Spatiotemporal Location Privacy Preservation in 5G-Enabled Sparse Mobile Crowdsensing

MingChu Li, Qifan Yang, Xiao Zheng, and Liqaa Nawaf

Abstract With the increasing popularity of 5G communications, smart cities have become one of the inevitable trends in the development of modern cities, and smart city services are the foundation of 5G smart cities. Sparse mobile crowdsensing (SparseMCS), as a new and informative urban service model, has attracted the attention of many researchers. Generally, the data required for a sensing task often has a high spatial and temporal correlation, which means that the data uploaded by users need to carry their location information, which may cause serious location privacy issues. The existing location privacy protection mechanism usually only pays attention to the location information of the user's travel and ignores that people's daily travel often has a fixed pattern. The attacker can use long-term observation and prior knowledge to infer the victim's travel mode and analyze its location information. To achieve efficient, robust, and private data sensing, we built a SparseMCS framework with the following three elements: (1) We train the data adjustment model offline on the server-side and solve the position mapping matrix; (2) Design a noise-sensitive data reasoning algorithm improves the accuracy of data; (3) Combining differences and spatiotemporal location privacy to protect the user's location information and travel mode. Experiments based on real datasets prove that our 5G-supported sparse mobile crowdsensing framework provides more comprehensive and effective location privacy protection.

Keywords 5G · Mobile crowdsensing · Location privacy · Differential privacy · Spatiotemporal phenomena

M. Li (✉) · Q. Yang
School of Software, Dalian University of Technology, Dalian, China
e-mail: mingchul@dlut.edu.cn

Q. Yang
e-mail: dekusmash_yqf@mail.dlut.edu.cn

X. Zheng
School of Computer Science and Technology, Shandong University of Technology, ZiBo, China
e-mail: xiao_zheng0910@163.com

L. Nawaf
Cardiff Metropolitan University, Cardiff, UK
e-mail: LLLNawaf@cardiffmet.ac.uk

A. K. Bashir et al. (eds.), *Proceedings of International Conference on Computing and Communication Networks*, Lecture Notes in Networks and Systems 394,
https://doi.org/10.1007/978-981-19-0604-6_24

1 Introduction

5G is the current mainstream new-generation mobile communication technology and an essential part of the next-generation information infrastructure [1]. The high-quality information services of 5G provide a good communication foundation for the construction of smart cities and industrial Internet of Things [2–6]. Mobile crowd-sensing systems can provide city services for the smart city systems, traffic information, weather information, and other services system. Therefore, mobile crowd-sensing (MCS) has developed rapidly in recent years and has become a significant computing paradigm in smart city data sensing scenarios. MCS plays a crucial role in collecting ambient temperature, traffic flow [7], noise [8], and air quality [9] in inter-city areas. In mainstream MCS, the publisher will launch a data sensing task for a specific target area. Service providers screen and recruit mobile users according to task requirements and perform tasks in the target area. However, large-scale data collection tasks such as urban tasks require many users to cover all target points. Therefore, urban tasks often require much budget, and target points are often missed due to uneven population distribution, and data redundancy may also occur in densely populated areas.

One solution is *Sparse Mobile Crowdsensing* (SparseMCS), which combines historical records and sensing data in nearby areas to infer task demand data in unperceived areas [10]. In SparseMCS, users need to report their location and time when uploading data, bringing considerable risks to user privacy [11]. Therefore, designing an effective privacy protection mechanism for the system can attract and retain more participants. In order to enable the MCS server to distinguish the data uploaded by each user, the privacy protection mechanism designed according to anonymity usually needs to retain the mapping information between the user's real identity and the anonymous information. If the server is attacked, users will face personal severe privacy risks. In contrast, according to the obfuscated design mechanism, it can usually be configured in a lightweight manner on a mobile device, thereby avoiding the hosting of accurate information. Therefore, we design a location privacy protection mechanism based on confusion.

Researchers have conducted extensive research on location privacy in location-based services. These two mechanisms are usually considered to protect user location privacy [12]: (a) The user protects privacy by making location tracking and personal identity impossible to associate by anonymous means; (b) Remapping the location to change the location information released by the user.

