\widetilde{x} \sim U(1, k')$ where symbol $U(\cdot)$ represents a uniform distribution. Thus, RA does not consider collusion among corrupted workers.

Maximum Utility Damage Attack (MA). In MA, the attackers obtain the inferred truths of $|O|$ objects by performing the truth discovery method (i.e., CRH) on the real sensory dataset of known workers U^{known}. Then, this attack takes the values that differ from the inferred truth of object o_n as the candidate set of the object o_n. Finally, all corrupted workers randomly choose the same sensory data from the candidate set, i.e., $\forall o_n \in O, \widetilde{x}_n \sim U([k']\backslash\{x_n^{truth}\})$ where $U(\cdot)$ denotes a uniform distribution. Thus, MA is an attack strategy that maximizes damage to the inferred truth while ignoring hidden malicious behavior.

Optimal Stealth Attack (OA). In this attack, the sensory data uploaded by the corrupted workers is the truth inferred from the real sensory data known to attackers, i.e., $\forall o_n \in O, \widetilde{x}_n = x_n^{truth}$. Thus, OA is an optimal stealth attack that ignores the maximum overturn of inferred truth x_n^{truth}.

Attack Under Disguise (AD) [6]. AD is a state-of-the-art data poisoning attack against discrete crowdsensing without privacy protection. In AD, each corrupted worker disguises himself (or improves his/her reliability) by trying to agree with normal workers on some objects whose truth is unlikely to be overturned; conversely, each corrupted worker disagrees with normal workers on other objects to maximize the damage of inferred truth.

3) Comparison of Defenses: We compare our defense method with MWA defense [8]. The MWA defense focuses on designing a resilient truth discovery method instead of introducing additional defense mechanisms (deployed on the crowdsensing server) to identify corrupted workers. Specifically, the MWA defense utilizes the following three steps to update truths: *First*, the server arranges workers in ascending order based on the uploaded data for n-th object (i.e., $\forall o_n \in O$), and then partitions those workers into L groups. *Second*, the server estimates the truths in each group according to (16), obtaining L aggregated results. *Third*, the server takes the median of these aggregated results as the inferred truth of n-th object. As discussed above,

the truth-updating function can be formulated as

$$x_{a,n}^{truth} = \text{Median}\Bigg(\frac{-(1+d)k\cdot q + 2\sum_{u\in U_{g,n}^1} w_u \cdot x_n^u}{2(p-q)\cdot \sum_{u\in U_{g,n}^1} w_u}, \ldots, \frac{-(1+d)k\cdot q + 2\sum_{u\in U_{g,n}^L} w_u \cdot x_n^u}{2(p-q)\cdot \sum_{u\in U_{g,n}^L} w_u}\Bigg), \quad \forall o_n \in O \tag{47}$$

where $U_{g,n}^i$ is the i-th worker group of the n-th object and $U_{g,n}^1 \cup \cdots \cup U_{g,n}^L = U$. In addition, the MWA defense employs the same worker-weight-updating functions as the LDP-based privacy-preserving truth discovery methods, i.e., (17) and (18), for updating the worker weights.

4) Metrics: We detail the metrics used to evaluate the effectiveness of our proposed data poisoning attacks and countermeasures.

The Effectiveness of Our Proposed Data Poisoning Attacks. To evaluate the effectiveness of data poisoning attacks, we compare the inferred truth before and after the attack and adopt Average Utility Damage (AUD) as a metric. The AUD is defined as

$$\text{AUD} := \frac{\sum_{o_n\in O} \mathbb{1}_{x_{a,n}^{truth} \neq x_{g,n}^{truth}}}{|O|}, \tag{48}$$

where $x_{g,n}^{truth}$ is the ground truth data of object o_n, and the indicator function $\mathbb{1}_{x_{a,n}^{truth} \neq x_{g,n}^{truth}}$ is formulated as

$$\mathbb{1}_{x_{a,n}^{truth} \neq x_{g,n}^{truth}} = \begin{cases} 1 & \text{if } x_{a,n}^{truth} \neq x_{g,n}^{truth} \\ 0 & \text{if } x_{a,n}^{truth} = x_{g,n}^{truth} \end{cases}. \tag{49}$$

Thus, the larger AUD, the better effectiveness of the data poisoning attack.

