IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 1, 2018, accepted March 9, 2018, date of publication March 19, 2018, date of current version April 4, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2816918

Density-Based Location Preservation for Mobile Crowdsensing With Differential Privacy

MENGMENG YANG[1], TIANQING ZHU[1,2], (Member, IEEE), YANG XIANG[3], (Senior Member, IEEE), AND WANLEI ZHOU[1], (Senior Member, IEEE)

[1]School of Information Technology, Deakin University, Melbourne, VIC 3125, Australia
[2]School of Mathematics and Computer Science, Wuhan Polytechnic University, Wuhan 430023, China
[3]Digital Research and Innovation Capability Platform, Swinburne University, Melbourne, VIC 3122, Australia

Corresponding author: Tianqing Zhu (t.zhu@deakin.edu.au)

This work was supported in part by the National Natural Science Foundation of China under Grant 61502362 and in part by the Australia Research Council Grant under Grant LP170100123.

ABSTRACT In recent years, the widespread prevalence of smart devices has created a new class of mobile Internet of Thing applications. Called mobile crowdsensing, these techniques use workers with mobile devices to collect data and send it to task requester for rewards. However, to ensure the optimal allocation of tasks, a centralized server needs to know the precise location of each user, but exposing the workers' exact locations raises privacy concerns. In this paper, we propose a data release mechanism for crowdsensing techniques that satisfies differential privacy, providing rigorous protection of worker locations. The partitioning method is based on worker density and considers non-uniform worker distribution. In addition, we propose a geocast region selection method for task assignment that effectively balances the task assignment success rate with worker travel distances and system overheads. Extensive experiments prove that the proposed method not only provides a strict privacy guarantee but also significantly improves performance.

INDEX TERMS Crowdsensing, differential privacy, location privacy.

I. INTRODUCTION

Crowd sensing as a new trend of development in Internet of Things (IoT) takes advantage of pervasive sensor-equipped mobile devices to collect and share data. The phenomenon has given rise to numerous large scale, real-world applications, which have the power to create awareness about a specific large-scale phenomena and to ignite crowd intelligence [1], such as environment monitoring [2], traffic condition detection [3], and point-of-interest characterization [4]. In a typical crowdsensing platform, participants are registered as candidate workers. A centralized server (hereafter, the Server) selects workers to complete a data-collection task, and they are paid a reward for doing so. The selected workers then travel to a predefined location to collect the required data. However, to be able to assign tasks more efficiently, workers need to submit their exact location to the Server. Disclosing one's location raises serious privacy concerns as the Server may not be trusted. Given a lack of privacy protection may affect worker uptake of such systems, ensuring the privacy of the worker locations is highly desirable.

Numerous techniques have been proposed to protect the privacy of user locations, such as dummy locations, k-anonymity, obfuscation methods, and differential privacy. Of these methods, differential privacy has been widely accepted because of its ability to provide rigorous privacy protection. Differential privacy ensures that no single individual, whether included or excluded from the dataset, can significantly affect the output of a query. It is normally achieved by injecting random noise into the query results. The process has already been well-implemented for location-based queries; however, many weaknesses remain in its application to spatial crowdsensing.

The current typical solution of differential privacy protects location privacy by introducing a trusted third party [5]. The third party partitions the domain of worker locations into small cells and hides each worker in a cell. The method is based on the assumption that workers are uniformly distributed within the domain. We argue this assumption is unreasonable, unless the cell size is very small. If the cell size was large, workers would likely be seen as clusters, as aggregation is a basic feature of human society; worker locations would be distributed more realistically as communities. However, this uneven distribution would cause significant errors during the task assignment process. In addition,

this existing method means the partitioning process needs to satisfy differential privacy, known as privacy spatial decomposition. Yet adding Laplace noise to each cell at each level consumes too much of the privacy budget and generates a significant volume of noise. Therefore, applying differential privacy introduces two challenges as discussed next.

The first challenge is how to more accurately measure the distance between the workers and the task. This factor is crucial for an efficient task-matching system. To preserve privacy in current crowdsensing systems, only a noisy count of the workers in each cell can be released. Therefore, the distance between a worker and a task is normally assumed to be equal to the average distance between the task and each of the four corners of the cell. If the workers are distributed uniformly in the cell, the distance measurement would be closer to reality. Hence, we propose a privacy data release method that partitions the domain of worker locations based on worker density and ensures the distribution of workers within a cell is as uniform as possible.

The second challenge is guaranteeing the success rate of task assignment while reducing system overhead. In this paper, system overhead refers to the distance workers must travel to complete a task and the number of workers who are notified. Under the veil of differential privacy, an exact count of workers in each cell cannot be released to the Server, so the Server cannot be sure of the exact number of workers that are notified about a task. In fact, it is possible for there to be no workers in a cell. As a result, the Server needs to allocate tasks to a large number of workers within a cell to guarantee a task assignment success rate, and this can increase system overheads. To solve this problem, we propose a two-pronged approach. First, a privacy budget is assigned to each cell when releasing the data. This reduces the noise and increases the accuracy of the released data. Second, to balance the task assignment success rate and the system overhead, the model is constructed by solving a geocast region optimization problem. The proposed method takes both travel distance and the number of notified workers into account, balancing the task assignment success rate and system overhead very well.

Overall, this paper makes the following contributions:

1) We propose a privacy protection data release method based on worker density that achieves differential privacy. The sanitized data is able to accurately represent the original distribution of the data, which contributes to a high task assignment success rate.
2) We introduce a geocast region selection method, which ensures highly efficient task assignments and adequately balances task assignment success rates with system overheads.

The rest of the paper is organized as follows. In Section II, we introduce the preliminaries. We propose our privacy crowdsensing method and the theoretically analyze privacy and utility in Sections III and IV, respectively. Section V details the results of the experiments. Section VI discusses related work, and Section VII concludes the paper.

II. PRELIMINARIES

In this section, we present a typical privacy framework for mobile crowdsensing, followed by the basic concepts of differential privacy.

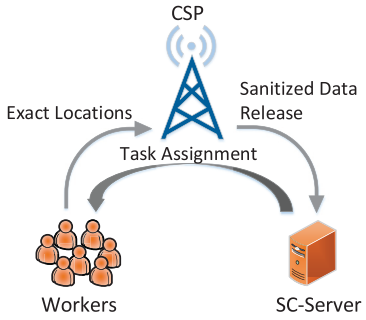


FIGURE 1. A framework for private spatial crowdsensing.

A. FRAMEWORK

Fig. 1 shows the private framework for spatial crowdsensing, which includes three entities: the workers, the cell service provider (CSP) and the Server.

Workers: The workers are the participants who are actively involved in collecting and contributing data. Workers must submit their location to the CSP, travel to the location designated for the task and collect data using their sensor-equipped device.

CSP: The CSP collects locations from workers and releases data in sanitized form to the Server for task assignment. The CSP has a signed agreement with the workers through a service contract, so a trust relationship exists between the CSP and the workers.