Cloaking is a prevalent obfuscation technology. The user can hide the actual location in multiple fine-grained stealth areas instead of one or several specific areas or units. However, when the adversary has some knowledge of the target user, the effect of cloaking may be significantly reduced [13]. For example, when the adversary learns that the target user is a doctor and that the user's cloaking area covers a hospital or other medical facilities, it is easy to locate the target.

In response to this problem, we use *differential privacy* [14] to ensure that the probability of accurate location mapping to different locations is approximate.

In location-based service (LBS), we usually use the distance between the actual location and the confusion location to measure data loss. Unlike it, the ultimate goal of SparseMCS is to collect target data, so the data loss in the system is determined by the difference between the actual location and the confusion location of the target data. According to this feature, we can think that in SparseMCS, as long as the target data difference between the actual location and the confusing location is slight, the user can theoretically map the actual location to a very far place. Therefore, we should redesign the location privacy protection mechanism according to the characteristics of SparseMCS.

Even if differential privacy is used, the user cannot control the range of the adversary's estimation of his location, which is an inference error [15]. Therefore, we added a *distortion privacy* [15] mechanism to control inference errors. Distortion privacy controls the adversary's estimated range by controlling the expected distance between the adversary's inferred location and the actual location. Applying distortion privacy requires the presumption of prior knowledge possessed by the adversary. Distortion privacy is to limit the inference attack to the preset inference error. The adversary cannot achieve a better inference error within the preset prior knowledge range than the optimal error. However, we cannot know the adversary's prior knowledge, so distortion privacy is not a powerful privacy protection mechanism. It needs to be used in conjunction with other privacy protection methods to provide more comprehensive protection.

However, the above LPPMs only consider the user's exact location information and do not realize that the location change of mobile users is a complex combination of time and space [16]. For example, "Alice went to a certain supermarket last week" (this behavior may occur more than once) and "Bob travels between A address and B address" (this behavior may occur every working day). In this article, we call it spatiotemporal location. We do not know whether the differential privacy mechanism can simultaneously guarantee a certain level of privacy in spatiotemporal locations. Therefore, we have introduced the privacy goal of the spatiotemporal location to ensure that users' daily travel patterns can be protected.

The main contributions of our work are:

- In order to provide more comprehensive location privacy protection, we propose a privacy protection framework that includes three privacy mechanisms. (a) Differential privacy guarantees the geographic indistinguishability; (b) Distortion privacy limits the adversary's optimal estimation error on prior knowledge; (c) Spatiotemporal location privacy guarantees the privacy of user behavior patterns.
- In order to improve the reliability and efficiency of the system, we designed a noise-aware reasoning algorithm to improve the data accuracy of the unperceived area.
- We validated our framework using real-world temperature datasets. The results show that our method, while providing a higher level of location privacy protection, limits the error within the range of 10^{-2}.

2 Related Work

With the rollout of 5G networks, the 5G environment integrates numerous location-based services, and mobile group awareness is one of them. Mobile crowdsensing is a data collection service that uses mobile devices to collect environmental data in the urban environment (for example, noise, air quality, temperature information, traffic flow) by hiring users distributed in different locations in the city. However, due to the large sensing area or limited budget, there may not be enough users to complete the sensing task. Wang et al. [10] proposed sparse mobile crowdsensing to solve this problem. Both MCS and SparseMCS can be regarded as a kind of LBS. Recruited users often need to expose their location to the task organizer, which involves serious location privacy issues.

Currently, most location privacy protection mechanisms mainly use two technologies: anonymity and obfuscation [17]. However, these two technologies will significantly reduce the strength of privacy protection when facing adversaries with prior knowledge [13]. In response to this problem, Andrés et al. [13] introduced the concept of differential privacy into location privacy protection to prevent attacks from adversaries with prior knowledge. According to the survey results of Pournajaf et al. [18], the current location privacy protection technology in MCS is mainly obfuscation technology. Many researchers combine MCS with edge computing. Putra et al. [19], Li et al. [20], and others have studied the location privacy protection mechanism in this environment. The DU-Min-$\epsilon\delta$ [21] proposed by Wang et al. realizes location privacy protection in the SparseMCS environment. Compared with this algorithm, our work considers the time dimension of location information and realizes the protection of user travel patterns.

