IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received June 10, 2020, accepted June 23, 2020, date of publication July 6, 2020, date of current version July 16, 2020.

*Digital Object Identifier 10.1109/ACCESS.2020.3007168*

# Privacy-Preserving Cost Minimization in Mobile Crowd Sensing Supported by Edge Computing

**ZHUO LI[1], ZIHUI SONG[2], AND XIN CHEN[2], (Member, IEEE)**

[1]Beijing Key Laboratory of Internet Culture and Digital Dissemination Research, Beijing Information Science and Technology University, Beijing 100101, China
[2]School of Computer Science, Beijing Information Science and Technology University, Beijing 100101, China

Corresponding author: Zhuo Li (lizhuo@bistu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61872044 and Grant 61502040, in part by the Beijing Municipal Program for Top Talent, in part by the Beijing Municipal Program for Top Talent Cultivation under Grant CIT & TCD201804055, and in part by the Qinxin Talent Program of Beijing Information Science and Technology University.

**ABSTRACT** To minimize the sensing cost in MCS while preserving the participants' privacy, in this paper we propose a Data Sensing mechanism with User Privacy Preserved (DS-UPP). We introduce edge computing into MCS to support task allocation and user privacy protection. In DS-UPP, based on compressive sensing theory we minimize the amount of data needed to be submitted. We also design an algorithm based on local differential privacy theory. Selected participants only need to submit their real data along with the reconstructed data generated by the algorithm. It is proved that DS-UPP satisfies $\varepsilon$-differential privacy. We give the mathematical lower bound and upper bound of the number of participants needed for task accomplishment with the constraints that privacy budget is $\varepsilon$ and recovery error of task data is 0, as well as the average amount of data that should be submitted by a participant. We also evaluate the performance of DS-UPP through simulations. Compared with the existing method PrivKV, DS-UPP can reduce the needed data amount by about 90% on the average while guarantee users' privacy preserved.

**INDEX TERMS** Mobile crowd sensing, privacy preservation, edge computing.

## I. INTRODUCTION

Mobile crowd sensing (MCS) is an efficient way to obtain physical social data through using mobile devices carried by people. According to IDC, the global sales volume of smartphones has reached 33.32 million in the second quarter of 2019. Due to more and more sensing equipment fit into mobile devices, their sensing capabilities are improved greatly and they become important supplement for the traditional deployed static sensors. Compared with the way of deploying embedded hardware nodes [1], MCS is more free and low-cost sensing data. Therefore, MCS has become more and more popular in academia and industry, which is widely used in environmental monitoring [2], [3], intelligent transportation [4], urban management [5] and so on.

In MCS, task allocation has been an important problem. Appropriate participants are needed to be selected to provide high-quality data for tasks with low cost. When the privacy problem is taken into consideration, the problem becomes more complex. This is because that the effectiveness of task allôction in MCS usually depends on accurate users' information and sensing data, which may leak users' sensitive private information. On the other side, traditional privacy preservation mechanisms always degrade data quality. In the process of data submission by participants, the existing techologies, e.g., differential privacy [6], fuzzy logic based routing [7] and etc., usually bring distortion to the sensing data, which reduces the accuracy of the MCS organizer's statistical results of the tasks.

In the MCS task allocation stage and data submission stage, the privacy of user location information is protected, and the trade-off between the privacy protection requirement of participants and the high-quality data requirement of the MCS organizer are achieved. We design and implement an MCS Data Sensing mechanism under User Privacy Preserved (DS-UPP). It introduce the technology of edge computing into MCS, and changes the traditional two-tier architecture, i.e., user and cloud, to three tier, i.e., user, edge computer server, and cloud. The introduction of edge computing can reduce response latency and eliminate the overhead on the backbone infrastructure. More importantly, the edge computing server can perform the management of mobile sensing users in its area where it is deployed, allocating sensing tasks and processing the raw sensed data properly [8].

The associate editor coordinating the review of this manuscript and approving it for publication was Maurice J. Khabbaz.

After appropriate aggregation, the users' privacy can be protected efficiently.

In DS-UPP, we try to select participants who can provide sensing data with high quality. Different from existing work, based on the theory of compressive sensing we utilize the relationship among sensing data from different users and minimize the amount of data needed to be collected. DS-UPP can improve sensing cost significantly, which is also proved through experiments. For the privacy protection problem of participants, we design a privacy preservation algorithm for their submitted sensing data based on LDP. There are two main advantages of our algorithm. The first one is that it is implemented locally by each single participant, which can provide quantitative privacy preservation without relying on any other trusted entity. The second one is that the published data by the participants have the same statistical characteristics of the original data. There is no noise in the published data, and no statistical calculation is required to restore the original information.

Our main contributions in this paper are as follows:

1. We introduce edge computing into MCS and proposed the three-tier architecture, under which we make innovative designs for task allocation and user data submission. In task allocation, the edge servers distribute the task requirement to users, whose precise personal information is no longer necessary. For data submission, edge servers help the participants submit high-quality perturbed data which will not expose their privacy.

2. We define the sensing cost minimization problem with privacy preserved, and propose the DS-UPP mechanism for it. In DS-UPP, we develop an algorithm based on compressive sensing to minimize the amount of necessary sensing data, as well as an algorithm based on LDP to protect the participants' privacy.

3. We analyse the theoretical characteristics of DS-UPP. It is proved that DS-UPP satisfies $\varepsilon$-differential privacy. We give the mathematical lower bound and upper bound of the number of participants needed for task accomplishment with the constraints that privacy budget is $\varepsilon$ and recovery error of task data is 0, as well as the average amount of data that should be submitted by a participant. Also, we implement a simulator and take thorough simulations to evaluate the performance of DS-UPP. For comparison, we implement the algorithm PrivKV. It is found from the results of experiments that DS-UPP can reduce the sensing cost by almost 90% on average under the same requirement of privacy and data quality.

This paper is organized as follows. In section II, we discuss the related work. In section III, we formalize the system model and define the problem. We then introduce our proposed architecture of MCS and the DS-UPP mechanism in section IV. In section V, we theoretically analyse the characteristics of DS-UPP and In section VI, we evaluate its performance through simulations. We conclude in Section VII.