The Server: The Server queries the CSP for a sanitized dataset once it receives a task. It then assigns the task to suitable workers, through the CSP, according to a task assignment algorithm. The algorithm helps the Server choose appropriate workers, balancing a high task assignment success rate with a low system overhead.

B. DIFFERENTIAL PRIVACY

Differential privacy is a provable privacy notation, provided by Dwork *et al.* [6] that has emerged as an important standard for preserving privacy in a variety of areas.

Definition 1 (ϵ-Differential Privacy): A randomized algorithm $\mathcal{M}$ gives ϵ-differential privacy for any pair of *neighboring datasets* D and D^*, and for every set of outcomes Ω, $\mathcal{M}$ satisfies

$$Pr[\mathcal{M}(D) \in \Omega] \leq \exp(\epsilon) \cdot Pr[\mathcal{M}(D^*) \in \Omega] \quad (1)$$

The definition gives a strong guarantee that the presence or absence of an individual will not significantly affect the final output of the query.

Definition 2 (Global sensitivity): For a query $Q : D \to \mathbb{R}$, the global sensitivity of Q is defined as follow:

$$GS = max_{D,D'} \|Q(D) - Q(D')\|_1. \quad (2)$$

*Definition 3 (*Laplace *mechanism):* Given a function f: $D \to \mathbb{R}$ over a dataset D, 3 provides ϵ-differential privacy.

$$\widehat{f}(D) = f(D) + Laplace(\frac{S}{\epsilon}). \tag{3}$$

A *Laplace* mechanism is used for numeric output. Differential privacy is achieved by adding *Laplace* noise to the true answer.

III. MOBILE CROWD SENSING UNDER DIFFERENTIAL PRIVACY PROTECTION

A. PROBLEM DEFINITION AND ASSUMPTIONS

1) NOTATIONS

Let D denote the domain of all worker location. $SD = \{c_1, c_2, \ldots, c_m\}$ is the spatial decomposition result of D. $SSD = \{r_1, r_2, \ldots, r_m\}$ is the sanitized version of C. Given a task t, $d_{w,t}$ represents the distance between a worker and the task, then the $p_{w,t}$ is worker w_i's acceptance rate.

Further notations are detailed in Table 2:

TABLE 1. Notations.

Parameter	Description
$d_{w,t}$	Distance between a worker and a task
$d_{c,t}$	Distance between a cell and a task
d_{mtd}	Maximum travel distance
p_w	Acceptance probability of a worker
p_c	Task assignment success rate with a cell
p_{ar}	Acceptance rate
ESR	Expected success rate

2) PROBLEM DEFINITION

Consider a location privacy problem in a crowdsensing system during the process of task assignment, and we consider the server assigned tasks model where the workers need to report their locations to the server, and the server will assign the task to appropriate workers. However, the Server may be untrusted. In typical privacy-preserving crowdsensing architectures, as shown in Fig. 1, workers submit their location to the CSP, the CSP applies an appropriate privacy protection method, and releases sanitized statistical data to the Server. Our goal is to design a data release method that accurately represents the distribution of the workers and helps the Server efficiently match workers with tasks without compromising the privacy of their locations. In addition, we need to develop a geocast region construction method that allows the Server choose appropriate workers based on a sanitized dataset, resulting in high task assignment success rate and a low system overhead.

3) ASSUMPTIONS

To clarify the problem, a few assumptions are necessary. First, we assume the Server is malicious; the participants do not trust the Server. Second, we assume the CSP is trusted and will not disclose worker location information.

B. SANITIZED DATA RELEASE

The basic idea of private data release is that the domain of worker locations is partitioned into small cells and Laplace noise is added to the count of workers in each cell to achieve a differential privacy guarantee.

1) DENSITY-BASED PARTITION

Previous literature assumes the worker locations are distributed uniformly, and the workers in each cell have the same acceptance rate, which is not the case in real-world scenarios. Partitioning the data domain into a uniform grid would result in sizeable errors. Therefore, we propose a recursive partitioning process based on worker density. The aim is to identify dense regions and sparse regions and make the distribution of the workers in each smaller region as near to uniform as possible. Multiple space-partitioning data structures can assist with this process. For the purposes of this paper, we used quadtree for the partitioning as it makes a good trade off between utility and efficiency.

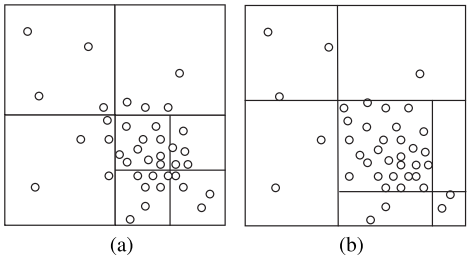


FIGURE 2. Partition data domain. (a) The standard quadtree. (b) The density based quadtree.

Traditional quadtrees recursively subdivide cells into four equal-sized subcells until the cell satisfies a stop condition. A cell becomes a leaf node if it can no longer be divided. In a data domain, this represents a region. Fig. 2 shows the traditional quadtree method. Note that the midpoint is always chosen to partition the parent cell. The drawback of this methods is that the partition is data-independent. Workers may be clustered together in a small area of the cell, which could reduce the accuracy of the data release. However, that problem can be solved by applying the quadtree technique in a data-dependent way, i.e., by partitioning the cells according to the density of the workers as shown in Fig. 2b.

First, several initial partition points in the location domain need to be selected. The differences in density between the subcells partitioned by each partition point are calculated, and the subcells with the biggest differences in density are then chosen as partitions. This process is repeated for each subcell until the stop condition is met. Algorithm 1 presents the details of this density-based partitioning process.

a: STOP CONDITION

The stop conditions are very important in the partitioning process, as they have an important effect on the assignment success rate. Traditional quadtrees require the data publisher to specify the height of the partitioning. It is difficult to

Algorithm 1 Density Based Partitioning

Require: Dataset W
Ensure: Spatial decomposition SD.

1: $SD = \phi$;
2: $cell = W$;
3: $m = \sqrt{S_{cell}}/\alpha$;
4: $n \leftarrow$ number of workers in cell;
5: **if** $n < 1 \parallel m <= 1$ **then**
6: $SD = SD \cup cell$;
7: **else**
8: Generate m partition points randomly within domain;
9: **for** $i = 1$ to m **do**
10: Subcells set $C \leftarrow$ partition cell;
11: **for** each cell $c_j \in C$ **do**
12: Calculate the workers density in cell c_j;
13: **end for**
14: Calculate $\Delta d_i = max\{den(c_j)\} - min\{den(c_j)\}$;
15: **end for**
16: **if** $max\{\Delta d\} > \beta$ **then**
17: Partition the cell at the point with biggest Δd into four subcells $C = \{c_1, c_2, c_3, c_4\}$;
18: **for** $c_i \in C$ **do**
19: $cell = c_i$;
20: Go to step 3;
21: **end for**
22: **else**
23: Determine whether the cell needs to be partitioned further by calculating $m' = \lceil \sqrt{\frac{n}{\sqrt{2}}} \rceil$;
24: **if** $m' > 1$ **then**
25: Partition cell to $m' \times m'$ subcells c_i, $1 \leq i \leq m$;
26: $SD = SD \cup c_i$, $1 \leq i \leq m$;
27: **end if**
28: **end if**
29: **end if**
30: **return** SD

calculate an effective height with non-uniform partitioning, so we defined three stop conditions for this scenario to improve efficiency and utility.