3 Sparse Mobile Crowdsensing Concepts

3.1 Sparse Mobile Crowdsensing

3.1.1 Computation Paradigm

As shown in Fig. 1 (Basic), when monitoring temperature changes in a target city is started, the city will be divided into multiple fine-grained target areas. The user will collect the temperature data of the current area and upload the collected temperature data, identity information, and location information to the server. The server will use real-time sensor data and historical data to infer temperature information in areas that the user has not reached.

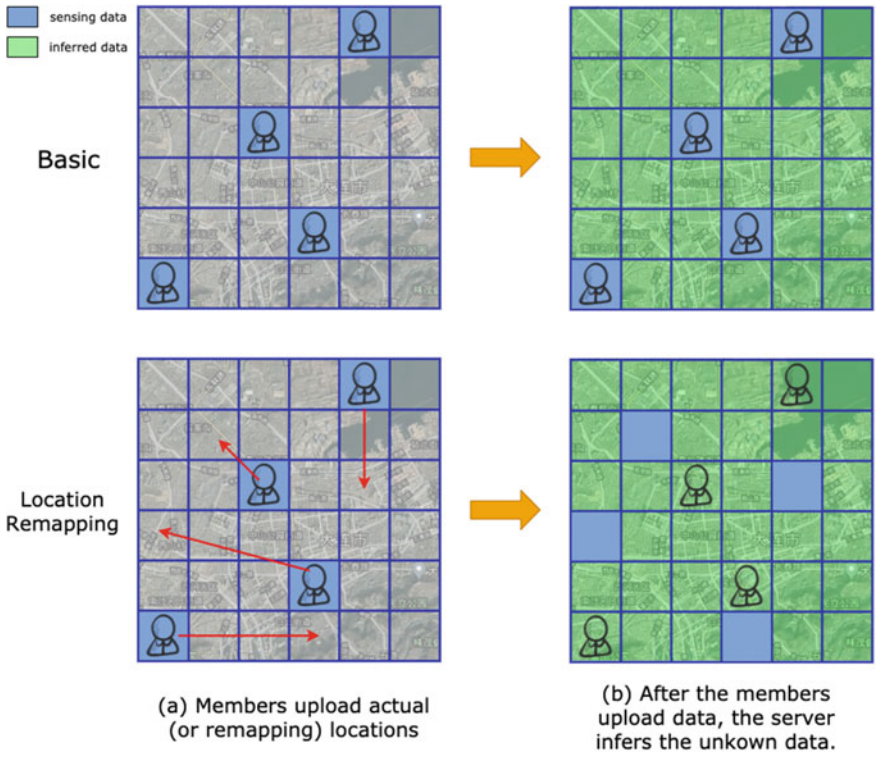


Fig. 1 Basic data collection is in sparse mobile crowdsensing, and data collection with location privacy protection added

3.1.2 Data Collection

The user will start the data collection task at the current location. In order to be able to verify identity and verify data, the server will require the user to upload identity information. At the same time, the target data collected by the user should have location information to achieve complete semantic functions, so location information should also be uploaded. Therefore, the simultaneous exposure of the user's identity information and location information to the task organizer will cause serious privacy risks. The SparseMCS system needs to enable strict LPPM to reduce the user's risk. However, LPPM will adjust the location information of the target data, which destroys the semantic function of the target data to a certain extent. The destruction of semantic functions will reduce the accuracy of target data.

3.1.3 Data Inference

In our work, we use compressed sensing as our data inference algorithm [22]. Candès and Plan [23] have proved that recovering an unknown low-rank matrix can uniformly sample a small number (less than the size of the matrix) with noisy entries. The recovery error is proportional to the noise level. In other words, there are two inherent assumptions in the use of compressed sensing algorithms to achieve data inference:

- Uniform distribution: The compressed sensing algorithm requires that sampled data are evenly distributed in the sampling space. That is to say, in SparseMCS, all the sensing locations in the target area should be evenly distributed. If not, for example, if no user exists in a specific area during all the sensing periods, it is impossible to infer the missing data in that area.
- Weak noise environment: When the sampling items do not carry noise and meet uniform sampling, the missing data in the matrix can be accurately inferred. When the sampling items carry noise, the total inference error is proportional to the noise level. That is to say, the smaller the noise carried in the sampling items, the higher the accuracy of the inference results.