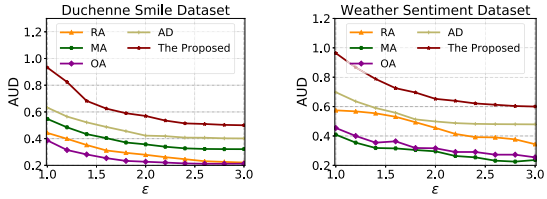
The Effectiveness of Our Proposed Countermeasures. The effectiveness of defenses can be evaluated by AUD (detailed in (48)), with a lower AUD indicating superior defense performance. Besides, our defense mechanism can be regarded as a classification classifier, which can be evaluated through AUC (i.e., the area under the Receiver Operating Characteristic (ROC) curve). Obviously, the larger AUC, the better the effectiveness of our proposed countermeasures.

5) Parameter Setting: We present parameter settings from two aspects: *Optimal Data Poisoning Attacks* and *Countermeasures*. Next, we overview these two aspects below.

Optimal Data Poisoning Attacks. As presented in Section V, we have two important parameters that affect the effectiveness of our proposed data poisoning attack, that is, ψ (i.e., the percentage of known workers and detailed in Definition 4) and φ (i.e., the percentage of corrupted workers and detailed in Definition 3). Particularly, the attackers not only know the real sensory data of $|U^{known}| = \psi \times |\widehat{U}|$ normal workers but also manipulate $|\widetilde{U}| = \varphi \times |U|$ corrupted workers. Next, we set parameters $\varepsilon = 1.8$, $\psi = 0.3$, $\varphi = 0.15$ based on the empirical experiments (see Section VII-B for details). Moreover, some empirical parameters in Algorithm 2 need to be set. We briefly list these parameters: the Lagrange multipliers $\lambda^0 = \mathbf{0}$, $\mu^0 = \mathbf{0}$, and $\nu^0 = \mathbf{0}$, the penalty factor $\sigma_0 = 2$, the threshold for convergence $\eta = 0.01$, the constraint violation constant $\tau = 0.05$, and the constants $\alpha = 0.1$, $\beta = 0.2$, and $\rho = 3$.

TABLE V
COMPUTATIONAL COMPLEXITY OF DATA POISONING ATTACKS

Attack Strategy	Computational Complexity
RA	$\mathcal{O}(\lvert O\rvert \cdot \lvert U\rvert)$
MA	$\mathcal{O}(\lvert O\rvert \cdot \lvert U\rvert)$
OA	$\mathcal{O}(\lvert O\rvert \cdot \lvert U\rvert)$
AD	$\mathcal{O}(k_{max} \cdot \lvert O\rvert \cdot \lvert U\rvert)$
Our	$\mathcal{O}(k_{max} \cdot c \cdot \lvert O\rvert \cdot \lvert U\rvert)$



(a) Impact of ε on effectiveness (b) Impact of ε on effectiveness

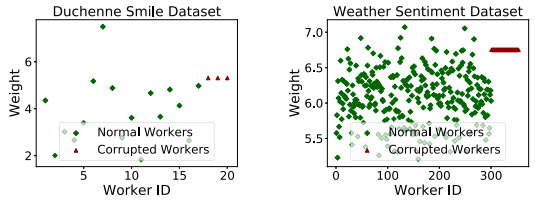
Fig. 3. Effectiveness of data poisoning attacks in both real-world datasets varying with privacy budget ε. (a) AUD in the Duchenne smile dataset and (b) AUD in the weather sentiment dataset.

Countermeasures. As presented in Section VI, we assume that the defenders know the number of corrupted workers. Thus, only some empirical parameters in the Algorithm 3 need to be set: the size of population SP, the maximum number of iterations gen_{max}, the amplification factor F and the crossover factor CR. Following [40], we set $SP = 20$, $gen_{max} = 1000$, $F = 0.5$, and $CR = 0.1$.

B. Attack Evaluation

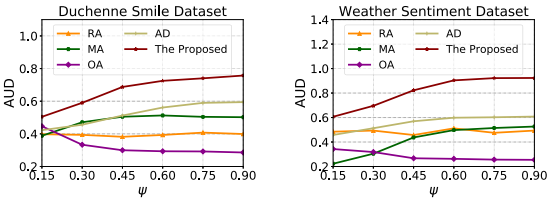
In this part, we show the effectiveness of data poisoning attacks against LDP-based privacy-preserving crowdsensing systems. Then, we evaluate the effectiveness of the proposed data poisoning attack with varying parameters. Additionally, we have shown the computational complexity of data poisoning attacks in Table V. This table shows that the proposed data poisoning attack has only slightly higher computational complexity than the comparisons. We observe that $\mathcal{O}(k_{max} \cdot c \cdot |O| \cdot |U|) \approx \mathcal{O}(|O| \cdot |U|)$ when crowdsensing involves a substantial number of participating workers and objects ($|U| \gg k_{max}, c$ and $|O| \gg k_{max}, c$), indicating that the computational complexity of the proposed data poisoning attack is approximately equal to that of the comparisons.