- If no workers exist in the cell, no further partitioning is needed as that cell cannot contribute to the task. Therefore, the cell is marked as a leaf node, as shown in Steps 3 to 6.
- If a cell is too small to be further partitioned, a stop condition is met. The parameter α in Step 2 controls the area of this cell ($S_{cell} <= \alpha^2$). The smaller the cell, the more uniform the distribution of workers within it.
- If the distribution of workers in a cell is relatively uniform, a stop condition is also met. We use maximum density difference Δd to measure whether the worker location distribution is uniform.

b: PARTITION POINT

The selection of the partition point directly affects the results of partitioning. It decides whether the distribution of workers in each cell is uniform. Therefore, m initial partition points are randomly generated within the cell. The parameter m is decided by the area of the cell being partitioned. Step 3 shows the calculation method. The intent is to find the best partitioning point that can divide the cell into four subcells with maximum density difference from the initial partition points. Therefore, more initial partitioning points mean more accurate segmentation.

All initial partition points are denoted as $p \in \{p_1, p_2, \ldots, p_m\}$. The score function for selecting each partition point is evaluated by the density difference, which is calculated as follow:

$$\Delta d(cell, p) = \max_{c_i \in C}\{den(c_i)\} - \min_{c_i \in C}\{den(c_i)\}. \quad (4)$$

Steps 9 to 15 calculate all the density differences based on the partition points. The partition point p with biggest density difference is chosen as a candidate. If the biggest density difference Δd_p is greater than the threshold β, the cell is partitioned at point p (Steps 16 and 17). The entire process is repeated for the partitioned subcells until no further cells can be partitioned (Steps 18 to 21). Otherwise, the cell will not be partitioned as the distribution of worker in the cell is already close to uniform. Yet even after this process, the number of workers in a cell may still be large, adding to the system overhead. Therefore, Step 23 determines whether the cells need to be further partitioned into smaller cells with fewer workers. If $m' > 1$, the cell is partitioned into a smaller one of equal size in Step 25 and add them to SD in Step 26.

2) DIFFERENTIAL PRIVACY DATA RELEASE

As previously mentioned, a noisy count of the number of workers in each cell is released to protect the privacy of worker locations. Algorithm 2 shows the details of this release.

Algorithm 2 Differential Privacy Data Release

Require: Spatial decomposition SD
Ensure: Sanitized data SSD

1: **for** $c_i \in SD$ **do**
2: $n_i \leftarrow$ number of workers in c_i;
3: $N_i = n + Laplace(\frac{s}{\epsilon})$;
4: **end for**
5: **return** $SSD = \{r_1, r_2, \ldots, r_m\}$

First, Step 2 calculates the number of workers in each cell, then Laplace noise is added to the count in Step 3. In Step 5, the sanitized SD, say SSD is released. r_i represents a region with a sanitized count of workers in the cell. According to the definition of differential privacy, whether or not a worker within a specific cell cannot be identified. Therefore, worker location privacy is preserved.

C. TASK ASSIGNMENT

When the server receives the sanitized data, it determines a geocast region GR to disseminate the task to the workers in GR. The goal is to reach an expected task assignment

success rate, while reducing system overhead at the same time, such as the distance workers need to travel and the number of workers notified of the task.

1) WORKER ACCEPTANCE PROBABILITY

The distance a worker has to travel to complete a task is an important issue to consider in task allocation because it has a significant impact on both worker acceptance probability and the task assignment success rate. Not only may workers be unwilling to accept tasks with long travel times but organizers might also have to pay higher incentives to workers who are further away. Therefore, worker acceptance probability p_w as is modeled as a function of distance $d_{w,t}$, as follows:

$$p_w = f(d_{w,t}). \tag{5}$$

Two cases are considered. In the first case, a worker's acceptance probability decreases linearly with an increase in the distance between her location to the task location, as shown in 6.

$$f(d_{w,t}) = \begin{cases} \frac{d_{mtd} - d_{w,t}}{d_{mtd}}, & d_{w,t} \le d_{mtd} \\ 0, & d_{w,t} > d_{mtd} \end{cases}. \tag{6}$$

where d_{mtd} is the maximum distance that most workers will travel.

In the second case, we use the nonlinear hyperbolic tangent function [7], a nonlinear function.

$$y(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}. \tag{7}$$

with the property $y \in [0, 1)$, when $x \ge 0$. The acceptance probability is defined as:

$$f(d_{w,t}) = \begin{cases} y(\frac{c}{d_{w,t}}) & d_{w,t} \le d_{mtd} \\ 0, & d_{w,t} > d_{mtd} \end{cases}. \tag{8}$$

where c is the parameter that regulates drops in the acceptance rate with increase in the travel distance.

Assuming there are n workers in a cell, the probability that workers in the cell will accept a task is

$$p_c = 1 - (1 - p_w)^n. \tag{9}$$

2) GEOCAST REGION SELECTION

There are two important standards when selecting the geocast region. First, the worker's travel distance should be short. Second, the worker acceptance rate over the geocast region should achieve the expected task assignment success rate. Although the acceptance probability is based on the travel distance, we cannot say the cell with higher acceptance probability has a shorter distance to the task location. Assume there are two split cells A and B, and the distances between these two cells and the task location l_t are $d_{a,t}$ and $d_{b,t}$, also $d_{a,t} > d_{b,t}$. If cells A and B contain the same number of workers, cell B has a greater acceptance probability. However, if there are more workers in cell A, it is possible that the acceptance probability of A is greater than B. Therefore, the problem of geocast region selection can be formalized as follows:

- The number of notified workers should be as small as possible, and the worker's travel distance should be as short as possible.
- The acceptance probability of the geocast region should reach the expected task assignment success rate.
- The distance between the selected cell and the task should be within the maximum travel distance of the workers.

To achieve our objective, we propose the geocast region selection method shown in Algorithm 3.

