# Privacy-Preserving Truth Discovery for Sparse Data in Mobile Crowdsensing Systems

Feng Liu, Bin Zhu, Shaoxian Yuan, Jian Li, Kaiping Xue†
School of Cyber Security, University of Science and Technology of China, Hefei, Anhui 230027, China
†Corresponding author, kpxue@ustc.edu.cn

GLOBECOM 2021 - 2021 IEEE Global Communications Conference | 978-1-7281-8104-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/GLOBECOM46510.2021.9685134

***Abstract*—Truth discovery is an effective method to infer truthful information from a large amount of sensory data in mobile crowdsensing systems. Privacy-preserving truth discovery schemes require the cloud server not to access each worker's sensory data directly so that the privacy of sensory data can be preserved. In some specific applications such as sparse mobile crowdsensing, workers can only contribute sensory data on a small part of sensing tasks, implying that the information of which tasks are completed by a worker should also be preserved. However, existing privacy-preserving truth discovery schemes do not consider such sparse data scenarios in mobile crowdsensing systems. In this paper, we first identify the privacy issues in truth discovery when sensory data are sparse. To address these issues, we design a privacy-preserving truth discovery scheme by employing the additively homomorphic cryptosystem and additive secret sharing with two non-colluding servers. Through detailed analysis and extensive experiments, we demonstrate that our proposed scheme can satisfy strong privacy-preserving requirements with low computation and communication overhead.**

***Index Terms*—Truth Discovery, Privacy Preservation, Sparse Data, Mobile Crowdsensing**

## I. Introduction

Over the past few years, developments in the field of cloud computing and the Internet of Things have led to a growing interest in mobile crowdsensing systems. In a typical mobile crowdsensing system, the cloud server can analyze the sensory data provided by a number of mobile devices (usually referred to as *workers*, e.g., smartphones, wearables, and on-board computers). Since the quality of sensory data provided by different workers may vary greatly, how to derive the truthful information from sensory data of different qualities becomes a major obstacle. To address this challenge, an effective method called truth discovery [1], [2] is proposed, where the general principle is to calculate each worker's reliability degree (usually referred to as *weight*) before estimating the truths.

Although truth discovery can help cloud server to derive truthful information in an effective way, it poses threats to workers' privacy directly. Several researchers [3]–[8] pointed out that the data provided by workers are sensitive to some extent, and designed diverse privacy-preserving truth discovery schemes to assure workers' privacy. These works mainly focus on the privacy of workers' sensory data, and some of them (such as [6]–[8]) further aruge that the privacy of workers' weights should also be preserved. So far, these schemes can only be applied to the situation where workers have to execute all sensing tasks. However, due to the variety of task types and workers' abilities, requiring each worker to execute all sensing tasks seems unrealistic. Especially in the mobile crowdsensing systems, it is quite common that a worker only executes partial tasks for saving energy [9], [10]. Under these circumstances, the sensory data provided by workers probably are only concerned with partial sensing tasks, namely, the sensory data are sparse. To the best of our knowledge, there has been little discussion about privacy-preserving truth discovery in such a sparse data environment.

In this paper, we focus on the privacy issues of sparse data for truth discovery and propose a privacy-preserving truth discovery scheme to handle these issues. In particular, our contributions are summarized in the following:

- We identify the privacy requirements in the situation of data sparsity for truth discovery and design an efficient privacy-preserving truth discovery scheme.
- By employing additively homomorphic cryptosystem and additive secret sharing with two non-colluding servers, our proposed scheme can provide strong privacy preservation for workers in an efficient way, where workers do not need to calculate heavy cryptographic operations and participate in the iteration phase.
- We conduct extensive experiments to evaluate the performance of the proposed scheme. The results also demonstrate that our design is efficient in terms of computation and communication overhead.

The remainder of this paper is organized as follows. Section II introduces the related work. The problem statement is given in Section III. In Section IV, we briefly describe the truth discovery algorithm, additively homomorphic cryptosystem, and additive secret sharing adopted in this work. Section V illustrates our proposed scheme in detail. After that, system analysis and performance evaluation are provided in Section VI and Section VII, respectively. Finally, Section VIII concludes this paper.

## II. Related Work

Recently, several studies have attempted to employ various methods such as cryptographic tools and data perturbation techniques to assure privacy preservation for truth discovery [3]–[8]. In general, these schemes can be categorized into the single-server setting and two-server setting.