## II. RELATED WORK

When the strategy of MCS is taken, coverage quality [9] is used to measure how the MCS tasks are allocated. It is the number of sensing readings at each task location. In order to select the least number of users to participate and guarantee the quality, the MCS organizer needs to know the location of every user for assigning tasks. Reference [10] investigates the situation in which participants perform multiple tasks. The interdependency among multiple tasks are taken into account when allocating the tasks, which optimizes the overall utility of the tasks while ensuring the sensing quality of each one. It can be observed that in order to improve the efficiency of task execution in a complex environment, detailed and rich information of the participants is necessary for the organizer, which makes the privacy of participants vulnerable for leakage.

It is challenging to design privacy-preserving task allocation mechanisms in MCS. It is infeasible to send accurate information of participants directly to MCS organizer due to privacy protection. As a result, it may lead to unreasonable task assignments. To achieve the trade-off between privacy preservation and effectiveness in task allocation, [11] proposes a method in which users obfuscate their reported locations before submission, while minimizing total distances traveled by the mobile users for accomplishing of the sensing tasks. Different user's privacy levels are introduced in [12], which also tries to optimize the users' travelling distance. These methods add noise to the user's distance to the task according to LDP. But it also reveals that the participant is near the task in which he or she is involved. And neither protects location privacy when users submit data.

In [13], tasks are allocated from the view of users and the centralized server. Both worker-selected tasks (WST) model and server-assigned tasks (SAT) model are considered, which tries to balance the requirement of user privacy and task accomplishment quality. The way the user privately chooses the tasks it wants to perform protects the user's location, but the MCS system is less efficient because task assignments are not controlled. Then, some work has been done to design anonymous methods for participants. Reference [14] proposed a private method of using group signatures. Reference [15] proposes a way for mobile nodes such as vehicular to interact with each other in the block chain. There is also mechanisms based on deep reinforcement learning [16]. But those works add complexity to the approach.

Different from the above work, we protect the participant's choice about the task and control the work efficiency through the private participant choice. The location information in the participant submitted data is also protected. We investigate the relationship among sensed data of different areas to select the pieces with high quality, minimizing the amount of data needed while satisfying users' privacy preserved.

Different technologies have been developed to protect data privacy in MCS, e.g., based on the theory of distributed agent [17], that of differential privacy [18]–[21] and etc. In the most

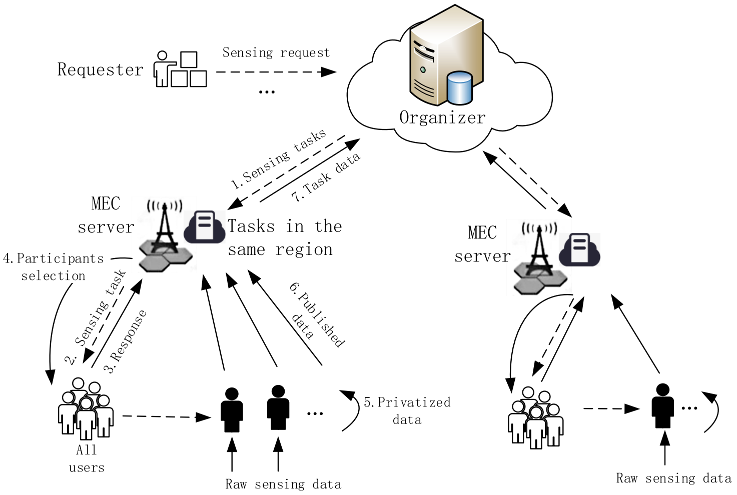


**FIGURE 1.** **MCS supported by edge computing.**

related work [22], a data privacy protection algorithm for the key-value type is designed. It privatizes the user-executed task (key) and the user's data (value) for the task based on LDP theory, which makes sure that the perturbed data has the same statistical characteristics of the original data, e.g., the same estimated key value and mean value. However, it needs a large amount of sample data which means that its cost is high. Our work is different from it in that we optimize the sensing cost through minimizing the necessary data amount for the tasks.

Mobile Edge Computing(MEC) is used to protect the privacy of healthcare internet of things devices [23]. For the MEC based architecture for MCS, [24] proposes that the system load can be reduced and users' privacy can be better protected through the partition and distribution of users' sensitive data. Different from our work, it is more about the architecture and application scenarios. There is no specific algorithms for privacy preservation nor for sensing cost minimization.

## III. SYSTEM MODEL AND PROBLEM DEFINITION

The architecture for mobile crowd sensing system supported by edge computing is shown in Figure.1. Sensing task requests are usually of different types and from different regions. We can assign different requests to different mobile edge computing (MEC) servers, which are deployed and in charge of sensing tasks in the certain areas. MEC servers select appropriate users to perform as participants for the tasks, collect data submitted by them, and then submit data to the MCS organizer after proper aggregation process. Thus, in this architecture the MEC server plays three important roles. One is to distribute tasks according to sensing requests and select suitable participants to join in. The second one is to collect and aggregate the data submitted by the participants and ensure that the task is completed successfully. Thirdly, the MEC server isolates participants and the organizer, which can effectively reduces the threats of privacy leakage of participants, especially after particular data aggregation.

We introduce some definitions and notations as follows.

### A. DEFINITION

MEC server can be responsible for multiple different requests if they are from the area where the server is deployed and in charge of. For a request, we denote all its required tasks as the set $T = \{t_1, t_2, \cdots, t_N\}$. We denote the users as $U = \{u_1, u_2, \cdots, u_j, \cdots\}$, and the selected participants as $U_c$, $U_c \subseteq U$.

*Definition 1:* Task data matrix $D$. $D$ is a one-dimensional column vector including all tasks sensing data of $T$.

*Definition 2:* Task Selection Matrix $\mathbf{C}$. Matrix element $c_i \in \{0, 1\}$. If $c_i^j = 1$ means the participant $u_j$ performs the task $t_i$, and if else $c_i^j = 0$.

*Definition 3:* Data Published Matrix $\mathbf{S}$. $s_i \in \{0, 1\}$. $s_i^j = 1$ means that the participant $u_j$ publishes the data of task $t_i$, and if else $s_i^j = 0$.

$C^i$ denotes the real sensing action of user $u_i$. When $c_n^i = 1$, it means $u_i$ really performs the task $t_n$. Different from it, $S^i$ denotes $u_i$ publishes data of task $t_n$, but it may be unreal and in fact $u_i$ does not performs the task. This perturbation is taken just for privacy preservation. We use the correlation between $S^i$ and $C^i$ to measure the degree of privacy threats of participant $u_i$.