Algorithm 3 Geocast Region Selection

Require: *SSD*
Ensure: *GR*
1: Order $SSD = \{r_1, r_2, \ldots, r_n\}$, where $d_{r_1,t} \le d_{r_2,t} \le \ldots \le d_{r_n,t}$
2: Choose r_1 as the initial geocast region *GR*.
3: **repeat**
4: Expanding *GR* by adding the closest cell in the remaining cells one by one;
5: **until** $p_{ar} < ES$ or $d_{r_i,t} > d_{mtd}$
6: Calculate the expectation of travel distance: $Ed = \sum_{i=1}^{m} p_{r_i} d_{r_i,t}$;
7: Calculate the number of workers in *GR*: $N = \sum_{i=1}^{m} n_{r_i}$;
8: $S \leftarrow$find c_i that $p_{r_i} > ES$, $n_{r_i} \le N$ and $d_{r_i} \le d_{mtd}$;
9: **if** $S \ne \emptyset$ **then**
10: **for** $r_i \in S$ **do**
11: find the cell r_i has the shortest distance to l_t;
12: **end for**
13: **if** $p_{r_i} d_{r_i,t} < Ed$ **then**
14: $GR = r_i$;
15: **end if**
16: **end if**
17: **return** *GR*

As shown in Step 1, the partitioned cells are sorted in increasing order according to the distance to the task. Initially, the closest cell to the task is chosen as the first *GR* in Step 2. If the acceptance probability does not reach expectations, the *GR* continues to expand by adding the closest cell to the task from the remaining cells until the acceptance probability reaches the expected goal or the cell's distance is beyond the maximum travel distance d_{mtd}, as shown in Steps 3 to 5. This method ensures the worker's travel distance is short. However, reducing the number of notified workers requires some exploration. A cell that can satisfy the expected task assignment success rate with the best balance between distance with worker numbers needs to be found. As it is not known which users will accept the task at this stage, Step 6 estimates the travel distance as an expectation of the distance of the selected cells, while Step 7 calculates the number of notified workers. Step 8 locates the cells with an acceptance probability higher than the expected acceptance probability and with fewer workers. If such cells exist, and their distance is under the threshold d_{mtd}, the cell $r_i \in S$ with the shortest distance is chosen as the candidate. If the

expected travel distance of $p_{r_i} d_{r_i,t} < Ed$, the candidate r_i is chosen as the geocast region (Steps 9 to 16).

IV. PRIVACY AND UTILITY ANALYSIS

A. PRIVACY ANALYSIS

User location information is preserved by hiding it in partitioned cells, and the sanitized data is released by adding Laplace noise to the statistical results of each cell. Theorem 1 shows that the proposed data release method satisfies ϵ-differential privacy.

Theorem 1: For a given dataset D, each record represents a user's location information, and the records are independent of each other. The proposed privacy preserving method can provide ϵ- differential privacy.

Proof: Assume the proposed method partitions the map into m disjoint cells. A set of Laplace mechanisms $\{M_1, M_2, \ldots, M_m\}$ are performed on each cell, and the assigned privacy parameter for each cell is ϵ_i. Each cell satisfies ϵ_i-differential privacy. The composite properties of the privacy budget are applied to the whole dataset to analyze the privacy guarantee, which is defined below.

Theorem 2 (Parallel Composition [8]): Suppose we have a set of privacy mechanisms $M = \{M_1, M_2, \ldots, M_m\}$, and each M_i provides ϵ_i privacy guarantee on a disjoint subset of the entire dataset, M provides $max(\epsilon_i)-$ differential privacy.

Theorem. 2 can be used to directly analyze the privacy guarantee of the proposed method. As mentioned earlier, assume the assigned privacy parameter for each cell is ϵ_i, and the cells are disjoint and independent of each other. According to Theorem. 2, the set of privacy mechanisms $\{\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_m\}$ will consume the $max\{\epsilon_1, \epsilon_2, \ldots, \epsilon_m\}$ privacy budget. In the proposed method, we assign each cell the same privacy budget ϵ; therefore, the proposed method preserves ϵ-differential privacy. □

B. UTILITY ANALYSIS

In this section, we apply a well-known utility definition suggested by Blum *et al.* [9] to measure prediction accuracy.

Definition 4 ((α, β)-useful): A database access mechanism $\mathcal{M}$ is (α, β)-useful with respect to count query, if for every database D, with a probability of at least $1-\beta$, the output of the mechanism $\mathcal{M}$ satisfies

$$Pr[max|\mathcal{M}(\hat{cell}_i) - \mathcal{M}(cell_i)| \leq \alpha] \geq 1-\beta. \quad (10)$$

Theorem 3: The output error of the count query on each cell caused by the proposed method is less than α with a probability of at least $1-\beta$. The proposed method is satisfied with (α, β)-useful when $\alpha \leq -\frac{sln2\beta}{\epsilon}$.

Proof: The error caused by the proposed method is only the noise, and is denoted as λ, and $\lambda \sim Laplace(\frac{s}{\epsilon})$.

Therefore,

$$\begin{aligned} &Pr[max|\mathcal{M}(\hat{cell}_i) - \mathcal{M}(cell_i)| > \alpha] \\ &= Pr[Laplace(\frac{s}{\epsilon}) > \alpha] \\ &= \int_{\alpha}^{\infty} \frac{1}{2b} e^{-\frac{x}{b}} dx. \end{aligned} \quad (11)$$

Let $\int_{\alpha}^{\infty} \frac{1}{2b} e^{-\frac{x}{b}} dx = \beta$, we have

$$\begin{aligned} &\int_{\alpha}^{\infty} e^{-\frac{x}{b}} dx = 2b\beta \\ &\Rightarrow -be^{-\frac{\alpha}{b}}|_{\alpha}^{\infty} = 2b\beta \\ &\Rightarrow be^{-\frac{\alpha}{b}} = 2b\beta \\ &\Rightarrow -\frac{\alpha}{b} = ln2\beta \\ &\Rightarrow \alpha = -bln2\beta. \end{aligned} \quad (12)$$

As $b = \frac{s}{\epsilon}$, therefore, $\alpha = -\frac{sln2\beta}{\epsilon}$. That is, when $\alpha \leq -\frac{sln2\beta}{\epsilon}$, the error introduced by the privacy operation is controlled within α with a high probability. □

V. EXPERIMENT EVALUATION

We evaluated the performance of our method through an extensive set of experiments. First, the experimental settings are presented, followed by a discussion of the results.

A. EXPERIMENT SETUP

Dataset: We used two real-world datasets.

1) *SimpleGeo Places Dataset [10]:* This dataset contains information on more than 20 million places in 63 countries around the world. We extracted 8275 business entries for the most populous city in Australia, Sydney. We randomly chose 1000 locations as tasks, and the rest of the locations were used as workers.
2) *Yelp Dataset [11]:* The Yelp dataset includes a business dataset, a check-in dataset, a user dataset, and so on. We used the business dataset, which includes information about local businesses in 11 cities across 4 countries. We chose the businesses located in Las Vegas, using the restaurant locations as workers and 1000 random shopping locations as the tasks.

Metrics: The effectiveness of the proposed method can be evaluated by the success rate and efficiency of the tasks assignment. Therefore, we use the following metrics:

1) *Task Assignment Success Rate:* Let $T = \{t_1, t_2, \ldots, t_m\}$ be a set of tasks. Each task was assigned to a group of workers to be accepted with a specific probability. Assume there are n tasks to be confirmed by the workers; the task assignment success rate can be represented as follows

$$TASR = \frac{n}{|T|}, \quad (13)$$

where $|T|$ is the number of tasks.

2) *Average Travel Distance:* Assume $T_s = \{t_1, t_2, \ldots, t_n\}$ is a successfully allocated task set, and the set $W = \{w_1, w_2, \ldots, w_n\}$ are the corresponding workers who performed the task. Then

$$ATD = \frac{\sum_{t_i \in T_s, w_i \in W} d(t_i, w_i)}{|T_s|}, \quad (14)$$

where $d(t_i, w_i)$ is the Euclidean distance between the task and the worker, $|T_s|$ is the number of tasks that assigned successfully.