1) Data Poisoning Attacks Against the LDP-Based Privacy-Preserving Crowdsensing: Figs. 3 and 4 show the effectiveness of our proposed poisoning attack against LDP-based privacy-preserving crowdsensing. As shown in Fig. 3, our proposed data poisoning attack outperforms RA, OA, MA, and AD in the sense that, for the same privacy budget ε, its AUD is significantly larger than that of RA, OA, MA, and AD. The superiority of our proposed data poisoning attack is attributed to the fact that the RA, OA, MA, and AD are not optimized for the LDP-based privacy-preserving truth discovery method (see Section V-A for



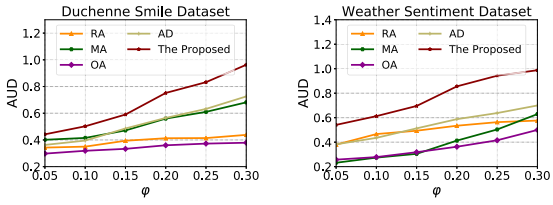
(a) Privacy budget $\varepsilon = 1.8$ (b) Privacy budget $\varepsilon = 1.8$

Fig. 4. Weights of workers U under our proposed data poisoning attack against LDP-based privacy-preserving crowdsensing with privacy budget $\varepsilon = 1.8$. (a) Worker weights in the Duchenne smile dataset and (b) worker weights in the weather sentiment dataset.



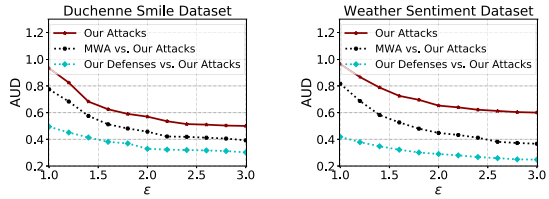
(a) Impact of ψ on effectiveness (b) Impact of ψ on effectiveness

Fig. 5. Effectiveness of data poisoning attacks in both real-world datasets varying with parameter ψ (the percentage of known workers). (a) AUD in the Duchenne smile dataset and (b) AUD in the weather sentiment dataset.



(a) Impact of φ on effectiveness (b) Impact of φ on effectiveness

Fig. 6. Effectiveness of data poisoning attacks in both real-world datasets varying with parameter φ (the percentage of corrupted workers). (a) AUD in the Duchenne smile dataset and (b) AUD in the weather sentiment dataset.



(a) Impact of ε on effectiveness (b) Impact of ε on effectiveness

Fig. 7. Effectiveness of defenses (measured by AUD) in both real-world datasets varying with privacy budget ε. (a) AUD in the Duchenne smile dataset and (b) AUD in the weather sentiment dataset.

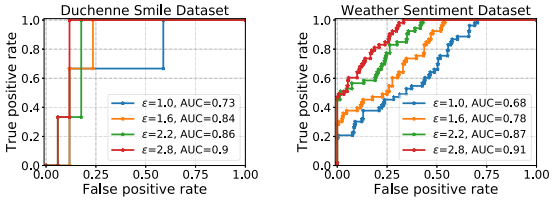
details). In other words, those comparisons do not leverage the risk (amplifying the impact of malicious sensory data) arising from the calibrating operation in LDP protocols. In addition, the superiority of our proposed data poisoning attack becomes larger when the privacy budget ε decreases. This occurs because the amplification of malicious sensory data becomes increasingly pronounced (detailed in (16)) as privacy protection becomes more rigorous, i.e., a smaller privacy budget ε. Moreover, as depicted in Fig. 4, our proposed data poisoning attack can evade the truth discovery methods by exploiting the LDP perturbation, resulting in the assignment of higher weights to the corrupted workers. It is crucial to emphasize that the equal weight of corrupted workers is due to their consistent malicious sensory data (refer to the process of relaxing the optimization problem P_u^1 into P_u^2). Therefore, the attack profits (measured by AUD) increase rapidly as the privacy budget decreases. As a result, Figs. 3 and 4 illustrate that the LDP facilitates the success of data poisoning attacks in evading the truth discovery methods.