In this paper, the participant's privacy discussed is mainly related with his preference for task selection, location, and the published data, which may reveal his trajectories and other private information. The emphasis of our work is on how the participants publish data. Our target is to guarantee that certain statistical features of published data are the same with those of the original sensed data, and from it attackers can hardly obtain the private information. Therefore, we regard the degree of privacy threat as equivalent to the risk of sensitive information to be leaked.

*Definition 4:* Privacy budget $\varepsilon$. We quantify the degree of privacy threats of participants based on the local differential privacy theory as follows.

$$\frac{P(s_n^i | c_n^i = s_n^i)}{P(s_n^i | c_n^i \neq s_n^i)} \leqslant e^{\varepsilon} \tag{1}$$

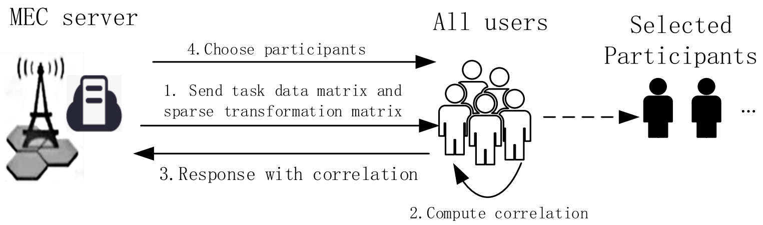


**FIGURE 2. Workflow of participants selection.**

**TABLE 1. Notations.**

| symbol | explanation |
|---|---|
| $T$ | Set of all MCS tasks |
| $U$ | Set of all MCS users |
| $U_C$ | Set of participants is selected from $U$ |
| $\mathbf{D}$ | Sensing data matrix of all tasks $T$ |
| $\mathbf{C}_i$ | A matrix that indicates which tasks $u_i$participated in |
| $\mathbf{S}_i$ | A matrix that indicates which tasks $u_i$ contributed data to |
| $H$ | Total amount of sensing data for all users |
| $\varepsilon$ | Privacy budget of user's submitted data |
| $\mathbf{\Psi}$ | A matrix transforms $\mathbf{D}$ to a sparse matrix |
| $\mathbf{\Phi}_i$ | A matrix represents how $u_i$ collects task data |
| $\hat{\mathbf{D}}^i$ | The recovery data matrix for user $u_i$ |
| $\mathbf{P}^i$ | The submit data matrix for user $u_i$ |

*Definition 5:* Task data error $\hat{E}$. We use *Mean Absolute Deviation* to measure sensing data error. $\hat{E}$ is the average error of each task in task $T$.

$$\hat{E} = \frac{\sum_{d_n \in D, d'_n \in D'} |d'_n - d_n|}{N}, \tag{2}$$

where $D'$ is the published data.

*Definition 6:* We define the total amount of data required to complete the tasks $H$ as follows, which is also the sum of data collected by all participants.

$$H = \sum_{u_i \in U_c} ||C^i||_1 \tag{3}$$

We explain all the symbols used in the article.

### B. SENSING COST MINIMIZATION PROBLEM WITH PRIVACY PRESERVED

We define the sensing cost minimization problem with participants' privacy preserved as follows.

$$\min \quad H \tag{4}$$

$$s.t. \quad T \leqslant \sum_{u_i \in U} s^i_n, \tag{5}$$

$$e^{\varepsilon} \geqslant \frac{P(s^i_n | c^i_n = s^i_n)}{P(s^i_n | c^i_n \neq s^i_n)}, \tag{6}$$

where $c \in \mathbf{C}$, $s \in \mathbf{S}$, $u_i \in U_c$ and $U_c \subseteq U$. (5) is to ensure that the MCS tasks are completed successfully, i.e., the amount of submitted sensing data for task $t_n$ satisfies All tasks $T$. (6) is the privacy budget requirement of participant $u_i$.

## IV. DATA SENSING MECHANISM UNDER USER PRIVACY PRESERVING IN MOBILE CROWD SENSING

We designed the DS-UPP privacy protection mechanism, which can satisfy the privacy preservation requirements, while minimizing the amount of data necessary for MCS.

We introduce the mechanism in two stages. The first step, MCS organizers allocate tasks to users in a private way. In the second step, the user submits the perceived data privacy to the MCS organizer.

### A. TASK ALLOCATION BASED ON COMPRESSIVE SENSING

The MCS organizer coordinates all sensing requests and assigns them to proper MEC servers according to the region where the tasks are located. After that, MEC servers select participants based on compressive sensing theory, in order to minimize the amount of sensing data for accomplishing the tasks. Users choose tasks which they can collect sensing data for, based on their trajectories or preferences. MEC is deployed in base stations or routers to provide sufficient computing power and external power supply.

After choosing tasks, every user generates the Task Selection Matrix (TSM), and then calculates the correlation between TSM and the sparse transformation matrix, based on the compressive sensing theory.

Based on the correlation of each user, the MEC server selects the optimal combination of users as the participants. Participants are selected in a greedy manner, i.e., those participants whose correlation value is biggest will be selected first. Figure 2 shows the participants selection process.

In detail, task allocation in DS-UPP consists of three key steps.

1) When the MEC server receives a task assigned by the organizer, it divides the request into the task set $T = \{t_1, t_2, \cdots, t_N\}$. At the same time, it use the second-order difference matrix as the sparse transformation matrix. The sparsity of the data transformed by the second-order difference matrix can reach 5% for perception data [25]. A principal component analysis algorithm based on singular value decomposition is presented in [25]. We used this method to decompose different types of historical data to get a sparse transformation matrix.

2) The MEC server distributes task set $T$ and the sparse transformation matrix $\Psi_{N\times N}$ to all users associated with it. Each user $u_i \in U$ selects tasks from $T$, and generates the Task Selection Matrix $C^i$ according to his preference. $u_i$ also calculates the correlation $\mu$ between $C^i$ and $\Psi$ as follows.

$$\mu_i(\Phi^i, \Psi) = \sqrt{N} \max(|\Phi^i_{N_i\times N} \cdot \Psi^{-1}_{N\times N}|) \tag{7}$$

$\Phi^i$ is generated by $C^i$. The number of rows in $\Phi^i$ is the number of tasks collected by $u_i$, i.e., $N_i$, and the number of columns is $N$. $\Phi^i$ has a similar meaning but different form