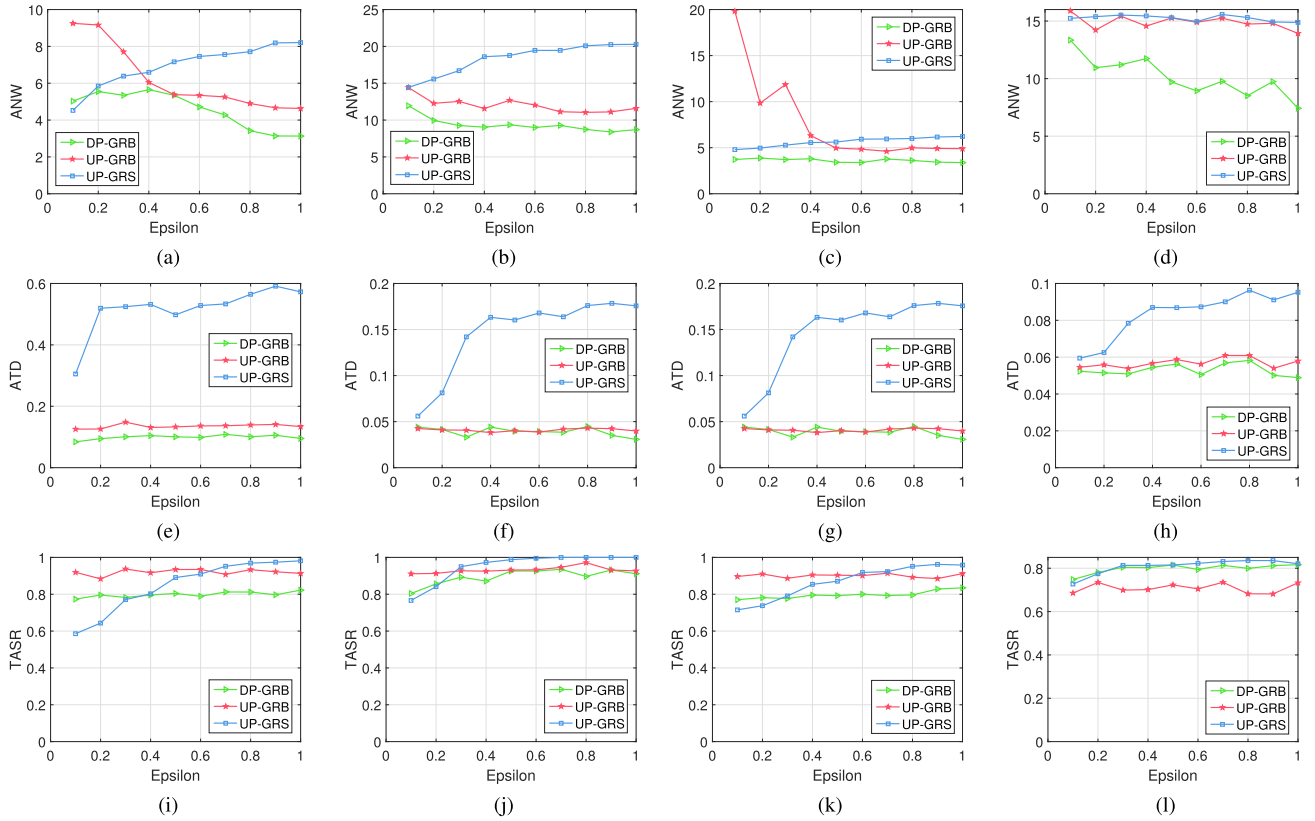


FIGURE 3. Performance by varying ϵ. (a), (e), and (i) Yelp-linear. (b), (f), and (j) SimpleGeo-linear. (c), (g), and (k) Yelp-nonlinear. (d), (h), and (l) SimpleGeo-nonlinear.

3) *Average Notified Workers:* For each task $t_i \in T$, there were n_i workers are notified about the task. We calculated the average number of notified workers as follows:

$$ANW = \frac{\sum_{i=1}^{|T|} n_i}{|T|}, \tag{15}$$

where $|T|$ is the number of tasks.

Comparison: We compared our method with the uniform partition method, which was proposed by To et.al. [5]. They proposed partitioning the space in sparse regions using a two-level grid by modifying the state-of-the-art adaptive grid method. All the partition processes were uniform across the data domain. We considered two scenarios within a uniform partition during the geocast region selection process. First, the cell with the closest distance to the task was preferred when expanding the geocast region. Second, the cell with the highest acceptance rate was chosen at each step in the construction of the geocast region. In the experiment, we considered both linear and nonlinear acceptance probabilities.

Parameters: Table 2 shows the parameter settings of our experiment; the default values are highlighted.

The privacy budget is a very important parameter. It determines how much noise is added to the released dataset, which affects utility. We set the privacy budget to $\epsilon \in [0.1, 1.0]$, and show the changes in performance. The default value was 1.0. The worker's maximum travel distance was changed from 1 *km* to 5 *km*, and the expected task assignment success rate was varied from 0.3 to 0.9.

TABLE 2. Parameter setting.

Parameter	Description	Value
ϵ	Privacy budget	$0.1 - 1.0$
MTD	Maximum travel distance	$1km - 5km$
ESR	Expected success rate	$0.3 - 0.9$

B. EXPERIMENT RESULTS

1) PERFORMANCE BY VARYING ϵ

We examined the performance of the three methods in relation to the different privacy budgets ϵ for an assigned task in terms of ANW, ATD, and TASR. We varied the privacy budget ϵ between 0.1 and 1 on both datasets using the linear and nonlinear acceptance rates. DP-GRB refers to the proposed method. UP-GRB represents the method with uniformed partitions and a balanced geocast region construction. UP-GRS gives priority to the task assignment success rate.

a: PERFORMANCE ON ANW

Figs. 3a-3d show the results in terms of ANW. We observed that ANW decreased as the privacy budget ϵ increased for both DP-GRB and UP-GRB with a reverse trend for UP-GRS. This is because a smaller privacy budget ϵ means more noise

needs to be added to each cell; therefore, more workers need to be notified to achieve the expected success rate for both DP-GRB and UP-GRB. Correspondingly, when the privacy budget ϵ is increased, less noise needs to be added to each cell, which means more workers need to be selected to achieve a higher task assignment success rate. In addition, we observed that our method always outperformed the other two methods, which have lower ANWs in all configurations. Specifically, as shown in Figs. 3a and 3b, when $\epsilon = 0.3$, our method, with linear acceptance rates, achieved an ANW of 5.3431 and 9.2615 for the Yelp and SimpleGeo datasets, respectively, UP-GRB achieved 7.7044 and 12.5312, an increase of around 50% and 30%, respectively and UP-GRS achieved 6.3847 and 16.7064, an increase of around 20% and 80%, respectively. When $\epsilon = 0.8$, UP-GRB and UP-GRS achieved ANWs of 4.8993 and 7.7132 for the Yelp dataset and 11.0441 and 20.0814 for the SimpleGeo dataset, which are much larger than the ANW values of 3.4231 and 8.7352 of our method. A similar observation was found in Fig. 3c and Fig. 3d. This is because the distribution of workers in each cell is not uniform in the UP-GRB method; however, their acceptance rates are considered to be the same, which causes some errors. Conversely, the proposed method partitions the worker domain based on worker density, which makes worker distribution in each cell close to uniform. That helps to choose more accurate cells for task assignment. Because UP-GRS always chooses the cell that produces the highest success rate at each step, more workers are needed to achieve a higher success rate.