3.2 Location Privacy-Preserving Framework for SparseMCS

The sensing data uploaded by the user in SparseMCS should include the target data and the actual location. Figure 1 (Location Remapping) shows that using obfuscation technology to add noise to the location information can reduce the user's privacy risk. However, this method will bring about data quality loss because the actual location and the target data of the confusing location may be different. In response to this problem, we designed a location privacy-sensitive SparseMCS framework composed of two parts: location remapping and data adjustment.

Figure 2 shows the location privacy protection framework of SparseMCS we designed, which consists of two parts: the server-side and the mobile user side. Before the task starts, the server will realize the data adjustment function and generate the location mapping probability matrix in the offline state according to the historical data. The data adjustment function is based on learning the relationship between the historical data in any two regions. This function will reduce the quality loss of the target data caused by the noise caused by the location remapping. By adjusting the probability matrix item $[i, j]$ (the probability that location i is mapped to location j), we can ensure that the adversary cannot accurately infer the user's actual location even if he gets the matrix.

Before performing the task, the user saves the data adjustment function and the mapping matrix on the mobile device. The task execution process is as follows: First, the user adjusts to the confusion location according to the current cycle and actual location according to the location mapping probability matrix (step M1). Subsequently, according to the actual location and the confused location, the original

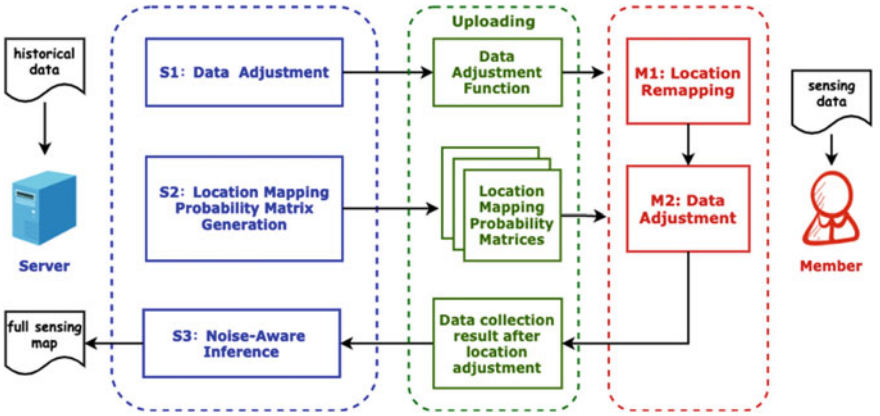


Fig. 2 Location privacy-preserving framework for SparseMCS

data is adjusted to the adjusted data with noise using the data adjustment function (step M2). The user uploads the adjustment data and the confused location to the server, and then the server combines the historical data to infer a complete sensing map (step S3).

4 Differential and Distortion and Spatiotemporal Location Privacy

This section introduces the privacy protection concept we applied in SparseMCS. Our privacy protection mechanism focuses on Bayesian attacks. Distortion privacy cannot resist Bayesian attacks, but it can effectively limit the optimal attack model based on Bayesian inference to minimize errors. The major notations are summarized in Table 1.

4.1 Differential Location Privacy

The purpose of introducing differential privacy is to bind the improvement of the posterior knowledge acquired by the adversary to the prior knowledge [14]. Differential privacy will make the probability of mapping from the real location r to any confusion location r^* similar.

Table 1 Notations

$\mathcal{R}$	Sensing task target area, $\mathcal{R} = \{r_1, r_2, \ldots, r_n\}$
P	Location mapping probability matrix
r	Real location $r \in \mathcal{R}$
r', r^*	Confusion locations $r', r^* \in \mathcal{R}$
$\tilde{r}$	Adversary inferred location $\tilde{r} \in \mathcal{R}$
$P(r^*\|r)$	The probability of location r mapped to location r^*
$\tilde{\beta}$	Adversary's inference attack
$\rho(\tilde{r}, r)$	The distance between $\tilde{r}$ and r^*
$\eta_u(r)$	The location distribution of target user
T	The time period for the user to release locations
$\mathcal{O}$	The user's observable location sequence
$\mathcal{S}$	User-defined sensitive areas

Definition 1 (ϵ-*Differential Privacy*) Assuming that the $\mathcal{R}$ is divided into multiple fine-grained areas r, then the P satisfies ϵ-Differential Privacy iff:

$$P(r^*|r) \leq e^{\epsilon} \cdot P(r^*|r'), \qquad \forall r, r', r^* \in \mathcal{R} \tag{1}$$

where ϵ represents the privacy budget.