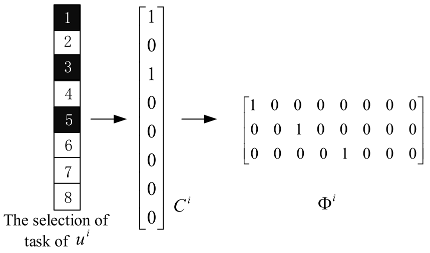


**FIGURE 3.** Generate an acquisition matrix based on TSM.

as compared with $C^i$. In $\Phi^i = (\phi_{mn})_{N_i \times N}$, $N_i = ||\mathbf{C}_u||_0$ is the number of non-zero elements in $\mathbf{C}_i$, and $N$ is the number of all elements in $\mathbf{C}_i$. The following equation gives the mathematical definition of $\Phi^i$ and Figure 3 demonstrates an example of how to use $\mathbf{C}_i$ to generate $\Phi^i$.

$$\Phi^i = \{\phi_{mn}|\phi_{mn} = \begin{cases} 1 & \text{if the } n_{th} \text{ element in } \mathbf{C}_i \text{ is} \\ & \text{the } m_{th} \text{ non-zero element,} \\ 0 & else. \end{cases}\} \quad (8)$$

As a response, $u_i$ sends back the correlation value $\mu_i$ to the MEC server.

3) After receiving correlation values from all users, the MEC server sorts them in an descending order. The server also calculates the number of required participants $U_c$ according to the relationship between the amount of participants and the data error threshold. Finally, in a greedy manner the server selects $U_c$ participants, i.e., the first $U_c$ users according to the descent order of their correlation values.

## B. PRIVACY PRESERVATION OF SENSING DATA BASED ON LDP

In this subsection, we introduce the method for participants to perturb data locally before publishing data. The working process of participants is shown in the Figure4.

For the participant $u_i$, he collects the sensing data according to the task requirement, as shown in Fig.4(a). Then, he tries to recover data of all tasks using Algorithm 1, shown as Fig.4(b). We use prediction methods to generate the missing data of $u_i$ for certain tasks. After that, Algorithm 2 is taken for $u_i$ to perturb data before publishing, through which his privacy can be protected.

**Algorithm 1** Recovery of Data for All Tasks

```
Input: Raw sensing data $\mathbf{D}^i$
       Task acquisition matrix $\Phi^i$
       Sparse transformation matrix $\Psi$
       The number of non-zero elements $K$ in $\Lambda$
Output: All tasks data $\hat{\mathbf{D}}^i$
$\mathbf{A} = \Phi^i \times \Psi^{-1}$
Initialize residual data $\mathbf{R}_0 = \mathbf{D}^i$,vector ordinal set $\mathbf{V}_0 = \emptyset$, vector set $\mathbf{A}_0 = \emptyset$
for ($t = 1$; $t \leqslant K$; $t++$) do
    for (column vector $\mathbf{a}_n$ in $\mathbf{A}$) do
        $\lambda_t = \arg_n \max_{n \in \{1,2,\cdots,N\}} |\mathbf{R}_{t-1} \cdot \mathbf{a}_n|$
    end for
    $\mathbf{V}_t = Add \quad \lambda_t \quad into \quad \mathbf{V}_{t-1}$
    $\mathbf{A}_t = Add \quad \mathbf{a}_{\lambda_t} \quad into \quad \mathbf{A}_{t-1}$
    Calculate the least squares solution $\hat{\Lambda}_t$:
    $\hat{\Lambda}_t = (\mathbf{A}_t^T \cdot \mathbf{A}_t)^{-1} \cdot \mathbf{A}_t^T \cdot \mathbf{D}^i$
    $\mathbf{R}_t = \mathbf{D}^i - \mathbf{A}_t \cdot \Lambda_t$
end for
Compute the coefficient of $\hat{\Lambda}_K$ in $\mathbf{V}_K$
$\hat{\mathbf{D}}^i = \Psi^{-1} \cdot \hat{\Lambda}_K$
return $\hat{\mathbf{D}}^i$
```

### 1) RECOVER ALL TASK DATA

We denote the sensing data of participant $u_i$ as $D^i$. According to the previous definitions, $D^i$ and $D$ have the following relationship.

$$\mathbf{D}^i_{N_i \times 1} = \Phi^i_{N_i \times N} \cdot \mathbf{D}^T_{N \times 1} \quad (9)$$

$T$ is the transpose of a matrix. The sparse transformation matrix $\Psi$ changes $D$ to $\Lambda$.

$$\Lambda_{N \times 1} = \Psi_{N \times N} \cdot \mathbf{D}^T_{N \times 1}, \quad ||\Lambda||_0 = K \quad (10)$$

$N$ is the number of all tasks, $N_i$ is the number of data collected by $u_i$, and $K$ is the number of non-zero data in the sparse matrix $\Lambda$. $M > N > K$.

We know that (9) is an under-determined equation, so we cannot solve it for $\mathbf{D}$ directly. We use the following formula.

$$\mathbf{D}^i_{N_i \times 1} = \Phi^i_{N_i \times N} \cdot \Psi^{-1}_{N \times N} \cdot \Lambda^T_{N \times 1} \quad (11)$$

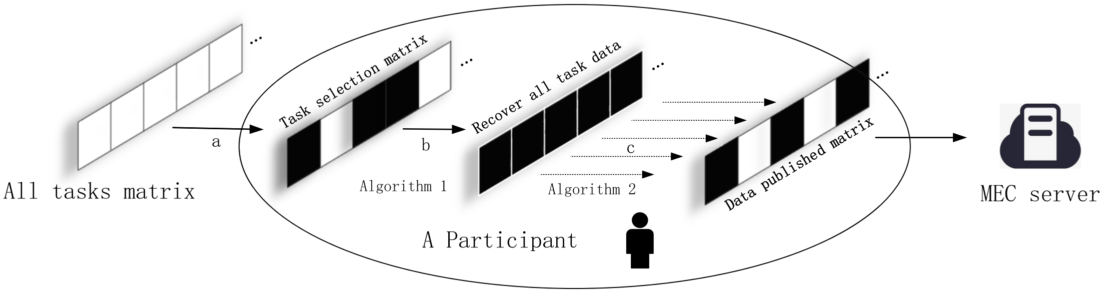


**FIGURE 4.** Sensing and perturbation of data by participants.

**Algorithm 2** Privacy Preserving for Participant Data

```
Input: All task data $\hat{\mathbf{D}}^i$ of $u_i$
       Task Selection Matrix $\mathbf{C}^i$ of $u_i$
       Privacy budget $\varepsilon$
Output: The publish data $\mathbf{P}^i$ of $u_i$
Initialize $\mathbf{S}^i = zeros.shapeas(\mathbf{C}^i)$
for $s \in \mathbf{S}^i, c \in \mathbf{C}^i$ do
    Compute
    $s = \begin{cases} c & w.p. \quad \frac{e^{\varepsilon}}{1+e^{\varepsilon}} \\ 1-c & w.p. \quad \frac{1}{1+e^{\varepsilon}} \end{cases}$
end for
$\mathbf{P}^i = \mathbf{S}^i \circ \hat{\mathbf{D}}^i$
return $\mathbf{P}^i$
```

Solving $\Lambda$ with $\mathbf{D}^i$ is a positive definite equation. Candes has verified that when $\Psi$ satisfies the RIP property, reconstruction $\Lambda$ is a solvable optimization problem [26].