b: PERFORMANCE ON ATD

Figs. 3e-3h show the change in ATD with a varied privacy budget. We observed that the ATD does not significantly increase with a reduced privacy budget in either our method or UP-GRB. However, the privacy budget had a significant effect on ATD for UP-GRS when $\epsilon < 0.4$. This proves that the proposed GR construction method did a good job in selecting which cells to balance the assignment of tasks and system overhead. Additionally, the added noise had a significant effect on the cell selection when achieving a high ASR. The UP-GRS method had a greater ATD compared to the other two methods under all configurations. This is because the construction of UP-GRS prefers to choose the cells with a higher utility regardless of the distance to the task, as long as the task is within the worker's maximum travel distance. In addition, we observed that the ATD value of our method was always lower than UP-GRB, which means tasks can be completed within shorter distance using our method. Specifically, our method achieved an ATD of around 1km for the Yelp dataset with a linear acceptance rate, as shown in Fig. 3e, and it outperformed the UP-GRB method by approximately 300 m. Fig. 3g shows the results for the Yelp dataset with a nonlinear acceptance rate. Our method achieved an ATD of around 0.12 km, while UP-GRB achieved around 0.145 km, which is an increase of 250 m. Figs. 3f and 3f show the results on the SimpleGeo dataset. The performance of the three methods are similar to the results for the Yelp dataset.

c: PERFORMANCE ON TASR

The TASR values corresponding to the different methods used on both the Yelp and SimpleGeo datasets are shown in Figs. 3i, 3j, 3k, and 3l. It is clear that the TASR values achieved by our method are basically around 0.8, which is the expected success rate (*ESR*). UP-GRB achieved a TASR of around 0.9, 10% higher than expected. The TASR achieved by UP-GRS significantly increased as the privacy budget increased. Specifically, as shown in Fig. 3i, when $\epsilon = 0.1$, our method achieved a TASR of 0.7734, only 3% below the *ESR*, while UP-GRB achieved a TASR of 0.9190, approximately 11% higher than the *ESR*. UP-GRS achieved a TASR of 0.5858, which is a decrease of about 22% compared to the *ESR*. When $\epsilon = 0.8$, our method and UP-GRB achieved a TASR of 0.8119 and 0.9335, respectively, which is slightly greater than the TASR achieved by both methods when $\epsilon = 0.1$. This indicates that a greater privacy budget means a higher task assignment success rate. UP-GRS achieved a TASR of 0.9690, which is much higher than the other two methods. The performance of the three methods in terms of TASR with nonlinear acceptance rates is shown in Fig. 3k. Similar to the result shown in Fig. 3i, the TASR achieved by our method was around 0.8, and UP-GRB achieved a TASR of around 0.9, which was much higher than expected. The TASR value changed significantly when the privacy budget ϵ varied. Figs. 3j and 3l show the results for the SimplGeo dataset. We were able to observe that the higher TASR was at the cost of increased overhead more notified workers and a longer travel distance to the task destination. Our method achieved a good trade off, achieving the *ESR* while reducing the number of notified workers and their travel distance.

2) PERFORMANCE BY VARYING MTD

We evaluated the performance of the proposed method on both datasets by varying the maximum travel distance (MTD). Fig. 4 shows the results when the acceptance rate has a linear distribution. The results with a nonlinear distribution show similar performance. We observed that when the MTD was small, more workers were required to achieve the *ESR*. For example, as shown in Fig. 5a, with $\epsilon = 0.5$, when $MTD = 1\ km$, more than 8 workers were needed to guarantee an 80% success rate on the Yelp dataset. While only around 4 workers were sufficient when the maximum travel distance was increased to 5 km. This is because a worker has a higher probability of accepting a task at a fixed distance when the maximum travel distance is longer. Meaning, fewer workers are needed to achieve the *ESR*. Fig. 5b shows a similar trend when increasing the MTD for the SimpleGeo dataset. We also observed that changing the MTD had little effect on the ATD, irrespective of the dataset, which is shown in Figs. 5c and 5d, respectively. The value of ATD basically remained the same, especially, when the added noise was smaller. This trend affects the MTD's influence on ATD.

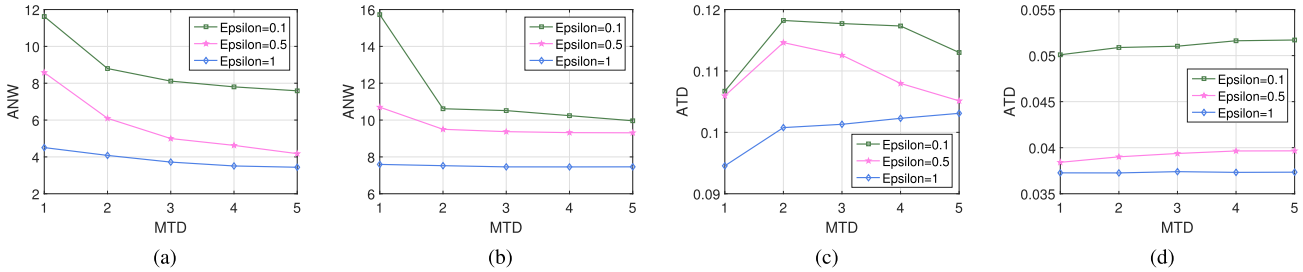


FIGURE 4. The performance by varying MTD. (a) and (c) Yelp-linear. (b) and (d) SimpleGeo-linear.

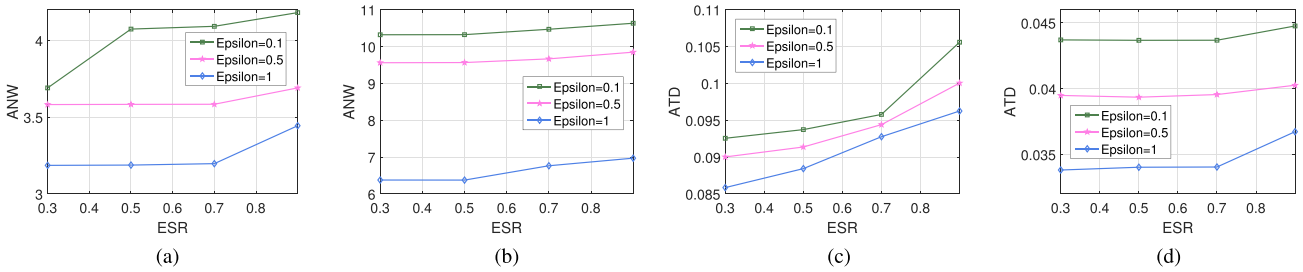


FIGURE 5. The performance by varying ESR. (a) and (c) Yelp-linear. (b) and (d) SimpleGeo-linear.