4.2 Distortion Location Privacy

Although differential privacy limits the adversary's information gain, users still cannot determine how close the adversary's estimated location is to its actual location, that is, how small the adversary's inference error is. In order to limit the inference error, we adopt distortion privacy [15]. This method can ensure that the adversary's optimal attack inference error will be greater than a particular value for a given user's public location distribution information.

4.2.1 Attack Inference Error

The attack inference error can be obtained by the following equation:

$$\sum_{r^* \in \mathcal{R}} P(r^*|r) \sum_{\tilde{r} \in \mathcal{R}} \tilde{\beta}(\tilde{r}|r^*) \cdot \rho(\tilde{r}, r) \tag{2}$$

We assume that the location distribution η_u is partially disclosed (for example, social network check-ins [24]). Furthermore, the adversary obtains the distribution, and he can minimize the expected reasoning error on η_u to achieve the optimal attack.

$$\arg\min_{\tilde{\beta}} \sum_{r^* \in \mathcal{R}} \eta_u(r) \sum_{r^* \in \mathcal{R}} P(r^*|r) \sum_{\tilde{r} \in \mathcal{R}} \tilde{\beta}(\tilde{r}|r^*) \cdot \rho(\tilde{r}, r) \tag{3}$$

4.2.2 Definition of Distortion Location Privacy

Definition 2 (*δ-Distortion Privacy*) The location mapping probability matrix P satisfies δ-Distortion Privacy iff:

$$\sum_{r^* \in \mathcal{R}} \eta_u(r) \sum_{r^* \in \mathcal{R}} P(r^*|r) \sum_{\tilde{r} \in \mathcal{R}} \tilde{\beta}(\tilde{r}|r^*) \cdot \rho(\tilde{r}, r) \geq \delta \tag{4}$$

where δ is the lower bound of privacy disclosure acceptable to users.

The disclosed location distribution η_u does not always summarize the adversary's prior knowledge. The distortion privacy only makes a mild assumption and cannot contain some extreme situations.

4.3 Spatiotemporal Location Privacy

Figure 3 vividly shows the relationship and difference between the spatial dimension, time dimension, and space-time dimension of location privacy. Differential and distortion privacy only realizes the case of privacy protection in the spatial dimension, and it is not clear whether it can provide location privacy protection in the spatial and temporal dimensions. Therefore, we use spatiotemporal location privacy [25] to extend location privacy protection to the spatial and temporal dimensions.

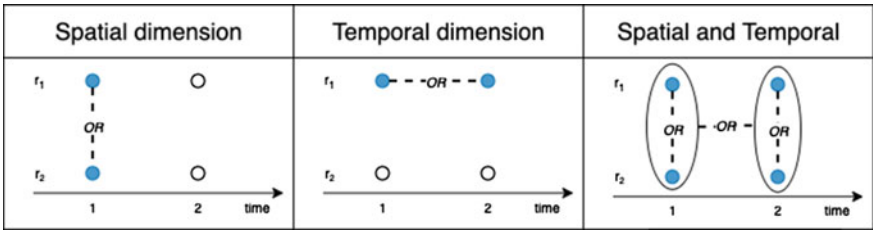


Fig. 3 Dimensional analysis of location privacy. Spatial dimension: privacy refers to a sensitive area including location r_1 and r_2; temporal dimension: privacy refers to accessing location r_1 at time point 1 or 2; spatial and temporal dimension: the user's sensitive area consists of locations r_1 and r_2 at time point 1 and 2

Definition 3 (ϵ-*Spatiotemporal Location Privacy*) Suppose that in period $T = \{1, 2, \ldots, t\}$ and area $\mathcal{R}$. The user sets the sensitive area $\mathcal{S}$, and generates the observation sequence $\mathcal{O} = \{r^1, r^2, \ldots, r^t\}, \forall r^i \in \mathcal{R}$. $\mathcal{O}$ satisfies the ϵ-Spatiotemporal Location Privacy iif:

$$P(\mathcal{O}|\mathcal{S}) \leq e^{\epsilon} \cdot P(\mathcal{O}|\neg\mathcal{S}) \tag{5}$$

where $\neg\mathcal{S}$ is the complementary set of $\mathcal{S}$, and $U = \mathcal{S} \cap \neg\mathcal{S}$ represents all possible sensitive areas in the target area.