In the single-server-based schemes [3], [4], workers have to participate in the iterative processes for weight update, which implies that they would suffer from additional computing

978-1-7281-8104-2/21/$31.00 ©2021 IEEE

and communication overhead. To reduce the interactions of workers, Miao *et al*. [5] proposed $L$-PPTD and $L^2$-PPTD by involving two non-colluding servers. After that, more and more schemes [6]–[8] are designed under the two-server setting. Overall, no matter the single-server-based schemes or two-server-based schemes, all the previously mentioned methods are based on the **C**onflict **R**esolution on **H**eterogeneous Data (CRH) algorithm [1], which is a representative truth discovery algorithm in recent years. Due to the limitation that each worker has to provide sensory data of all sensing tasks in CRH algorithm, the CRH-based privacy-preserving truth discovery schemes fail to handle the sparse sensory data.

Fortunately, there is another truth discovery algorithm called **C**onfidence-**A**ware **T**ruth **D**iscovery (CATD) [2] for handling sparse data. Zheng *et al*. [11] designed an encrypted truth discovery based on CATD with two non-colluding servers, where all procedures are conducted in the encrypted domain. Their proposed scheme can realize strong privacy protection for workers' sensory data and weights by adopting Garbled Circuit (GC). However, only achieving privacy preservation of sensory data and weights is not enough in sparse data situation, because the information of which tasks are completed by a worker is also sensitive. Taking the sparse mobile crowdsensing [12] as an example, the sensing tasks are distributed in different locations of a spatial area. So that workers cannot execute all tasks within a short period, which leads to sparse sensory data. This research pointed out that the information of which tasks are executed by a worker can disclose the worker's location privacy. Even though, their work only focuses on the privacy requirements in sparse mobile crowdsensing rather than designing the privacy-preserving truth discovery scheme. To the best of our knowledge, there still lacks discussions about the privacy issues for truth discovery in the sparse data scenarios.

TABLE I
COMPARISON WITH VARIOUS SCHEMES

| Schemes | Sensory Data Privacy | Weight Privacy | Sparse Data Support | Indicator Privacy |
|---|---|---|---|---|
| $L$-PPTD [5] | √ | √ | × | - |
| $L^2$-PPTD [5] | √ | × | × | - |
| EPTD [4] | √ | √ | × | - |
| RPTD-I [6] | √ | √ | × | - |
| RPTD-II [6] | √ | √ | × | - |
| InPPTD [7] | √ | √ | × | - |
| Encrypted CATD [11] | √ | √ | √ | × |
| Our Propsed Scheme | √ | √ | √ | √ |

As listed in TABLE I, we present the comparison between our proposed scheme with several state-of-the-art privacy-preserving truth discovery schemes. The indicator privacy means the privacy of which tasks are executed by a worker, since we use an indicator to mark which tasks are completed by a worker in this work. Only the encrypted CATD and our proposed scheme can work in the sparse data scenarios, but the encrypted CATD cannot guarantee the indicator privacy.

## III. PROBLEM STATEMENT

### A. System Model

In this paper, we call the sensing tasks as objects for simplicity. As shown in Fig. 1, the participants are consisted of a number of workers and two servers. In brief, workers are responsible for collecting sensory data of different sensing objects. Since most mobile devices are resource-limited, it is not necessary for each worker to collect sensory data of all sensing objects. To assure privacy preservation, workers need to perturb their data before uploading. Upon receiving the perturbed sensory data uploaded by workers, servers start to execute the secure weight update and secure truth update iteratively. In general, we assume that the servers have sufficient computation and storage capabilities. These two servers are denoted by $S_0$ and $S_1$, respectively.

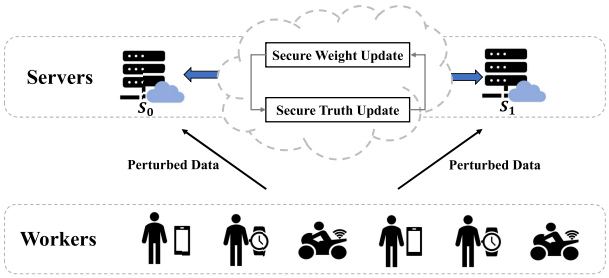



Fig. 1. System Model

Assume that there are $M$ objects to be sensed, and the number of workers is $K$. We denote $x_m^k$ as the sensory data of object $m$ provided by worker $k$. Due to the sparsity of sensory data $\{x_m^k\}_{m=1}^M$, we use an indicator vector $[\phi_1^k, \ldots, \phi_M^k]$ to mark the missing sensory objects where $\phi_m^k = 1$ means worker $k$ has collected the sensory data of object $m$, and $\phi_m^k = 0$ otherwise. Note that if $\phi_m^k = 0$, then the corresponding $x_m^k = 0$. Besides, we use $w_k$ and $x_m^*$ to denote the weight for worker $k$ and the estimated truth for object $m$, receptively.

### B. Design Goals

The main goal of our proposed scheme is to design a privacy-preserving truth discovery scheme in the situation where sensory data are sparse. Inspired by [12], the privacy of indicator vector for each worker also needs to be considered for sparse data. Therefore, the main goal is to protect workers' privacy including sensory data, indicator vector, and weights simultaneously. In addition, considering that most workers are resource-limited, it is also necessary to reduce the computation overhead and communication overhead on the workers' side.