3) PERFORMANCE BY VARYING ESR

The variations in the tendencies of ANW and ATD for the Yelp and SimpleGeo datasets along with the parameter's ESRs are shown in Fig. 5. We observed that a higher ESR results in both higher ANW and ATD. Fig. 5a shows the effect of ESR on ANW for the Yelp dataset with linear acceptance rates. We can observe that when $\epsilon = 0.5$, and $ESR0.7$, 3.6 workers are enough to achieve a 0.7 success rate. To achieve a higher ESR, more workers need to be notified. Fig. 5b shows similar results for the SimpleGeo dataset in terms of ANW. Figs. 5c and 5d indicate the impact of increasing the ESR for ATD. When the $ESR = 0.3, \epsilon = 1$, a travel distance of 0.085 km was needed to finish the task in Fig. 5c. However, when the $ESR = 0.9$, the travel distance was increased to 0.096 km. This is because obtaining a higher task assignment success rate requires more cells to construct a larger geocast region, which leads to an increased travel distance as well as an increase in the number of notified workers.

VI. RELATED WORK

Location privacy has been studied extensively. For example, dummy locations [12] were proposed to protect user locations by adding false positions to the true locations; cloaking region techniques [13] transform the exact location to a sufficiently larger region to reduce location precision; the transformation method [14] performs some basic geometric operations over a user's location; private information retrieval [15] uses encryption to protect a user's location; and differential privacy-based perturbation methods [16] have also been proposed.

These techniques have largely been used and studied in location-based services. However, only a few studies focus on crowdsensing [17]. Kazemi and Shahabi [18] presented a privacy framework in which each participant forms their own cloaked region by computing a Voronoi cell in a distributed fashion. Then, a voting mechanism is devised to select the set of representative participants and send their cloaked regions to the server. The query results are subsequently shared with the rest of the participants. Similar to the method proposed by Kazemi *et al.*, Hu *et al.* [19] employed a peer-to-peer cloaking technique to cloak worker locations among $k - 1$ other workers. In addition, Bin *et al.* [20] presented a clustering method in which the location of the virtual cluster center is reported to the server by the cluster head. Once the cluster head receives the task from the server, tasks are assigned to the chosen cluster member according to their exact location. However, none of these obfuscation-based techniques provide a rigorous privacy guarantee. Their reliability is highly dependent on an adversary's background knowledge. Once the attacker obtains a key piece of background knowledge, such as a location the user visits frequently, the user's location can easily be inferred. Shen *et al.* [21] applied an encryption technique and proposed a privacy framework that performs worker task matching in an encrypted domain. In particular, they introduced a semi-trusted third party to provide privacy functionality and collect encrypted data from workers. The server communicates with the third party in the encrypted domain to find workers at a minimum cost. The advantage is that it can provide a strong privacy guarantee. However, encryption-based technology is often computationally and communicationally expensive.

Differential privacy is a powerful privacy model that satisfies privacy regardless of the attacker's background knowledge. It is also less computationally expensive. With this

privacy definition, To *et al.* [5] proposed a privacy-aware framework to protect user's location information in spatial crowd-sourcing by introducing a cellular service provider (CSP) as a trusted third party. The CSP collects workers' locations and then partitions the entire spatial region into a grid of indexed cells by applying the CSP's partition algorithm. Laplace noise is added to the count of each cell, and the sanitized data are released to the service provider. Latter, they also presented a tool box [22] to display the framework in a visual, interactive environment. Their recent work [23] was extended in a further solution, addressing dynamic worker datasets [5] that investigate privacy budget allocation techniques across consecutive releases and employ post-processing based on Kalman filters to improve the accuracy. Yanmin *et al.* [24] proposed a similar differential privacy framework for task assignment in ad hoc mobile clouds. They not only consider location privacy but also service quality, which considers the mobile servers' reputation. In addition, Wang *et al.* [25] propose a location privacy-preserving task allocation framework with geo-obfuscation to protect users' locations during task assignments, which make participants obfuscate their reported locations under the guarantee of differential privacy. Xiong *et al.* [26] presented a differentially private allocation mechanism for reward-based spatial crowdsourcing. They presented a contour plot to characterize location distribution and proposed an optimized-reward allocation method to achieve a specified probability of assignment success.

VII. CONCLUSION

In this paper, we proposed a privacy-preserving data release method based on worker density. This method satisfies differential privacy and enables workers to participate in crowdsensing platforms without disclosing their location. In addition, the proposed method improves the accuracy of the released data. We also proposed an optimal geocast region selection strategy that considers the distance workers must travel and the number of workers that are notified of available tasks. The proposed geocast region selection strategy not only achieves the expected task assignment success rate but also reduces system overhead. We evaluated the performance through extensive experiments, and the results prove that our method achieves a better balance between task assignment success rate and system overhead with the same privacy guarantee.

REFERENCES

[1] A. Antonić, M. Marjanović, K. Pripužić, and I. P. Žarko, "A mobile crowd sensing ecosystem enabled by CUPUS: Cloud-based publish/subscribe middleware for the Internet of Things," *Future Generat. Comput. Syst.*, vol. 56, pp. 607–622, Mar. 2016.

[2] R. K. Rana, C. T. Chou, S. Kanhere, N. Bulusu, and W. Hu, "Ear-phone: An end-to-end participatory urban noise mapping system," in *Proc. IPSN*, Stockholm, Sweden, 2010, pp. 105–116.

[3] P. Mohan, V. N. Padmanabhan, and R. Ramjee, "Nericell: Rich monitoring of road and traffic conditions using mobile smartphones," in *Proc. Sensys*, Raleigh, NC, USA, 2008, pp. 323–336.

[4] Y. Chon, N. D. Lane, F. Li, H. Cha, and F. Zhao, "Automatically characterizing places with opportunistic crowdsensing using smartphones," in *Proc. Unicomp*, Pittsburgh, PA, USA, 2012, pp. 481–490.

[5] H. To, G. Ghinita, and C. Shahabi, "A framework for protecting worker location privacy in spatial crowdsourcing," *Proc. VLDB Endowment*, vol. 7, no. 10, pp. 919–930, Jun. 2014.

[6] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Proc. TCC*, New York, NY, USA, 2006, pp. 265–284.

[7] J. W. Anderson, *Hyperbolic Geometry*, Springer, Feb. 2006.

[8] F. McSherry and I. Mironov, "Differentially private recommender systems: Building privacy into the Netflix prize contenders," in *Proc. KDD*, Paris, France, 2009, pp. 627–636.

[9] A. Blum, K. Ligett, and A. Roth, "A learning theory approach to non-interactive database privacy," *J. JACM*, vol. 60, no. 2, pp. 12:1–12:25, Apr. 2013, doi: 10.1145/2450142.2450148.