4.4 Combined Location Privacy-Preserving Mechanism

In this section, we will briefly describe the advantages of combining the above three privacy concepts. For differential privacy, it is difficult for the adversary to predict the correct location of the user accurately. For distortion location privacy, the adversary's prediction should keep a certain distance from the correct location even if the guess is wrong. For spatiotemporal location privacy, it is difficult for adversaries to analyze the user's travel habits through long-term observation. Therefore, we combine these three privacy concepts to provide users with more comprehensive location privacy protection.

5 Location Privacy Protection Mechanism with Minimal Data Loss

5.1 Data Quality Requirements for Location Mapping

Recall that the prerequisites for data inference introduced in Sect. 3.1.3 include the average distribution of sampled data and a weak noise environment [23]. However, the introduction of a location privacy mechanism will destroy these two premises:

(a) *Uniform distribution of confusion locations*: In real life, users may be evenly distributed within the city, but the location distribution may be very uneven after location mapping. For example, suppose that no user's location may map to location i. Then, the value of the ith row in the sensing matrix will be lost, and the ith row data will not be restored during the inference process.
(b) *Weak noise environment*: After the location remapping process, the target data submitted by the user corresponds to the original location rather than the confusing location. Although the resulting error can be reduced through data adjustment, the uploaded target data is still inaccurate. Therefore, we need to generate a matrix P that can minimize the loss of data quality.

5.2 Optimal Location Mapping Matrix Generation

In order to meet the challenge mentioned in Sect. 5.1, we designed an optimization problem to generate the position mapping probability matrix.

5.2.1 Objective: Minimize Data Loss

We denote the data loss produced by this process as a matrix L. The entry $L[r, r^*] \in L$ records the residual standard deviation between the original data and the adjusted data. Intuitively, the less data loss caused by the location mapping process, the better. We hope to find a mapping matrix P that can minimize the overall expectation of the data loss caused by the process, denoted as $\bar{L}$. Our optimization goal is to minimize $\bar{L}$.

$$\bar{L} = \sum_{r \in \mathcal{R}} \eta_u(r) \cdot \sum_{r^* \in \mathcal{R}} L[r, r^*] \cdot P[r, r^*] \tag{6}$$

5.2.2 Location Mapping Probability Matrix Generation

According to Definition 5, spatiotemporal location privacy should protect a sensitive area in the time period, and differential and distortion location privacy is protected for the location at the time point. Therefore, we generate a matrix P^t that satisfies differential and distortion location privacy at each time point in the sensing cycle T. $\mathcal{P} = \{P^1, P^2, \ldots, P^t\}, \forall t \in T$ satisfies the spatiotemporal location privacy in the period. In order to minimize the expectation of data loss and ensure the location mapping probability matrix $\mathcal{P}$ of differential privacy, distortion privacy, and spatiotemporal location privacy.

(a) *Constraint 1*: ϵ-*Differential Privacy*: The first constraint is ϵ-Differential Privacy, which is implemented by Eq. 8.
(b) *Constraint 2*: δ-*Distortion Privacy*: The second constraint is δ-Distortion Privacy, which is implemented by Eqs. 9 to 10. Because Eq. 4 in Definition 2 contains an optimization problem (Eq. 3), we cannot directly use it as a constraint. Therefore, according to Shokri's work [15], we convert Eq. 4 into Eqs. 9 and 10.
(c) *Constraint*: ϵ-*Spatiotemporal Location Privacy*: The third constraint is ϵ-Spatiotemporal Location Privacy, which is implemented by Eq. 5. At each time t in the period T, a P^t that meets the requirements is generated. P^t meets the requirements of spatiotemporal location privacy, which means that the probability of the user appearing in the sensitive area and not appearing in the sensitive area within this period is similar. According to Definition 5, we can abstract the user's presence in the sensitive area $\mathcal{S}$ in T into a boolean expression. Assuming that the period $T = \{1, 2\}$, the sensitive area $\mathcal{S} = \{r_1, r_2\}, \forall r_i \in \mathcal{R}$, the user's appearance in the sensitive area during this period can be abstracted as $[(u^1 = r_1) \vee (u^1 = r_2)] \wedge [(u^2 = r_1) \vee (u^2 = r_2)]$. In the expression, $u^t = r_i$

means that the user is at the location of r_i at time t. Abstracting the period T and the sensitive area $\mathcal{S}$ and converting the Boolean expression into a probability expression, Eq. 11 can be obtained.