Note that the lazy workers are not involved, because this issue can be solved by integrating incentive mechanism [7]. In addition, the issues of inferring missing data from historical data records are challenging and not considered in this work.

### C. Threat Model

In the proposed scheme, we assume that all entities are semi-honest, which means that both workers and servers will

honestly execute the protocols, but they will also try to infer private information about participants. For each participant in this system, we assume that there always exist secure channels among workers, $S_0$, and $S_1$. Besides, similar to other two-sever-based schemes [6], [13], the two servers are assumed not to collude with each other in this system.

## IV. Preliminaries

### A. CATD

In general, the truth discovery algorithm CATD [2] consists of two parts: weight update and truth update.

*1) Weight Update:* Given the truthful information $\{x_m^*\}_{m=1}^M$, the weight for each worker $k$ is computed as

$$w_k = \frac{\chi^2_{(1-\alpha/2,\sum_{m=1}^M \phi_m^k)}}{\sum_{m=1}^{M} \phi_m^k (x_m^k - x_m^*)^2}.$$

Note that $\chi^2$ denotes the Chi-squared distribution, and the system-wide constant $\alpha$ is known as the significance level which is usually a small number such as 0.05.

*2) Truth Update:* Given the weight $w_k$ for each worker $k$, the truth for each object $m$ is computed as

$$x_m^* = \sum_{k=1}^{K} w_k x_m^k / \sum_{k=1}^{K} \phi_m^k w_k.$$

In real practice, weight update and truth update should be executed iteratively to achieve more accurate results until the convergence criteria are met.

### B. Additively Homomorphic Cryptosystem

Let $\mathcal{M}$ denote the message space, $\mathcal{C}$ denote the ciphertext space, an additive homomorphic cryptosystem consists of the following four probabilistic poly-time algorithms.

- $\mathsf{Setup}(1^\kappa) \rightarrow pp$: Taken the input of security parameter $\kappa$, the algorithm returns the public parameter $pp$. Unless otherwise stated, $pp$ is implicitly fed in the following algorithms.
- $\mathsf{KeyGen}(1^\kappa) \rightarrow (pk, sk)$: Taken the input of security parameter $\kappa$, the algorithm returns the public key $pk$ and private key $sk$.
- $\mathsf{Enc}_{pk}(m) \rightarrow c$: Given the message $m \in \mathcal{M}$, the encryption algorithm outputs $c$ which is the ciphertext of message $m$.
- $\mathsf{Dec}_{sk}(c) \rightarrow m$: Given the ciphertext $c$, the decryption algorithm outputs the corresponding plaintext $m$.

We claim that the above public-key cryptosystem is additively homomorphic if it satisfies the following properties for some operators $\oplus$ and $\otimes$ in probabilistic polynomial time.

- $\forall m_1, m_2 \in \mathcal{M}$, given two ciphertexts $c_1 = \mathsf{Enc}_{pk}(m_1)$ and $c_2 = \mathsf{Enc}_{pk}(m_2)$, it holds that $\mathsf{Dec}_{sk}(c_1 \oplus c_2) = m_1 + m_2$.
- $\forall m \in \mathcal{M}$, given a constant $a$ and a ciphertext $c = \mathsf{Enc}_{pk}(m)$, it holds that $\mathsf{Dec}_{sk}(a \otimes c) = a \times m$.

### C. Additive Secret Sharing

Secret sharing is one of the most important tools in secure multi-party computation. The additive secret sharing means that a secret $x$ can be randomly split into $n$ shares, for example $x = x_0 + x_1 + \ldots, x_{n-1}$. The only approach to recover the secret is to collect all $n$ shares. In this paper, we just pay attention to the situation where $n = 2$. That is, the data owner can randomly split a secret data $x$ into two additive shares $x_0$ and $x_1$, namely, $x = x_0 + x_1$. For each server $S_i$ $(i \in \{0, 1\})$ owns the share $x_i$, the secret $x$ cannot be recovered without the help of $S_{1-i}$. In the rest of the paper, we use a share to represent an additive share for simplicity.

## V. The Proposed Scheme

### A. Overview

As we mentioned above, our privacy-preserving truth discovery scheme is designed for scenarios where the sensory data provided by workers are sparse. In the proposed scheme, workers need to upload indicator vectors to mark which objects they observed. To satisfy the privacy requirements, it is necessary to protect the privacy of indicator vectors, sensory data, and workers' weights in the same time.