2) The Effect of the Percentage of Known Workers: Fig. 5 shows the AUD of different data poisoning attacks (the proposed data poisoning attack and the comparison methods) varying with ψ (i.e., the percentage of known workers and defined in Definition 4) on two real-world datasets. We find that the AUD of the proposed data poisoning attack, RA, MA, and AD increase with parameter ψ. Conversely, the AUD of OA slightly decreases along with the parameter ψ increase. The attackers could infer more accurate truths when they know more perturbed sensory data of normal workers, reducing the error between the inferred truths X^{truth} and the truth data X_r^{truth}. As a result, both our proposed attack and the comparison attacks can better disguise themselves, but the malicious sensory data uploaded by OA is closer to the truth data. Therefore, the AUD of OA (or the proposed attack, RA, MA, and AD) decreases (or increases) as parameter ψ increases. Besides, Fig. 5 also shows that our proposed data poisoning attack significantly outperforms RA, MA, AD, and OA in terms of AUD, and the superiority of our proposed data poisoning attack increases as parameter ψ increases. This is because the stealth strategy of our proposed data poisoning attack is optimal for LDP-based privacy-preserving crowdsensing. That is, our data poisoning attacks could exploit the calibrating operation in LDP protocols to amplify the damage of malicious sensory data on the truth data, an aspect not considered by the comparison attacks.

3) The Effect of the Percentage of Corrupted Workers: We also investigate the impact of φ (i.e., the percentage of corrupted workers and defined in Definition 3) on the attack effectiveness (i.e., AUD), which is shown in Fig. 6. It shows that the AUD of our proposed data poisoning attack increases as the percentage of corrupted workers (i.e., the parameter φ) grows. This observation resonates with our intuition, as data poisoning attacks could exhibit strong performance when a substantial number of corrupted workers are manipulated, i.e., the attackers can inject more malicious sensory data. In addition, Fig. 6 also shows that the AUD of our proposed data poisoning attack is greater than all comparison attacks. This result verifies the attackers could bypass the truth discovery methods by strategically employing LDP noise, optimizing their profits of overturning the truth data.

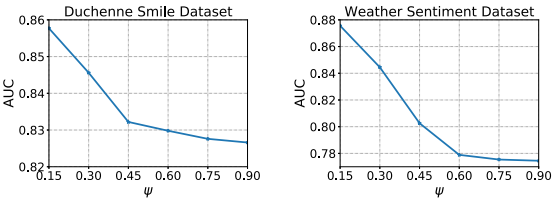
C. Defense Evaluation

Fig. 7 shows the effectiveness of our countermeasure and MWA in defending against the proposed data poisoning attacks. We observe that our countermeasure outperforms MWA, i.e., the AUD of our countermeasure is notably lower than that of



(a) ROC for different ε (b) ROC for different ε

Fig. 8. Effectiveness of the proposed countermeasure (measured by AUC) varying with the privacy budget ε. (a) ROC in the Duchenne smile dataset and (b) ROC in the weather sentiment dataset.



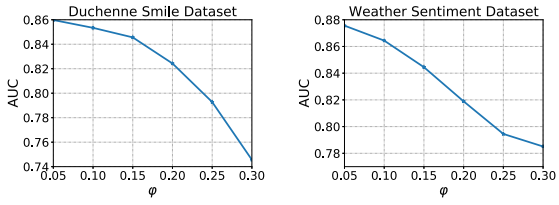
(a) Impact of ψ on AUC (b) Impact of ψ on AUC

Fig. 9. Effectiveness of the proposed countermeasure (measured by AUC) varying with parameter ψ (the percentage of known workers). (a) AUC in the Duchenne smile dataset and (b) AUC in the weather sentiment dataset.

MWA for the same privacy budget ε. The superiority of our countermeasure stems from its capability to efficiently identify workers causing substantial harm to the truth data through an optimization-based method. In addition, the superiority of our countermeasure becomes larger when the privacy budget ε decreases, verifying this countermeasure could mitigate the potential risk arising from LDP.

As shown in Fig. 8, we carry out four groups of experiments to evaluate the effectiveness of the proposed countermeasure in identifying corrupted workers. Particularly, we adopt the proposed countermeasure to detect corrupted workers, where the attack behaviors are hidden in perturbation with privacy budgets $\varepsilon = 1.0$, $\varepsilon = 1.6$, $\varepsilon = 2.2$, and $\varepsilon = 2.8$, respectively. In the Duchenne smile dataset, the AUC value is $0.9 \gg 0.5$ when privacy budget $\varepsilon = 2.8$, and even AUC $= 0.73 \gg 0.5$ when ε is reduced to 1.0. These results demonstrate the proposed countermeasure is an effective classifier.