$\mathbf{D}$ can be obtained through (10). Inferring the task of non-zero elements in is a convex optimization problem

The time complexity of algorithm 1 is $O(KN)$. According to Orthogonal Matching Pursuit [27], we generate the sparse matrix $\Lambda$ and restore all tasks data $\hat{\mathbf{D}}$ by the Algorithm 1. First, we calculate the reduced dimensional transformation matrix $\mathbf{A} = \Phi^i \times \Psi^{-\mathbf{1}}$. Then, we initialize the residual $\mathbf{R}_0 = \mathbf{D}^i$. The column vector $\mathbf{a}_{\lambda_t}$ which is selected from the matrix $\mathbf{A}$ has the largest inner product with the residual $\mathbf{R}_t$ for each iteration. We calculate the least squares of the linear equation $\mathbf{D} = \mathbf{A}_t \cdot \Lambda_t$ using all iterative column vectors, and the number of iterations is $K$. We can get the matrix $\hat{\Lambda}_K$ according to the set $\mathbf{V}_K$. Finally, data for all tasks, i.e., the matrix $\hat{\mathbf{D}}$, can be obtained.

### 2) PARTICIPANT LOCAL DATA PRIVACY

We use $\mathbf{D}^i$ to represent the raw sensing data of participant $u_i$. It is known that $\mathbf{C}^i$ and $\mathbf{D}^i$ indicates the user's real choice of tasks and the real sensed data. We use $\mathbf{S}^i$ to indicate the tasks to be published by $u_i$. Through Algorithm 1, data for all tasks data can be obtained. Thus, the published data of $u_i$ is

$$\hat{\mathbf{D}}^i = S^i \circ \hat{\mathbf{D}}. \tag{12}$$

We protect user privacy based on LDP theory, as shown in Algorithm 2. The relationship between the task in the data published by the $u_i$ and the real task of $u_i$ satisfies the $\varepsilon$-LDP constraint. We set the relationship between $c_n^i \in C^i$ and $s_n^i \in S^i$ as follows.

$$P(s_n^i | c_n^i) = \begin{cases} \dfrac{e^{\varepsilon}}{1+\varepsilon} & s_n^i = c_n^i \\ \dfrac{1}{1+\varepsilon} & s_n^i = !c_n^i \end{cases} \tag{13}$$

Specifically, there are four cases for the element in $C^i$ and $S^i$.

$0 \rightarrow 0$: Participants neither sense nor publish.

$1 \rightarrow 0$: Participants sense but do not publish.

$0 \rightarrow 1$: Participants do not sense but publish.

$1 \rightarrow 1$: Participants sense and publish.

Any element $c_n^i$ in $C^i$ and $s_n^i$ in $S^i$ of the same task $n$ satisfies

$$\frac{P(s_n^i = 1 | c_n^i = 1)}{P(s_n^i = 1 | c_n^i = 0)} \leqslant e^{\varepsilon}, \quad \text{and} \quad \frac{P(s_n^i = 0 | c_n^i = 0)}{P(s_n^i = 0 | c_n^i = 1)} \leqslant e^{\varepsilon}. \tag{14}$$

Following Algorithm 2, participants locally process the sensing data and generate published data. The MEC server collects the published data from all associated participants and send it back to the MCS organizer, to accomplish the required tasks.

The time complexity of algorithm 2 is $O(N)$.

When the user performs enough tasks, the missing data generated by the user can have no error. Theorem 2 shows the number of tasks collected is lower bound. If the lowe bound is not satisfied, there will be distorted data. We calculate the mean or mode of all participants' data. In practical applications, more incentives are provided to motivate users to perform more tasks, and multiple data are collected for each task to ensure the accuracy of the data.

## V. THEORETICAL ANALYSIS

In this section, we analyse the performance of DS-UPP theoretically. Firstly, we prove that the published data in the DS-UPP paradigm satisfies the LDP privacy constraint.

*Theorem 1:* For the published data from any participant in DS-UPP, it satisfies $\varepsilon$- differential privacy.

Besides privacy preservation, DS-UPP also optimize the selection of high-quality sensed data, in order to minimize the sensing cost as shown in (4). From [28], we can obtain the following theorem. It gives the lower bound of the amount of data that every participant should submit to satisfy the requirement of compressive sensing.

*Theorem 2:* When the average amount of data for a participant in DS-UPP satisfies

$$N_i \geq \mu_i(\Phi^i, \Psi) \cdot K \cdot ln(N), \tag{15}$$

the original data can be recovered without error [25].

For the number of participants needed in DS-UPP, we have the following theorem, which gives its lower bound and upper bound.

*Theorem 3:* Under the constraints that privacy budget is $\varepsilon$ and recovery error of task data $\hat{E} = 0$, the number of participants needed for task accomplishment in DS-UPP $m_c$ must satisfy $log_{1+e^{\varepsilon}} N \leqslant m_c < log_{\frac{1+e^{\varepsilon}}{e^{\varepsilon}}} N$.