For the sake of illustration, we choose a widely used additively homomorphic cryptosystem called the Paillier cryptosystem [14] to formulate the homomorphic operations. In brief, $\forall m_1, m_2, m \in \mathcal{M}$, given the corresponding ciphertexts $c_1, c_2, c$ and a constant $a$, it always holds that

$$\mathsf{Dec}_{sk}(c_1 \times c_2) = m_1 + m_2, \mathsf{Dec}_{sk}(c^a) = a \times m.$$

Note that the above Paillier cryptosystem only works over integers, while the sensory data in our scheme may be floating-point numbers. To handle this situation, a typical approach is to round the floating-point numbers by multiplying a factor $L$, and the final results can be recovered by dividing $L$ [7], [11].

To reduce the complicated operations on the workers' side, we shift all heavy cryptographic operations to the servers' side. In other words, workers only need to generate shares and upload them to two servers respectively. After that, the truths can be estimated by two servers. The whole procedure can be divided into 5 phases: *Initialization*, *Report*, *Pre-Processing*, *Secure Weight Update* and *Secure Truth Update*.

### B. Initialization Phase

In this phase, $S_0$ first generates an asymmetric key pair $(pk, sk)$ of the additively homomorphic cryptosystem by invoking $\mathsf{KeyGen}(\cdot)$. Then $S_0$ sets a small number for the significance level $\alpha$ ($\alpha$ is set to 0.05 by default) and randomly generates the initial truths $\{x_m^*\}_{m=1}^M$. Finally, $S_0$ publishes the public key $pk$ along with the significance level $\alpha$, and sends the initial truths to $S_1$.

### C. Report Phase

In this phase, each worker collects sensory data for distinct objects. Taking worker $k$ as an example, after obtaining a set of sensory data $\{x_m^k\}_{m=1}^M$ and generating an indicator vector

$\{\phi_m^k\}_{m=1}^M$, worker $k$ computes $y_k = \chi^2_{(1-\alpha/2,\sum_{m=1}^M \phi_m^k)}$. Then for each object $m$, worker $k$ computes $\tilde{\phi}_m^k = \phi_m^k / y_k$. After that, worker $k$ computes the shares of $x_m^k$, $\phi_m^k$ and $\tilde{\phi}_m^k$. Namely, $x_m^k = x_{m,0}^k + x_{m,1}^k$, $\phi_m^k = \phi_{m,0}^k + \phi_{m,1}^k$, $\tilde{\phi}_m^k = \tilde{\phi}_{m,0}^k + \tilde{\phi}_{m,1}^k$. Finally worker $k$ uploads $\{x_{m,0}^k, \phi_{m,0}^k, \tilde{\phi}_{m,0}^k\}_{m=1}^M$ to $S_0$, $\{x_{m,1}^k, \phi_{m,1}^k, \tilde{\phi}_{m,1}^k\}_{m=1}^M$ to $S_1$. When the uploading process is complete, worker $k$ can go offline.

### D. Pre-Processing Phase

We emphasize that this phase is executed by $S_0$. For worker $k$, $S_0$ first computes $C_0^k$ as:

$$C_0^k = \mathsf{Enc}_{pk}\left[\sum_{m=1}^M \tilde{\phi}_{m,0}^k \left(x_{m,0}^k\right)^2\right].$$

Then $S_0$ encrypts $\{x_{m,0}^k, (x_{m,0}^k)^2, \tilde{\phi}_{m,0}^k, \tilde{\phi}_{m,0}^k \cdot x_{m,0}^k\}_{m,k=1}^{M,K}$ respectively. These ciphertexts are typically denoted by $C_{pack1}$. Finally, $S_0$ sends $\{C_0^k\}_{k=1}^K$ and $C_{pack1}$ to $S_1$.

### E. Secure Weight Update Phase

In this phase, upon receiving the ciphertexts, $S_1$ computes $C_1^k$,$C_2^k$,$C_3^k$,$C_4^k$ as follows.

$$\begin{cases} C_1^k = \prod_{m=1}^M \left[\mathsf{Enc}_{pk}\left[\left(x_{m,0}^k\right)^2\right]^{\tilde{\phi}_{m,1}^k}\right], \\ C_2^k = \prod_{m=1}^M \left[\left[\mathsf{Enc}_{pk}\left(x_{m,0}^k\right)\right]^{2\left(x_{m,1}^k - x_m^*\right)\tilde{\phi}_{m,1}^k}\right] \times \\ \qquad \prod_{m=1}^M \left[\left[\mathsf{Enc}_{pk}\left(\tilde{\phi}_{m,0}^k x_{m,0}^k\right)\right]^{2\left(x_{m,1}^k - x_m^*\right)}\right], \\ C_3^k = \prod_{m=1}^M \left[\mathsf{Enc}_{pk}\left(\tilde{\phi}_{m,0}^k\right)^{\left(x_{m,1}^k - x_m^*\right)^2}\right], \\ C_4^k = \mathsf{Enc}_{pk}\left[\sum_{m=1}^M \tilde{\phi}_{m,1}^k \left(x_{m,1}^k - x_m^*\right)^2\right]. \end{cases}$$