As depicted in Fig. 9, in the Duchenne smile dataset, the AUC values decrease sharply when parameters ψ (the percentage of known workers and defined in Definition 4) increases from 0.15 to 0.45, and saturates afterward. Additionally, increasing ψ from 0.15 to 0.90 leads to around 0.03% performance drop in the Duchenne smile dataset. Similarly, as shown in Fig. 10, the AUC values decrease with increasing parameter φ (the percentage of corrupted workers and defined in Definition 3). In the Duchenne smile dataset, the AUC values decreased by about 13% when φ is increased from 0.05 to 0.30, but AUC $\gg 0.5$ when $\varphi = 0.30$. As ψ and φ increase, the slight decline of AUC is attributed to the attackers becoming more skilled at concealing themselves (refer to Figs. 5 and 6 for details on the attackers' disguise ability), making identification more challenging. In summary, Figs. 9 and 10 show that the proposed countermeasure is robust (the AUC only slightly declines) to the parameters ψ and φ.



(a) Impact of φ on AUC (b) Impact of φ on AUC

Fig. 10. Effectiveness of the proposed countermeasure (measured by AUC) varying with parameter φ (the percentage of corrupted workers). (a) AUC in the Duchenne smile dataset and (b) AUC in the weather sentiment dataset.

TABLE VI
COMPUTATIONAL COMPLEXITY OF DEFENSES

Defense Strategy	Computational Complexity
MWA	$\mathcal{O}(\lvert O\rvert \cdot \lvert U\rvert \cdot \log \lvert U\rvert)$
Our	$\mathcal{O}(gen_{max} \cdot SP \cdot \lvert U\rvert \cdot (\log \lvert U\rvert + \lvert O\rvert))$

As shown in Table VI, we also conducted a comparison of the computational complexity between the proposed defenses and the comparison. Particularly, the computational complexity of the proposed defenses is slightly higher than that of MWA when the lower number of participating workers or objects, i.e., $|O| \ll gen_{max} \cdot SP$ or $\log |U| \ll gen_{max} \cdot SP$. Nevertheless, with the growth in the number of participating workers and objects, the disadvantage of computational complexity becomes negligible. This is because the computational complexity of the proposed defenses is approximately equal to that of MWA when $|O| \gg |U| \gg gen_{max} \cdot SP$ or $\log |U| \gg |O| \gg gen_{max} \cdot SP$. That is $\mathcal{O}(|O| \cdot |U| \cdot \log(|U|)) \approx \mathcal{O}(|O|)$ and $\mathcal{O}(gen_{max} \cdot SP \cdot |U| \cdot (\log |U| + |O|)) \approx \mathcal{O}(|O|)$ when $|O| \gg |U| \gg gen_{max} \cdot SP$, and $\mathcal{O}(|O| \cdot |U| \cdot \log(|U|)) \approx \mathcal{O}(|U| \cdot \log |U|)$ and $\mathcal{O}(gen_{max} \cdot SP \cdot |U| \cdot (\log |U| + |O|)) \approx \mathcal{O}(|U| \cdot \log |U|)$ when $\log |U| \gg |O| \gg gen_{max} \cdot SP$.

VIII. CONCLUSION

In this paper, we have proposed the optimal data poisoning attacks and defenses on LDP-based privacy-preserving crowdsensing. The attacks can disrupt the aggregated results of the truth discovery methods by exploiting LDP perturbation, and the defense methods can effectively counter these attacks. Importantly, the proposed attacks expose significant consequences of LDP misuse, which may not only occur in crowdsensing applications but also in other LDP-based applications such as location services. The proposed defenses can be adapted to these applications with minor modifications to defend against similar attacks. For the future work, we will explore stronger data poisoning attacks that exploit personalized noise instead of general noises and design more robust personalized privacy protection solutions.

ACKNOWLEDGMENT

The authors are very grateful to Dr. Dongxiao Liu at the University of Waterloo for valuable suggestions on the article.

REFERENCES

[1] S. Jiang, J. Liu, Y. Zhou, and Y. Fang, "FVC-Dedup: A secure report deduplication scheme in a fog-assisted vehicular crowdsensing system," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 4, pp. 2727–2740, Aug. 2022.

[2] L. Hu, Y. Qian, J. Chen, X. Shi, J. Zhang, and S. Mao, "Photo crowdsourcing based privacy-protected healthcare," *IEEE Trans. Sustain. Comput.*, vol. 4, no. 2, pp. 168–177, Second Quarter, 2019.

[3] Z. Wang et al., "Towards privacy-driven truthful incentives for mobile crowdsensing under untrusted platform," *IEEE Trans. Mobile Comput.*, vol. 22, no. 2, pp. 1198–1212, Feb. 2023.

[4] C. Cai, Y. Zheng, A. Zhou, and C. Wang, "Building a secure knowledge marketplace over crowdsensed data streams," *IEEE Trans. Dependable Secure Comput.*, vol. 18, no. 6, pp. 2601–2616, Nov./Dec. 2021.