*Proof:* We assume that the number of tasks performed by the participants is $N_1, N_2, N_3, \cdots N_i \cdots N_{m_c}$. For a participant $i$, we can set the probability that any task will be taken by him is $\frac{N_i}{N}$. We can get that, in any participant's submitted data the probability that the perceived task is not involved is

$$\frac{N - N_i}{N} \frac{e^{\varepsilon}}{1+e^{\varepsilon}} + \frac{N_i}{N} \frac{1}{1+e^{\varepsilon}}. \tag{16}$$

Because participants perform tasks independently, after the $k$ participants submit data the probability that the task is not involved is

$$\prod_{k=1}^{m_c}\left(\frac{N-N_k}{N}\frac{e^{\varepsilon}}{1+e^{\varepsilon}}+\frac{N_k}{N}\frac{1}{1+e^{\varepsilon}}\right). \tag{17}$$

So if the data completely covers all tasks, it must satisfy

$$N \times \prod_{k=1}^{m_c}\left(\frac{N-N_k}{N}\frac{e^{\varepsilon}}{1+e^{\varepsilon}}+\frac{N_k}{N}\frac{1}{1+e^{\varepsilon}}\right) = 1 \tag{18}$$

We can simplify it as

$$\prod_{k=1}^{m_c}\left(\frac{N-N_k}{N}\frac{e^{\varepsilon}}{1+e^{\varepsilon}}+\frac{N_k}{N}\frac{1}{1+e^{\varepsilon}}\right) = \frac{1}{N},$$

$$\prod_{k=1}^{m_c}\left(\frac{e^{\varepsilon}}{1+e^{\varepsilon}}-\frac{N_k}{N}\frac{e^{\varepsilon}}{1+e^{\varepsilon}}+\frac{N_k}{N}\frac{1}{1+e^{\varepsilon}}\right) = \frac{1}{N},$$

$$\prod_{k=1}^{m_c}\left(\frac{e^{\varepsilon}}{1+e^{\varepsilon}}-\left(\frac{N_k}{N}\times\frac{e^{\varepsilon}-1}{1+e^{\varepsilon}}\right)\right) = \frac{1}{N}. \tag{19}$$

We know that

$$0 < N_k \leqslant N, \tag{20}$$

and we can analyse the upper bound and lower bound of the number of necessary participants.

In the worst case, participants only to collect the smallest size of samples. $\frac{N_i}{N}$ can be regarded as 0 and the above formula is reduced to

$$\left(\frac{e^{\varepsilon}}{1+e^{\varepsilon}}\right)^{m_c^{max}} = \frac{1}{N} \tag{21}$$

We can get

$$m_c^{max} = log_{\frac{1+e^{\varepsilon}}{e^{\varepsilon}}}N. \tag{22}$$

In the best case, each participant collects as many samples as possible and we have

$$\left(\frac{e^{\varepsilon}}{1+e^{\varepsilon}}-\frac{e^{\varepsilon}-1}{1+e^{\varepsilon}}\right)^{m_c^{min}} = \frac{1}{N},$$

$$\left(\frac{1}{1+e^{\varepsilon}}\right)^{m_c^{min}} = \frac{1}{N},$$

$$m_c^{min} = log_{1+e^{\varepsilon}}N. \tag{23}$$

Therefore, the range of $m_c$ is

$$log_{1+e^{\varepsilon}}N \leqslant m_c < log_{\frac{1+e^{\varepsilon}}{e^{\varepsilon}}}N. \tag{24}$$

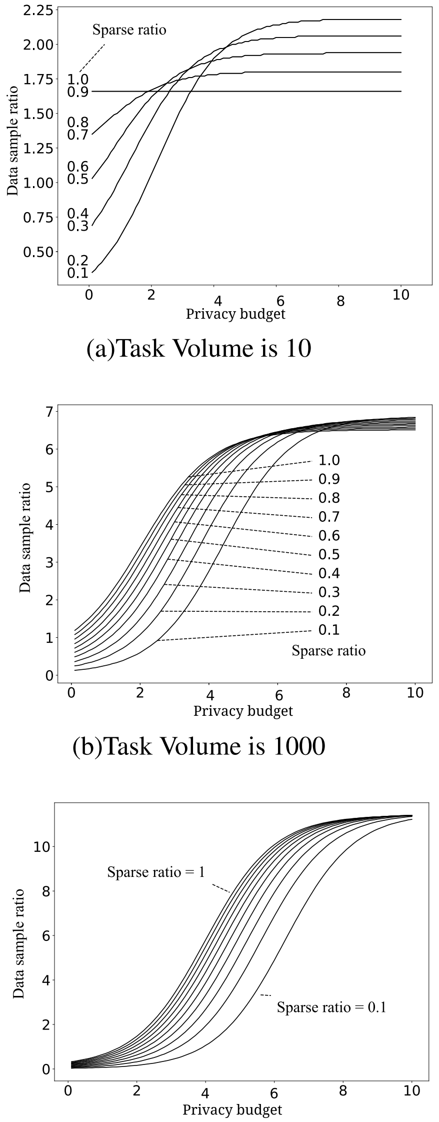
■

According to the above two theorems, we can get that the total amount of data needed in DS-UPP for the optimization problem (4) satisfies

$$H \geqslant log_{1+e^{\varepsilon}}N \times \mu_i(\Phi_i, \Psi) \cdot K \cdot lnN. \tag{25}$$

## VI. PERFORMANCE EVALUATION

To evaluate the performance of DS-UPP, we implement a simulator in Python and take thorough simulations. We analyse the influence of different parameters, i.e., number of different tasks, privacy budget and data sparsity ratio, as shown in Figure 5. The number of tasks included in each of the three scenarios is 10, 1000 and 100000. We use the metric Data Sample Ratio (DSR) to measure the performance, which is the average amount of data needed for accomplishing a piece of task. It also means the amount of data that must be collected for each task.



(a)Task Volume is 10

(b)Task Volume is 1000

(c)Task Volume is 100000

**FIGURE 5. Influence of task volume, privacy budget and data sparsity rate.**

### A. THE AMOUNT OF DATA REQUIRED OF DS-UPP

We evaluate the DSR under different settings of privacy budget and data sparsity ratio. Privacy budget changes from 0.1 to 10 and data sparsity ratio changes from 0.1 to 1, both with the step size 0.1. When the privacy budget is the minimum, equal to 0, the privacy of data is the greatest. With the increase of privacy budget, the privacy of data decreases and converges. When the privacy budget is larger than 10, the privacy state tends to be the same. For the results of experiments, we take 100 simulations to get the average value and in every simulation we generate the sensing data of each user randomly.