To preserve the privacy of workers' weights, $S_1$ chooses a random value $b_k$ and computes the ciphertexts $C_k$ as

$$C_k = \left(C_0^k \cdot C_1^k \cdot C_2^k \cdot C_3^k \cdot C_4^k\right)^{b_k}.$$

Finally $S_1$ sends $\{C_k\}_{k=1}^K$ to $S_0$. After receiving $\{C_k\}_{k=1}^K$, $S_0$ decrypts $C_k$ with its private key and obtains the perturbed weight $\tilde{w}_k$ of worker $k$ as follows,

$$\tilde{w}_k = \frac{1}{\mathsf{Dec}_{sk}\left(C_k\right)} = \frac{w_k}{b_k}. \tag{1}$$

### F. Secure Truth Update Phase

In the truth update phase, $S_0$ first encrypts $\{\tilde{w}_k\}_{k=1}^K$ and $\{\tilde{w}_k \cdot x_{m,0}^k, \tilde{w}_k \cdot \phi_{m,0}^k\}_{m,k=1}^{M,K}$ respectively. The sets of ciphertexts are denoted by $C_{pack2}$. Finally $S_0$ sends $C_{pack2}$ to $S_1$.

After receiving the ciphertexts, $S_1$ computes $C_5^m$ and $C_6^m$ as follows,

$$\begin{aligned} C_5^m &= \prod_{k=1}^K \left[\left[\mathsf{Enc}_{pk}\left(\tilde{w}_k \cdot x_{m,0}^k\right) \cdot \left[\mathsf{Enc}_{pk}\left(\tilde{w}_k\right)\right]^{x_{m,1}^k}\right]^{b_k}\right] \\ &= \mathsf{Enc}_{pk}\left[\sum_{k=1}^K w_k x_m^k\right], \\ C_6^m &= \prod_{k=1}^K \left[\left[\mathsf{Enc}_{pk}(\tilde{w}_k \cdot \phi_{m,0}^k) \cdot \left[\mathsf{Enc}_{pk}(\tilde{w}_k)\right]^{\phi_{m,1}^k}\right]^{b_k}\right] \\ &= \mathsf{Enc}_{pk}\left[\sum_{k=1}^K w_k \phi_m^k\right]. \end{aligned}$$

Finally, $S_1$ sends $\{C_5^m\}_{m=1}^M$ and $\{C_6^m\}_{m=1}^M$ to $S_0$. Then $S_0$ decrypts these ciphertexts and obtains the estimated truth as follows,

$$x_m^* = \frac{\mathsf{Dec}_{sk}(C_5^m)}{\mathsf{Dec}_{sk}(C_6^m)} = \sum_{k=1}^K w_k x_m^k / \sum_{k=1}^K w_k \phi_m^k. \tag{2}$$

## VI. System Analysis

### A. Correctness

Since the correctness of decryption can be guaranteed by the additively homomorphic cryptosystem, we only need to prove that the Eq. 1 and Eq. 2 hold. The correctness of Eq. 2 is straightforward, here we give the correctness analysis of Eq. 1. According to the properties of the additively homomorphic cryptosystem, we have

$$\begin{aligned} C_0^k C_1^k &= \mathsf{Enc}_{pk}\left[\sum_{m=1}^M \tilde{\phi}_{m,0}^k \left(x_{m,0}^k\right)^2 + \tilde{\phi}_{m,1}^k \left(x_{m,0}^k\right)^2\right] \\ &= \mathsf{Enc}_{pk}\left[\sum_{m=1}^M \tilde{\phi}_m^k \left(x_{m,0}^k\right)^2\right], \\ C_3^k C_4^k &= \mathsf{Enc}_{pk}\left[\sum_{m=1}^M \left(\tilde{\phi}_{m,0}^k + \tilde{\phi}_{m,1}^k\right)\left(x_{m,1}^k - x_m^*\right)^2\right] \\ &= \mathsf{Enc}_{pk}\left[\sum_{m=1}^M \tilde{\phi}_m^k \left(x_{m,1}^k - x_m^*\right)^2\right]. \end{aligned}$$

Since $\tilde{\phi}_m^k (x_{m,0}^k)^2 + 2\tilde{\phi}_m^k x_{m,0}^k \left(x_{m,1}^k - x_m^*\right) + \tilde{\phi}_m^k (x_{m,1}^k - x_m^*)^2 = \tilde{\phi}_m^k (x_{m,0}^k + x_{m,1}^k - x_m^*)^2$, then we can further obtain

$$\begin{aligned} C_k &= \left(C_0^k \cdot C_1^k \cdot C_2^k \cdot C_3^k \cdot C_4^k\right)^{b_k} \\ &= \left[\mathsf{Enc}_{pk}\left[\sum_{m=1}^M \tilde{\phi}_m^k (x_{m,0} + x_{m,1} - x_m^*)^2\right]\right]^{b_k} \\ &= \mathsf{Enc}_{pk}\left[b_k \sum_{m=1}^M \tilde{\phi}_m^k (x_m - x_m^*)^2\right]. \end{aligned}$$