[5] Q. Li, Y. Li, J. Gao, B. Zhao, W. Fan, and J. Han, "Resolving conflicts in heterogeneous data by truth discovery and source reliability estimation," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2014, pp. 1187–1198.

[6] C. Miao, Q. Li, L. Su, M. Huai, W. Jiang, and J. Gao, "Attack under disguise: An intelligent data poisoning attack mechanism in crowdsourcing," in *Proc. World Wide Web Conf.*, 2018, pp. 13–22.

[7] C. Miao, Q. Li, H. Xiao, W. Jiang, M. Huai, and L. Su, "Towards data poisoning attacks in crowd sensing systems," in *Proc. 18th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2018, pp. 111–120.

[8] M. Fang, M. Sun, Q. Li, N. Z. Gong, J. Tian, and J. Liu, "Data poisoning attacks and defenses to crowdsourcing systems," in *Proc. Web Conf.*, 2021, pp. 969–980.

[9] Z. Li, Z. Zheng, S. Guo, B. Guo, F. Xiao, and K. Ren, "Disguised as privacy: Data poisoning attacks against differentially private crowdsensing systems," *IEEE Trans. Mobile Comput.*, vol. 22, no. 9, pp. 5155–5169, Sep. 2023.

[10] Y. Li et al., "Towards differentially private truth discovery for crowd sensing systems," in *Proc. IEEE 40th Int. Conf. Distrib. Comput. Syst.*, 2020, pp. 1156–1166.

[11] C. Huang, D. Liu, A. Yang, R. Lu, and X. Shen, "Multi-client secure and efficient DPF-based keyword search for cloud storage," *IEEE Trans. Dependable Secure Comput.*, vol. 21, no. 1, pp. 353–371, Jan./Feb. 2024.

[12] Z. Zheng, Z. Li, H. Jiang, L. Y. Zhang, and D. Tu, "Semantic-aware privacy-preserving online location trajectory data sharing," *IEEE Trans. Inf. Forensics Secur.*, vol. 17, pp. 2256–2271, 2022, doi: 10.1109/TIFS.2022.3181855.

[13] C. Huang et al., "Blockchain-assisted transparent cross-domain authorization and authentication for smart city," *IEEE Internet of Things J.*, vol. 9, no. 18, pp. 17 194–17 209, Sep. 2022.

[14] D. Liu, H. Wu, C. Huang, J. Ni, and X. Shen, "Blockchain-based credential management for anonymous authentication in SAGVN," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 10, pp. 3104–3116, Oct. 2022.

[15] C. Cai, Y. Zheng, Y. Du, Z. Qin, and C. Wang, "Towards private, robust, and verifiable crowdsensing systems via public blockchains," *IEEE Trans. Dependable Secure Comput.*, vol. 18, no. 4, pp. 1893–1907, Jul./Aug. 2021.

[16] Z. Zheng, Z. Li, J. Li, H. Jiang, T. Li, and B. Guo, "Utility-aware and privacy-preserving trajectory synthesis model that resists social relationship privacy attacks," *ACM Trans. Intell. Syst. Technol.*, vol. 13, no. 3, May 2022, Art. no. 44.

[17] Y. Zheng, H. Duan, X. Yuan, and C. Wang, "Privacy-aware and efficient mobile crowdsensing with truth discovery," *IEEE Trans. Dependable Secure Comput.*, vol. 17, no. 1, pp. 121–133, Feb. 2020.

[18] Y. Li, H. Sun, and W. H. Wang, "Towards fair truth discovery from biased crowdsourced answers," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2020, pp. 599–607.

[19] Y. Li et al., "Conflicts to harmony: A framework for resolving conflicts in heterogeneous data by truth discovery," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 8, pp. 1986–1999, Aug. 2016.

[20] U. Erlingsson, V. Pihur, and A. Korolova, "RAPPOR: Randomized aggregatable privacy-preserving ordinal response," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2014, pp. 1054–1067.

[21] S. P. Kasiviswanathan, H. K. Lee, K. Nissim, S. Raskhodnikova, and A. Smith, "What can we learn privately?," in *Proc. IEEE 49th Annu. Symp. Found. Comput. Sci.*, 2008, pp. 531–540.

[22] J. C. Duchi, M. I. Jordan, and M. J. Wainwright, "Local privacy and statistical minimax rates," in *Proc. IEEE 54th Annu. Symp. Found. Comput. Sci.*, 2013, pp. 429–438.