The optimization problem established based on the above content is as follows:

$$\arg\min_{P^t} \quad \bar{L}(P^t) = \sum_{r\in\mathcal{R}} \eta_u(r) \sum_{\substack{r^*\in\mathcal{R}\\ t\in T}} L\left[r, r^*\right] \cdot P^t\left(r^* \mid r\right) \tag{7}$$

$$\text{s.t.} \quad P^t(r^*|r) \leq e^{\epsilon} \cdot P^t(r^*|r'), \forall r, r', r^* \in \mathcal{R}, \forall t \in T \tag{8}$$

$$\sum_{r\in R} \eta_u(r) P^t\left(r^*|r\right) d(\tilde{r}, r) \geqslant x^t\left(r^*\right), \forall \tilde{r}, r^* \in \mathcal{R}, \forall t \in T \tag{9}$$

$$\sum_{r^*\in\mathcal{R}} x^t(r^*) \geq \delta, \forall t \in T \tag{10}$$

$$\prod_{t\in T} \eta_u(r) \sum_{r^*\in R} P^t\left(r^*|r\right) \leqslant e^{\epsilon} \prod_{t\in T} \eta_{\mu}\left(r'\right) \sum_{r^*\in R} P^t\left(r^*|r'\right), \quad \forall r \in \mathcal{S}^*, r' \in \mathcal{S}_i, \forall \mathcal{S}_i \in \neg\mathcal{S}^* \tag{11}$$

$$\sum_{r\in\mathcal{R}} \eta_u(r) \cdot P^t(r^*|r) = 1/\mathcal{R}, \forall r^* \in \mathcal{R}, \forall t \in T \tag{12}$$

$$\sum_{r^*\in\mathcal{R}} P^t(r^*|r) \ = \ 1, \forall r \in \mathcal{R}, \forall t \in T \tag{13}$$

$$P^t(r^*|r) \geq 0, \forall r, r^* \in \mathcal{R}, \forall t \in T \tag{14}$$

5.3 Noise-Aware Inference Algorithm

Compressed sensing algorithms require a weak noise environment, but in order to ensure location privacy, we have violated this condition. In order to solve this problem, we designed a noise-aware mechanism to sample data with a small amount of noise with a higher weight. The following equation obtains the data loss expectation corresponding to each mapping location:

$$\bar{L}_{\cdot,r^*} = \sum_{r\in\mathcal{R}} \eta_u(r) \cdot P(r^*|r) \cdot L[r, r^*] \tag{15}$$

According to the expected data loss of each mapping location, we obtain the sampling weight corresponding to each mapping location through the following equation:

$$\omega_{r^*} = \omega_0 + (1 - \omega_0) \cdot \frac{\bar{L}_{\max} - \bar{L}_{\cdot,r^*}}{\bar{L}_{\max} - \bar{L}_{\min}} \tag{16}$$

where, $\bar{L}_{\max}$ and $\bar{L}_{\min}$, respectively, represent the largest and smallest data loss expectations in all mapping locations. We denote the sampling weight of the mapping location corresponding to the expected maximum data loss as ω_0. According to the experiment in Sect. 6, we recommend setting the weight to 0.75.

6 Evaluation

6.1 Configuration Environment

6.1.1 Baseline

We use three baselines that implement differential location privacy protection. Under the same level of differential privacy, we will show that our method will additionally protect user behavior patterns with similar data quality loss.

(a) *Self*: *Self* [26] algorithm provides a higher probability for location self-mapping. Formally, the location mapping matrix generated by this algorithm satisfies differential privacy:

$$P_{i,j} = \begin{cases} \alpha\, e^{\epsilon}, & \text{if } i = j, \\ \alpha, & \text{o.w.} \end{cases}$$

(b) *Laplace*: The Laplacian mechanism [13] completes privacy protection by adding Laplacian noise to the actual data. This method is more inclined to map locations to neighbor locations.

(c) *DU-Min-$\epsilon\delta$*: This method [21] constructs a linear optimization problem to achieve local location differential privacy in a sparse crowdsensing environment.