Hence, the perturbed weight $\tilde{w}_k$ can be written as

$$\tilde{w}_k = \frac{1}{\mathsf{Dec}_{sk}\left(C_k\right)} = \frac{1}{b_k \sum_{m=1}^M \tilde{\phi}_m^k (x_m - x_m^*)^2} = \frac{w_k}{b_k}.$$

Therefore, the correctness of Eq. 1 holds.

### B. Security Discussion

Note that our design goal is to protect the privacy of sensory data, indicator vectors and weights for each worker simultaneously. According to the threat model in Section III-C, each entity is assumed to be semi-honest, and two servers are non-colluding. Since workers do not take part in the iteration phase, we only need to consider that the privacy of $x_m^k$, $\phi_m^k$ and $w_k$ are not disclosed from the views of $S_0$ and $S_1$.

For $S_0$, it only knows $\{x_{m,0}^k, \phi_{m,0}^k, \tilde{\phi}_{m,0}^k\}_{m=1}^M$, $C_k$, $C_5^m$ and $C_6^m$. $S_0$ can decrypt $C_k$, $C_5^m$, $C_6^m$, and further obtain the perturb weight $\tilde{w}_k$ along with the aggregation information such as $\sum_{k=1}^{K} w_k x_m^k$ and $\sum_{k=1}^{K} w_k \phi_m^k$. Since $x_{m,1}^k$ and $\phi_{m,1}^k$ are generated randomly from additive secret sharing by worker $k$, $b_k$ is chosen randomly by $S_1$. Without knowing $x_{m,1}^k$, $\phi_{m,1}^k$ and $b_k$, $S_0$ cannot infer $x_m^k$, $\phi_m^k$ and $w_k$ for worker $k$.

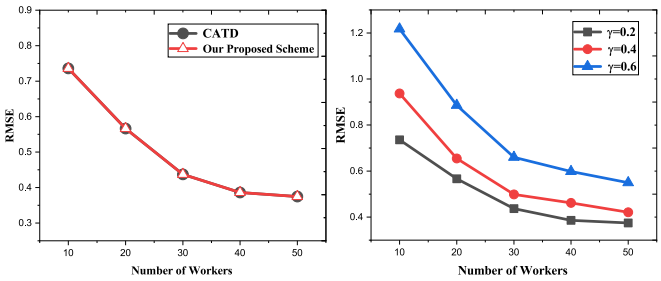
For $S_1$, it only knows $\{x_{m,1}^k, \phi_{m,1}^k, \tilde{\phi}_{m,1}^k\}_{m=1}^M$, $C_0^k$, $C_{pack1}$ and $C_{pack2}$. Similarly, since $x_{m,0}^k$ and $\phi_{m,0}^k$ are generated randomly from additive secret sharing by worker $k$, $S_1$ cannot infer $x_m^k$ and $\phi_m^k$ for worker $k$ without knowing $x_{m,0}^k$ and $\phi_{m,0}^k$. Note that $C_0^k$, $C_{pack1}$ and $C_{pack2}$ are ciphertexts under additively homomorphic cryptosystem, $S_1$ has no mechanism to decrypt them without the private key $sk$, and hence cannot learn worker's weight $w_k$ even it knows the ciphertext $C_k$.

## VII. Performance Evaluation

In this section, we evaluate the performance of the proposed scheme in terms of accuracy, convergence, computation overhead, and communication overhead. All procedures are conducted on a computer with an Intel Core i5-10400 CPU (4.30 GHz) and 16GB RAM running Linux operation system. We invoke the Python-Paillier library [15] for rapidly building advance cryptosystems (key size is set to 2048 bits by default). All experiments are executed 10 times and we take the average results. For the sake of illustration, we use $\gamma$ to denote the sparsity of sensory data. $\gamma$ is defined as $\gamma = 1-(\sum_{m=1}^{M}\sum_{k=1}^{K}\phi_m^k)/(M\cdot K)$. Hereafter, unless otherwise stated, the default sparsity is set to 0.2 by default.