[23] C. Zhang, L. Zhu, C. Xu, X. Liu, and K. Sharif, "Reliable and privacy-preserving truth discovery for mobile crowdsensing systems," *IEEE Trans. Dependable Secure Comput.*, vol. 18, no. 3, pp. 1245–1260, May/Jun. 2021.

[24] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Proc. Theory Cryptogr. Conf.*, 2006, pp. 265–284.

[25] X. Cao, J. Jia, and N. Z. Gong, "Data poisoning attacks to local differential privacy protocols," in *Proc. 30th USENIX Secur. Symp.*, 2021, pp. 947–964.

[26] A. Cheu, A. Smith, and J. Ullman, "Manipulation attacks in local differential privacy," in *Proc. IEEE Symp. Secur. Privacy*, 2021, pp. 883–900.

[27] Y. Wu, X. Cao, J. Jia, and N. Z. Gong, "Poisoning attacks to local differential privacy protocols for key-value data," in *Proc. 31st USENIX Secur. Symp.*, 2022, pp. 519–536.

[28] Y. Li et al., "An efficient two-layer mechanism for privacy-preserving truth discovery," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2018, pp. 1705–1714.

[29] W. Lin, B. Li, and C. Wang, "Towards private learning on decentralized graphs with local differential privacy," *IEEE Trans. Inf. Forensics Secur.*, vol. 17, pp. 2936–2946, 2022, doi: 10.1109/TIFS.2022.3198283.

[30] W. Zhang, M. Li, R. Tandon, and H. Li, "Online location trace privacy: An information theoretic approach," *IEEE Trans. Inf. Forensics Secur.*, vol. 14, no. 1, pp. 235–250, Jan. 2019.

[31] G. Xu et al., "Catch you if you deceive me: Verifiable and privacy-aware truth discovery in crowdsensing systems," in *Proc. 15th ACM Asia Conf. Comput. Commun. Secur.*, 2020, pp. 178–192.

[32] X. Pang, Z. Wang, D. Liu, J. C. S. Lui, Q. Wang, and J. Ren, "Towards personalized privacy-preserving truth discovery over crowdsourced data streams," *IEEE/ACM Trans. Netw.*, vol. 30, no. 1, pp. 327–340, Feb. 2022.

[33] P. Sun et al., "Towards personalized privacy-preserving incentive for truth discovery in mobile crowdsensing systems," *IEEE Trans. Mobile Comput.*, vol. 21, no. 1, pp. 352–365, Jan. 2022.

[34] Z. Huang, M. Pan, and Y. Gong, "Robust truth discovery against data poisoning in mobile crowdsensing," in *Proc. IEEE Glob. Commun. Conf.*, 2019, pp. 1–6.

[35] S. L. Warner, "Randomized response: A survey technique for eliminating evasive answer bias," *J. Amer. Statist. Assoc.*, vol. 60, no. 309, pp. 63–69, 1965.

[36] K. Thomas, D. McCoy, C. Grier, A. Kolcz, and V. Paxson, "Trafficking fraudulent accounts: The role of the underground market in twitter spam and abuse," in *Proc. 22nd USENIX Secur. Symp.*, 2013, pp. 195–210.

[37] J. Giraldo, A. A. Cárdenas, M. Kantarcioglu, and J. Katz, "Adversarial classification under differential privacy," in *Proc. 27th Annu. Netw. Distrib. Syst. Secur. Symp.*, 2020, pp. 1–18.

[38] J. Nie, L. Wang, and J. J. Ye, "Bilevel polynomial programs and semidefinite relaxation methods," *SIAM J. Optim.*, vol. 27, no. 3, pp. 1728–1757, 2017.

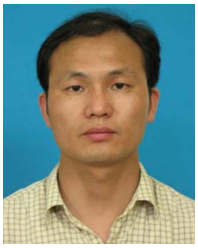
[39] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 398–417, Apr. 2009.

[40] R. Storn and K. Price, "Differential evolution-A simple and efficient heuristic for global optimization over continuous spaces," *J. Glob. Optim.*, vol. 11, no. 4, 1997, Art. no. 341.

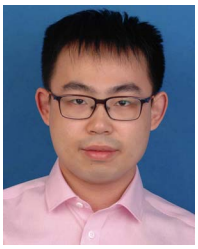
[41] M. Venanzi, W. Teacy, A. Rogers, and N. Jennings, "Weather sentiment - Amazon mechanical turk dataset," 2015. [Online]. Available: https://eprints.soton.ac.uk/376543/

[42] J. Whitehill, T.-F. Wu, J. Bergsma, J. Movellan, and P. Ruvolo, "Whose vote should count more: Optimal integration of labels from labelers of unknown expertise," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2009, pp. 2035–2043.