[10] (2011). *SimpleGeo Public Spaces CC0 Colletion*. [Online]. Available: http://archive.org/details/2011-08-SimpleGeo-CC0-Public_Spaces

[11] *Yelp Dataset Challenge*. [Online]. Available: https://www.yelp.com/dataset_challenge

[12] T. Hara, A. Suzuki, M. Iwata, Y. Arase, and X. Xie, "Dummy-based user location anonymization under real-world constraints," *IEEE Access*, vol. 4, pp. 673–687, 2016, doi: 10.1109/ACCESS.2016.2526060.

[13] C. A. Ardagna, M. Cremonini, S. D. C. D. Vimercati, and P. Samarati, "An obfuscation-based approach for protecting location privacy," *IEEE Trans. Depend. Sec. Comput.*, vol. 8, no. 1, pp. 13–27, Jan. 2011, doi: 10.1109/TDSC.2009.25.

[14] A. Gutscher, "Coordinate transformation—A solution for the privacy problem of location based services?" in *Proc. IPDPS*, Rhodes Island, Greece, 2006, p. 7.

[15] X. Yi, R. Paulet, E. Bertino, and V. Varadharajan, "Practical approximate k nearest neighbor queries with location and query privacy," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 6, pp. 1546–1559, Jun. 2016, doi: 10.1109/TKDE.2016.2520473.

[16] M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Geo-indistinguishability: Differential privacy for location-based systems," in *Proc. CCS*, Berlin, Germany, 2013, pp. 901–914.

[17] B. Liu, W. Zhou, T. Zhu, H. Zhou, and X. Lin, "Invisible hand: A privacy preserving mobile crowd sensing framework based on economic models," *IEEE Trans. Veh. Technol.*, vol. 66, no. 5, pp. 4410–4423, May 2017, doi: 10.1109/TVT.2016.2611761.

[18] L. Kazemi and C. Shahabi, "A privacy-aware framework for participatory sensing," *ACM SIGKDD Explorations Newslett.*, vol. 13, no. 1, pp. 43–51, Jun. 2011, doi: 10.1145/2031331.2031337.

[19] J. Hu, L. Huang, L. Li, M. Qi, and W. Yang, "Protecting location privacy in spatial crowdsourcing," in *Proc. APWeb*, Guangzhou, China, Sep. 2015, pp. 113–124.

[20] B. Zhu, S. Zhu, X, Liu, Y. Zhong, and H. Wu, "A novel location privacy preserving scheme for spatial crowdsourcing," in *Proc. ICEIEC*, Beijing, China, 2016, pp. 34–37.

[21] Y. Shen, L. Huang, L. Li, X. Lu, S. Wang, and W. Yang, "Towards preserving worker location privacy in spatial crowdsourcing," in *Proc. GLOBECOM*, San Diego, CA, USA, 2015, pp. 1–6.

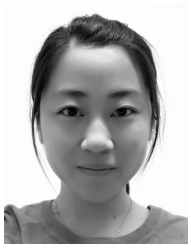
[22] H. To, G. Ghinita, and C. Shahabi, "PrivGeoCrowd: A toolbox for studying private spatial crowdsourcing," in *Proc. ICDE*, Seoul, South Korea, 2015, pp. 1404–1407.

[23] H. To, G. Ghinita, L. Fan, and C. Shahabi, "Differentially private location protection for worker datasets in spatial crowdsourcing," *IEEE Trans. Mobile Comput.*, vol. 16, no. 4, pp. 934–949, Apr. 2017, doi: 10.1109/TMC.2016.2586058.

[24] Y. Gong, C. Zhang, Y. Fang, and J. Sun, "Protecting location privacy for task allocation in ad hoc mobile cloud computing," *IEEE Trans. Emerg. Topics Comput.*, vol. 6, no. 1, pp. 110–121, Jan./Mar. 2015.

[25] 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. WWW*, Perth, Australia, 2017, pp. 627–636

[26] P. Xiong, L. Zhang, and T. Zhu, "Reward-based spatial crowdsourcing with differential privacy preservation," *Enterprise Inf. Syst.*, vol. 11, no. 10, pp. 1500–1517, Nov. 2017, doi: 10.1080/17517575.2016.1253874.



MENGMENG YANG received the B.Eng. degree from Qingdao Agricultural University, China, in 2011, and the M.Eng. degree from Shenyang Normal University, China, in 2014.

She is currently pursuing the Ph.D. degree with the School of Information Technology, Deakin University, Australia. Her research interests include privacy preserving, machine learning, and network security.

TIANQING ZHU (M'11) received the B.Eng. and M.Eng. degrees from Wuhan University, China, in 2000 and 2004, respectively, and the Ph.D. degree in computer science from Deakin University, Australia, in 2014.

She served as a Lecturer with Wuhan Polytechnic University, China, from 2004 to 2011. She is currently a Lecturer with the School of Information Technology, Deakin University, Australia. Her research interests include privacy preserving, data mining, and network security.

Dr. Tianqing has received the Best Student Paper Award in PAKDD 2014.

YANG XIANG (M'07–SM'12) received the Ph.D. degree in computer science from Deakin University, Australia.

He is the Dean of Digital Research and Innovation Capability Platform, Swinburne University of Technology, Australia. He has published over 200 research papers in many international journals and conferences. His research interests include cyber security, which covers network and system security, data analytics, distributed systems, and networking. In particular, he is currently leading his team developing active defense systems against large-scale distributed network attacks.

He is the Chief Investigator of several projects in network and system security, funded by the Australian Research Council. Two of his papers were selected as the featured articles in 2009 and 2013 issues of the IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS. Two of his papers were selected as the featured articles in 2014 issues of the IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING. He has served as the Program/General Chair for many international conferences. He has been the PC member for over 80 international conferences in distributed systems, networking, and security.

WANLEI ZHOU (SM'09) received the B.Eng. and M.Eng. degrees from the Harbin Institute of Technology, Harbin, China, in 1982 and 1984, respectively, the Ph.D. degree from Australian National University, Canberra, ACT, Australia, in 1991, all in computer science and engineering, and the D.Sc. degree from Deakin University, Melbourne, VIC, Australia, in 2002.

He is currently the Alfred Deakin Professor, the Chair Professor of information technology, and the Associate Dean of the Faculty of Science, Engineering and Built Environment with Deakin University. He served as a Lecturer with the University of Electronic Science and Technology of China, a System Programmer with Hewlett Packard, Boston, MA, USA, and a Lecturer with Monash University, Melbourne, VIC, Australia, and the National University of Singapore, Singapore. He has published over 300 papers in refereed international journals and refereed international conferences proceedings, including over 30 articles in IEEE journal in the last five years. His research interests include distributed systems, network security, bioinformatics, and e-Learning.

Dr. Wanlei was the General Chair/Program Committee Chair/Co-Chair of a number of international conferences, including ICA3PP, ICWL, PRDC, NSS, ICPAD, ICEUC, and HPCC.