### A. Accuracy

Since the CATD [2] is adopted as the fundamental truth discovery algorithm in our proposed scheme, we first measure the accuracy of estimated ground truth between our proposed scheme and the baseline algorithm. Similar to [6], [7], we use the Root of Mean Squared Error (RMSE) defined as $(\sum_{m=1}^{M}(x_m^* - \hat{x}_m)/M)^{\frac{1}{2}}$ to evaluate the deviation between the estimated truths $\{x_m^*\}_{m=1}^M$ and the ground truths $\{\hat{x}_m\}_{m=1}^M$. In this experiment, the number of objects is fixed as 20, the number of workers varies from 10 to 50. As shown in Fig. 2a, we observe that the proposed scheme achieves a similar accuracy to the baseline algorithm. Additionally, the estimated truths are closer to the real ground truths with the increase in the number of workers. To further analyze the impact of sparsity $\gamma$ for accuracy, we conduct the experiment




(a) Accuracy Comparison with CATD (b) The Impact of Different Sparsity

Fig. 2. Accuracy Evaluation

under different sparsity settings from 0.2 to 0.6. The results in Fig. 2b show that in a high sparsity situation, it is necessary to recruit more workers to get more accurate results.

### B. Convergence

As for convergence, we use $\sum_{m=1}^{M}(x_m^t - x_m^{t-1})^2$ as the convergence value in the $t$-th iteration, where $x_m^t$ is the estimated truth in $t$-th iteration and $x_m^0$ is initialized randomly. In this experiment, the number of workers and objects are fixed as 10 and 20 respectively. As illustrated in Fig. 3, the convergence speed is very fast at the first few iterations. Meanwhile, the sparsity has little impact on the convergence.

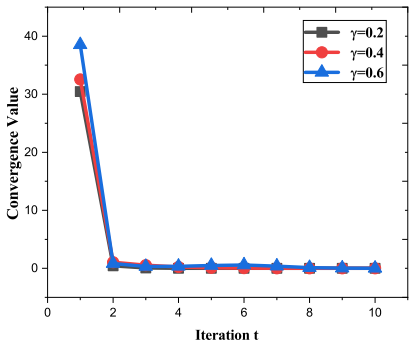



Fig. 3. Convergence Evaluation

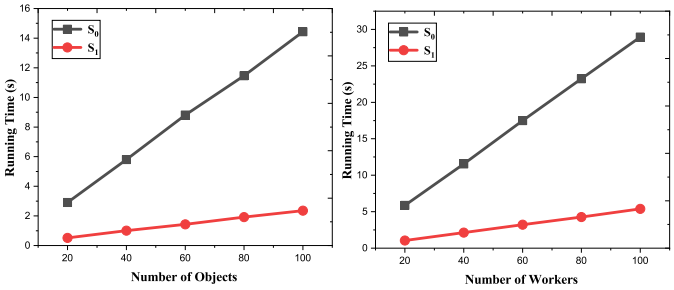
### C. Efficiency Evaluation

In this part, we measure the performance of computation overhead and communication overhead respectively. TABLE II lists the computation overhead of workers and $S_0$ in non-iteration phases including the report phase and pre-processing phase with the different number of objects and workers. We observe the computation cost of workers is negligible compared to $S_0$ since there are no heavy cryptographic operations on the workers' side.

As for the iteration phase, we measure the computation cost of $S_0$ and $S_1$ under different numbers of workers and objects for each iteration. As shown in Fig. 4, the computation cost for $S_1$ is much less compared to $S_0$ under some conditions. For the communication overhead, TABLE III reports the communication overhead for each entity in different

TABLE II
COMPUTATION OVERHEAD IN NON-ITERATION PHASES (S)

| Number of Workers and Objects | | Report Phase | Pre-Processing Phase |
|---|---|---|---|
| | | Workers | $S_0$ |
| $K = 10$ | $M = 20$ | 0.0011 | 6.93 |
| | $M = 40$ | 0.0011 | 20.98 |
| | $M = 60$ | 0.0012 | 34.86 |
| $M = 20$ | $K = 20$ | 0.0049 | 14.04 |
| | $K = 40$ | 0.0059 | 42.41 |
| | $K = 60$ | 0.0104 | 70.92 |




(a) $K = 10$ (b) $M = 20$

Fig. 4. Computation Overhead for Each Iteration

phases. It is clear that the communication overhead in the iteration phase is less than that in the non-iteration phase. Note that the alternative privacy-preserving truth discovery scheme [11] based on CATD is implemented by GC. However, the computation cost of GC generation and the communication of GC transmission are huge. We demonstrate that the computation overhead and communication overhead on the workers' side are lightweight, and the communication overhead between two servers is also acceptable for truth discovery in reality.