Zhirun Zheng received the BSc degree from the School of Mathematics and Statistics, Henan University of Science and Technology, China, in 2017. He is currently working toward the PhD degree with the School of Mathematics and Computational Science, Xiangtan University, China. In addition, he is currently a visiting PhD student with the University of Waterloo, Canada. His research interests include the areas of network security, data privacy, and mobile computing.

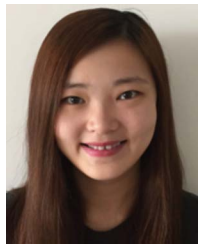


Zhetao Li (Member, IEEE) received the BEng degree in electrical information engineering from Xiangtan University, in 2002, the MEng degree in pattern recognition and intelligent systems from Beihang University, in 2005, and the PhD degree in computer application technology from Hunan University, in 2010. He is a professor with the College of Information Science and Technology, Jinan University. From 2013 to 2014, he was a postdoc in wireless networks with Stony Brook University. He is a member of the CCF.



Cheng Huang (Member, IEEE) received the PhD degree in electrical and computer engineering from the University of Waterloo, ON, Canada, in 2020. He is currently a postdoctoral research fellow with the Department of Electrical and Computer Engineering, University of Waterloo. His research interests include the areas of security and privacy in mobile networks, databases, and blockchain.

Saiqin Long received the PhD degree in computer applications technology from the South China University of Technology, Guangzhou, China, in 2014. She is currently a professor with the College of Information Science and Technology, Jinan University, China. Her research interests include cloud computing, edge computing, parallel and distributed systems, and Internet of Things. She has published more than 20 refereed papers in these areas, most of which are published in premium conferences and journals, including the *IEEE Transactions on Services Computing*, *IEEE Transactions on Parallel and Distributed Systems*, *IEEE Transactions on Mobile Computing*, etc. She is a member of the Chinese Computer Federation (CCF).



Mushu Li (Member, IEEE) received the MASc degree from Toronto Metropolitan University, Canada, in 2017, and the PhD degree in electrical and computer engineering from the University of Waterloo, Canada, in 2021. She is currently a postdoctoral fellow with Toronto Metropolitan University, ON, Canada. She was a postdoctoral fellow with the University of Waterloo, ON, Canada, from 2021 to 2022. Her research interests include mobile edge computing, the system optimization in wireless networks, and machine learning-assisted network management. She was the recipient of Natural Science and Engineering Research Council of Canada (NSERC) Postdoctoral Fellowship (2022), NSERC Canada Graduate Scholarship (2018), and Ontario Graduate Scholarship (2015 and 2016).

Xuemin (Sherman) Shen (Fellow, IEEE) received the PhD degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990. He is a University professor with the Department of Electrical and Computer Engineering, University of Waterloo, Canada. His research focuses on network resource management, wireless network security, Internet of Things, 5 G and beyond, and vehicular ad hoc and sensor networks. He is a registered professional engineer of Ontario, Canada, an Engineering Institute of Canada fellow, a Canadian Academy of Engineering fellow, a Royal Society of Canada fellow, a Chinese Academy of Engineering foreign member, and a distinguished lecturer of the IEEE Vehicular Technology Society and Communications Society. He received the Canadian Award for Telecommunications Research from the Canadian Society of Information Theory (CSIT) in 2021, the R.A. Fessenden Award in 2019 from IEEE, Canada, Award of Merit from the Federation of Chinese Canadian Professionals (Ontario) in 2019, James Evans Avant Garde Award in 2018 from the IEEE Vehicular Technology Society, Joseph LoCicero Award in 2015 and Education Award in 2017 from the IEEE Communications Society, and Technical Recognition Award from Wireless Communications Technical Committee (2019) and AHSN Technical Committee (2013). He has also received the Excellent Graduate Supervision Award in 2006 from the University of Waterloo and the Premier's Research Excellence Award (PREA) in 2003 from the Province of Ontario, Canada. He served as the Technical Program Committee chair/co-chair for IEEE Globecom' 16, IEEE Infocom'14, IEEE VTC'10 Fall, IEEE Globecom'07, and the chair for the IEEE Communications Society Technical Committee on Wireless Communications. He is the president of the IEEE Communications Society. He was the vice president for Technical & Educational Activities, vice president for Publications, member-at-large on the Board of Governors, chair of the Distinguished Lecturer Selection Committee, member of IEEE Fellow Selection Committee of the ComSoc. He served as an editor-in-chief of the *IEEE Internet of Things Journal*, *IEEE Network*, and *IET Communications*.