In the previous section, it is proved that DS-UPP meets the privacy requirement. It is observed from the experiments that in DS-UPP the DSR is below 2.25, 7 and 12 in the three different scenarios, which means that the average amount of sample data needed for each task is less than 2.25, 7

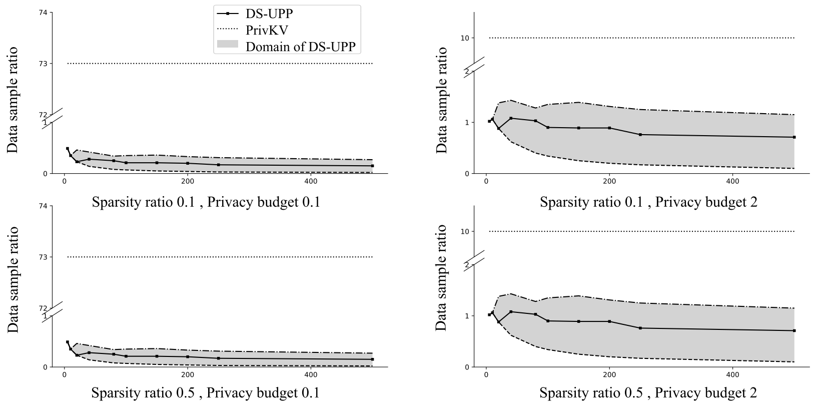


**FIGURE 6.** Comparison of DS-UPP and PrivKV on DSR.

and 12, respectively. Also, it can be found that the DSR increases with the sparsity ratio, as well as the privacy budget, and converges to a upper bound. This upper bound slightly reduces when the sparsity ratio increases.

The increase of sparsity ratio requires more acquired values for the recovery of all data. Thus, the size of sampled data becomes larger and DSR increases, as shown in the left half part of the curves in Figure 5(a) and (b). On the other hand, when the privacy budget is large enough, the probability that the data stays true in the perturbation approaches to 1 and the sample size is mainly affected by the number of tasks that do not have collected data. In this case, as the sparsity ratio increases the amount of samples increases, too. As a result, the number of tasks missing data is reduced and DSR decreases, as shown in the right half part of the curves in Figure 5(a) and (b). As the total amount of tasks increases, the proportion of changes on the number of tasks missing data becomes smaller and smaller. Thus, the velocity of the DSR increase in the right half of the curves in the three sub-figures becomes slower and slower.

The increase of privacy budget loosens the privacy requirement, but it also leads to an increase of DSR. This is due to that the increase of the privacy budget raises the probability that the data stays true, but decreases the probability that users who do not have sensed data submit perturbed data just as they had finished the sensing task. As a result, more participants are needed to take part in the MCS tasks and DSR is increased.

The number of tasks affects the amount of data collected by a participant according to Theorem 2. Moreover, the number of tasks affects the number of participants according to Theorem 3. As a result, there is a logarithmic relationship between Task Volume and data sample ratio.

There are two special cases. For small number of tasks, the amount of data needed to be collected is almost the same for the close sparsity ratio, e.g., 0.1 and 0.2, 0.2 and 0.3, and etc., as shown in Figure 5 a. We can also find that when the sparse ratio is 0.9 and 1, the amount of necessary data is almost half of the total number of tasks. Therefore, regardless how the privacy budget changes, the number of tasks without sensing data in the submission remains the same. Therefore, DSR stays unchanged.

### B. THE COMPARISON OF THE AMOUNT OF DATA NECESSARY WHEN THE DATA IS ACCURATE

We also implement PrivKV [22] for comparison. PrivKV privatizes the user-executed task (key) and the user's data (value) for the task based on the LDP theory, Which makes sure that the perturbed data has the same statistical characteristics of the original data. The difference between DS-UPP and PrivKV lies in two points. The first is that DS-UPP uses Compressed Sensing to generate data for the user's data collection task, while PrivKV generates random false data. Second, the data generated by DS-UPP is directly the MCS demand data, and no statistical calculation is needed.

The data accuracy of DS-UPP algorithm and PrivKV algorithm increases with the increase of the amount of data collected by users. We verify the comparison of the minimum amount of data collected under the privacy constraint and the data accuracy constraint.

For both PrivKV and DS-UPP, we set the same range and distribution of the task values in simulations. We evaluate their performance on DSR under four different parameter settings, i.e., different sparsity ratios and privacy budgets, as shown in Figure 6. The x-axis of the sub-figures denotes the number of tasks. It can be found that DSR of PrivKV does not change with the number of tasks. This is because in PrivKV DSR is only related with the range of the task values, which remains the same in simulations. In addition with the average DSR values of DS-UPP in simulations, we also evaluate its lower bound and upper bound values, which are demonstrated as the domains of DS-UPP in the figures. It is observed that the size of sample data required in DS-UPP is much smaller than that in PrivKV, which can be reduced by about 90% in each parameter settings. This proves the effectiveness of our optimization on the selection of sensing data with high quality, which greatly reduces the necessary sensing cost.

## VII. CONCLUSION

In this paper, we designed the DS-UPP mechanism to solve the problem of maximizing data efficiency while protecting users' privacy in MCS supported by edge computing. Edge servers help the participants submit high-quality perturbed data which will not expose their privacy. In DS-UPP, we develop a compressive sensing based algorithm to minimize the amount of necessary sensing data. Based on LDP theory, we develop an algorithm to protect participants' privacy. We analyse the performance of DS-UPP theoretically, and also evaluate its performance through simulations. It is found from the results of experiments that DS-UPP can reduce the sensing cost by almost 90% on average compared with the existing algorithm PrivKV.

## REFERENCES

[1] A. Chianese, F. Piccialli, and G. Riccio, "SMuNe: A smart multisensor network based on embedded systems in IoT environment," in *Proc. 11th Int. Conf. Signal-Image Technol. Internet-Based Syst. (SITIS)*, Nov. 2015, pp. 23–27.

[2] M. Zappatore, A. Longo, M. A. Bochicchio, D. Zappatore, A. A. Morrone, and G. De Mitri, "A crowdsensing approach for mobile learning in acoustics and noise monitoring," in *Proc. 31st Annu. ACM Symp. Appl. Comput. SAC*, 2016, pp. 219–224.

[3] Y. Jing, B. Guo, Z. Wang, V. O. K. Li, J. C. K. Lam, and Z. Yu, "CrowdTracker: Optimized urban moving object tracking using mobile crowd sensing," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3452–3463, Oct. 2018.

[4] G. B. Kalejaiye, H. R. Orefice, T. Moura, M. Bafutto, and M. M. Carvalho, "Frugal crowd sensing for bus arrival time prediction in developing regions," in *Proc. 2nd Int. Conf. Internet Things Design Implement.*, Apr. 2017, pp. 355–356.

[5] X. Zhang, Z. Yang, W. Sun, Y. Liu, S. Tang, K. Xing, and X. Mao, "Incentives for mobile crowd sensing: A survey," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 54–67, 1st Quart., 2016.

[6] L. Y. Wang, D. Q. Zhang, and D. Q. Yang, "Differential location privacy for sparse mobile crowdsensing," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Barcelona, Spain, Dec. 2016, pp. 1257–1262.