TABLE III
COMMUNICATION OVERHEAD (KB)

| Number of Workers and Objects | | Report Phase | Pre-Processing Phase | Iteration Phase | |
|---|---|---|---|---|---|
| | | Workers to Servers | $S_0$ to $S_1$ | $S_0$ to $S_1$ | $S_1$ to $S_0$ |
| $K = 10$ | $M = 20$ | 9.60 | 8.08 | 3.44 | 0.04 |
| | $M = 40$ | 19.20 | 16.08 | 6.80 | 0.72 |
| | $M = 60$ | 28.80 | 24.08 | 10.16 | 1.04 |
| $M = 20$ | $K = 20$ | 19.20 | 16.16 | 6.72 | 0.48 |
| | $K = 40$ | 38.40 | 32.32 | 13.28 | 0.64 |
| | $K = 60$ | 57.60 | 48.48 | 19.84 | 0.80 |

## VIII. CONCLUSION

In this paper, we proposed a privacy-preserving truth discovery scheme targeted for privacy-preserving problems in sparse data scenarios. Firstly, we carefully discussed the privacy issues for truth discovery when sensory data provided by workers are sparse. After that, we presented a scheme to guarantee these requirements by utilizing additively homomorphic cryptosystem based on the CATD framework. Finally, through the analysis of security, our proposed scheme achieves strong privacy protection of workers' sensory data, weights, and indicator vectors simultaneously. Extensive experiments also indicate that the computation overhead of workers and the communication overhead in the iteration phase are lightweight, which implies that the proposed scheme is practical in real mobile crowdsensing systems.

## ACKNOWLEDGMENT

This work is supported in part by the National Natural Science Foundation of China under Grant No. 61972371 and Youth Innovation Promotion Association of the Chinese Academy of Sciences (CAS) under Grant No. Y202093.

## REFERENCES

[1] 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 *Proceedings of the 2014 International Conference on Management of Data (SIGMOD)*. ACM, Jun. 2014, pp. 1187–1198.

[2] Q. Li, Y. Li, J. Gao, L. Su, B. Zhao, M. Demirbas, W. Fan, and J. Han, "A confidence-aware approach for truth discovery on long-tail data," *Proceedings of the VLDB Endowment*, vol. 8, no. 4, pp. 425–436, Dec. 2014.

[3] C. Miao, W. Jiang, L. Su, Y. Li, S. Guo, Z. Qin, H. Xiao, J. Gao, and K. Ren, "Cloud-enabled privacy-preserving truth discovery in crowd sensing systems," in *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems (SenSys)*. ACM, Nov. 2015, pp. 183–196.

[4] G. Xu, H. Li, S. Liu, M. Wen, and R. Lu, "Efficient and privacy-preserving truth discovery in mobile crowd sensing systems," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3854–3865, Jan. 2019.

[5] C. Miao, L. Su, W. Jiang, Y. Li, and M. Tian, "A lightweight privacy-preserving truth discovery framework for mobile crowd sensing systems," in *Proceedings of the 2017 IEEE Conference on Communications (Infocom)*. IEEE, May 2017, pp. 1–9.

[6] C. Zhang, L. Zhu, C. Xu, X. Liu, and K. Sharif, "Reliable and privacy-preserving truth discovery for mobile crowdsensing systems," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 3, pp. 1245–1260, May 2019.

[7] K. Xue, B. Zhu, Q. Yang, N. Gai, D. S.L. Wei, and N. Yu, "InPPTD: An lightweight incentive-based privacy-preserving truth discovery for crowd sensing systems," *IEEE Internet of Things Journal*, vol. 8, no. 6, pp. 4305–4316, Oct. 2020.

[8] J. Tang, S. Fu, X. Liu, Y. Luo, and M. Xu, "Achieving privacy-preserving and lightweight truth discovery in mobile crowdsensing," *IEEE Transactions on Knowledge and Data Engineering*, Early Access, 2021.

[9] L. Wang, D. Zhang, Y. Wang, C. Chen, X. Han, and A. M'hamed, "Sparse mobile crowdsensing: Challenges and opportunities," *IEEE Communications Magazine*, vol. 54, no. 7, pp. 161–167, Jul. 2016.

[10] J. Wang, Y. Wang, D. Zhang, and S. Helal, "Energy saving techniques in mobile crowd sensing: Current state and future opportunities," *IEEE Communications Magazine*, vol. 56, no. 5, pp. 164–169, May 2018.

[11] Y. Zheng, H. Duan, and C. Wang, "Learning the truth privately and confidently: Encrypted confidence-aware truth discovery in mobile crowdsensing," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 10, pp. 2475–2489, Mar. 2018.

[12] L. Wang, D. Zhang, D. Yang, B.Y. Lim, X. Han, and X. Ma, "Sparse mobile crowdsensing with differential and distortion location privacy," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 2735–2749, Feb. 2020.

[13] K. Xue, S. Li, J. Hong, Y. Xue, N. Yu, and P. Hong, "Two-cloud secure database for numeric-related SQL range queries with privacy preserving," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 7, pp. 1596–1608, Mar. 2017.

[14] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Proceedings of the 1999 International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, Apr. 1999, pp. 223–238.

[15] "Python paillier library," https://github.com/data61/python-paillier.