[7] K. Z. Ghafoor, L. Kong, A. S. Sadiq, Z. Doukha, and F. M. Shareef, "Trust-aware routing protocol for mobile crowdsensing environments," in *Proc. IEEE INFOCOM Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, Apr. 2018.

[8] Y. Pu, C. Hu, S. Deng, and A. Alrawais, "RPEDS: A recoverable and revocable privacy-preserving edge data sharing scheme," *IEEE Internet Things J.*, early access, May 26, 2020, doi: 10.1109/JIOT.2020.2997389.

[9] H. Xiong, D. Zhang, G. Chen, L. Wang, and V. Gauthier, "CrowdTasker: Maximizing coverage quality in piggyback crowdsensing under budget constraint," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, Mar. 2015, pp. 55–62.

[10] J. Wang, Y. Wang, D. Zhang, F. Wang, H. Xiong, C. Chen, Q. Lv, and Z. Qiu, "Multi-task allocation in mobile crowd sensing with individual task quality assurance," *IEEE Trans. Mobile Comput.*, vol. 17, no. 9, pp. 2101–2113, Sep. 2018.

[11] L. Wang, D. Yang, X. Han, T. Wang, D. Zhang, and X. Ma, "Location privacy-preserving task allocation for mobile crowdsensing with differential geo-obfuscation," in *Proc. 26th Int. Conf. World Wide Web*, Apr. 2017, pp. 627–636.

[12] Z. B. Wang, J. H. Hu, R. Z. Lv, J. Wei, Q. Wang, D. Yang, and H. Qi, "Personalized privacy preserving task allocation for mobile crowdsensing," *IEEE Trans. Mobile Comput.*, vol. 18, no. 1, pp. 1330–1341, Jun. 2019.

[13] J. Wang, F. Wang, Y. Wang, L. Wang, Z. Qiu, D. Zhang, B. Guo, and Q. Lv, "HyTasker: Hybrid task allocation in mobile crowd sensing," *IEEE Trans. Mobile Comput.*, vol. 19, no. 3, pp. 598–611, Mar. 2020.

[14] C. Q. Hu, X. Z. Cheng, Z. Tian, J. G. Yu, and W. F. LV, "Achieving privacy preservation and billing via delayed information release," *IEEE/ACM Trans. Netw.*, vol. 1, no. 1, pp. 1–14.

[15] Y. W. Pu, T. Xiang, C. Q. Hu, A. Alrawais, and H. Y. Yan. *An Efficient Blockchain-Based Privacy Preserving Scheme for Vehicular Social Networks*. Accessed: Jun. 29, 2020. [Online]. Available: https://doi.org/10.1016/j.ins.2020.05.087

[16] L. Xiao, Y. Li, G. Han, H. Dai, and H. V. Poor, "A secure mobile crowdsensing game with deep reinforcement learning," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 1, pp. 35–47, Jan. 2018.

[17] Z. Wang, X. Pang, Y. Chen, H. Shao, Q. Wang, L. Wu, H. Chen, and H. Qi, "Privacy-preserving crowd-sourced statistical data publishing with an untrusted server," *IEEE Trans. Mobile Comput.*, vol. 18, no. 6, pp. 1356–1367, Jun. 2019.

[18] Z. Song, Z. Li, and X. Chen, "Local differential privacy preserving mechanism for multi-attribute data in mobile crowdsensing with edge computing," in *Proc. IEEE Int. Conf. Smart Internet Things (SmartIoT)*, Aug. 2019, pp. 283–290.

[19] Q. Wang, Y. Zhang, X. Lu, Z. B. Wang, Q. Zhang, and K. Ren, "Real-time and spatio-temporal crowdsourced social network data publishing with differential privacy," *IEEE Trans. Depend. Sec. Comput.*, vol. 15, no. 4, pp. 591–606, Jul./Aug. 2018.

[20] J. Chen, H. Ma, D. Zhao, and L. Liu, "Correlated differential privacy protection for mobile crowdsensing," *IEEE Trans. Big Data*, early access, Dec. 4, 2017, doi: 10.1109/TBDATA.2017.2777862.

[21] X. Ren, C.-M. Yu, W. Yu, S. Yang, X. Yang, J. A. McCann, and P. S. Yu, "LoPub: High-dimensional crowdsourced data publication with local differential privacy," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 9, pp. 2151–2166, Sep. 2018.

[22] Q. Ye, H. Hu, X. Meng, and H. Zheng, "PrivKV: Key-value data collection with local differential privacy," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2019, pp. 317–331.

[23] M. Min, X. Wan, L. Xiao, Y. Chen, M. Xia, D. Wu, and H. Dai, "Learning-based privacy-aware offloading for healthcare IoT with energy harvesting," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4307–4316, Jun. 2019.

[24] M. Marjanovic, A. Antonic, and I. P. žarko, "Edge computing architecture for mobile crowdsensing," *IEEE Access*, vol. 57, no. 04, pp. 68–73, 2018.

[25] Y. Guo, X. Song, N. Li, and D. Fang, "An efficient missing data prediction method based on kronecker compressive sensing in multivariable time series," *IEEE Access*, vol. 6, pp. 57239–57248, 2018.

[26] E. J. Candes, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Commun. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, Aug. 2006.

[27] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.

[28] E. J. Candes and Y. Plan, "A probabilistic and RIPless theory of compressed sensing," *IEEE Trans. Inf. Theory*, vol. 57, no. 11, pp. 7235–7254, Nov. 2011.

**ZHUO LI** received the Ph.D. degree from Nanjing University, in 2012. He is currently an Associate Professor and an Associate Dean of the Graduate School, Beijing Information Science and Technology University. His research interests include wireless networks, distributed computing, and network security.

**ZIHUI SONG** received the M.S. degree from the School of Computer Science, Beijing Information Science and Technology University, Beijing, China, in 2020. His research interests include performance valuation of mobile crowd sensing and privacy preserving.

**XIN CHEN** (Member, IEEE) received the Ph.D. degree in computer science from the Beijing Institute of Technology, Beijing, China. He is currently a Professor with the Computer School, Beijing Information Science and Technology University. His current research interest includes performance evaluation of wireless networks. He is a Senior Member of the China Computer Federation (CCF). He is also a member of the CCF Technical Committee of Theoretical Computer Science, and the CCF Technical Committee of Petri Nets. He received the Postdoctoral Fellowship in computer architecture from Tsinghua University, in 2006.