6.1.2 Evaluation Environment

We used SensorScope [27] open-source actual temperature sensing data as our experimental dataset. They deployed temperature sensors on the EPFL campus, covering an area of 300 m × 500 m. We divide it into 100 sub-areas with 30 m × 50 m, of which 57 contain temperature sensors (that is, contain real data). The data collection lasted for a week, the sensing period was 30 min, the first day's data was used as the training data adjustment function, and the location mapping matrix was solved, and the rest were used as tests (Table 2).

Table 2 Evaluation parameters

	Default	Description
k	4	Number of sensing data collected in each cycle
ϵ	$\ln 4$	Differential privacy budget
c	3	Number of cycles users perform sensing tasks
ω	0.75	Basic sampling weight

6.1.3 Experimental Parameters

We assign different privacy budgets, ϵ the number of sensing data collected by participants in each sensing cycle, k, and the number of cycles that participants perform sensing tasks, c, as experimental independent variables. ϵ is usually customized by the user. For convenience, we set it from $\ln 2$ to $\ln 8$. The publisher generally determines k based on the budget held and the quality of the data required. The service provider will set c based on the user's travel mode and expected data quality.

6.1.4 Data Quality Metric

We use the *Mean Absolute Error (MAE)* to calculate the data loss of the inferred data compared to the real data. Every time we modify the experimental parameters, we perform five repeated experiments and take the average value. The data loss caused by the location privacy protection mechanism (LPPM) is defined as follows:

$$L_{\mathrm{MAE}}(\mathrm{LPPM}) = \mathrm{MAE}(\mathrm{LPPM}) - \mathrm{MAE}(\text{No-Privacy})$$

6.2 Experimental Performance

Our experimental results show that our work can provide more effective location privacy protection at a lower cost of data loss. Compared with the baseline algorithm, our work provides more comprehensive location privacy protection, and the additional data loss generated on this basis is also controlled within the range of 10^{-2}. When the number of task cycles c is small, the data quality loss caused by our work can be further controlled within 10^{-3}.

We measured how the target data quality loss changes with the privacy budget ϵ. From Fig. 4, we can see that as the privacy budget increases (the intensity of privacy protection decreases), data quality loss will decrease. In general, our work is better than the Laplace and self versions under the same conditions. When the privacy budget is small, the error level of our work is similar to that of DU-Min-$\epsilon\delta$.

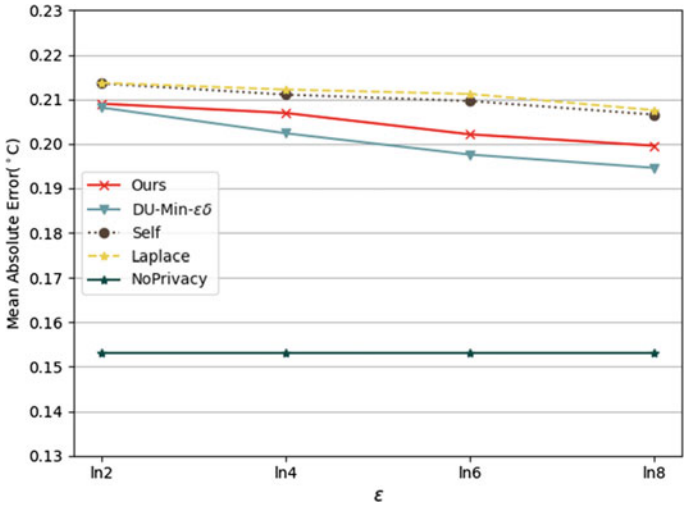
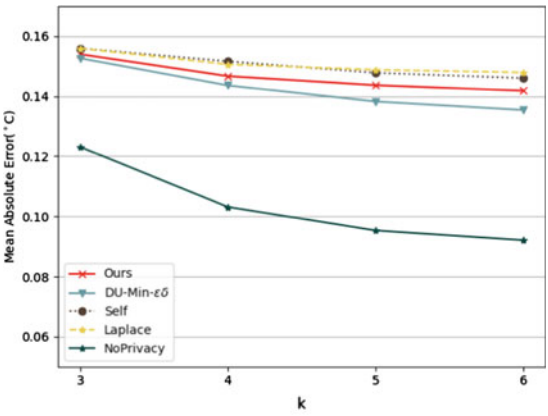


Fig. 4 MAE changes with ϵ

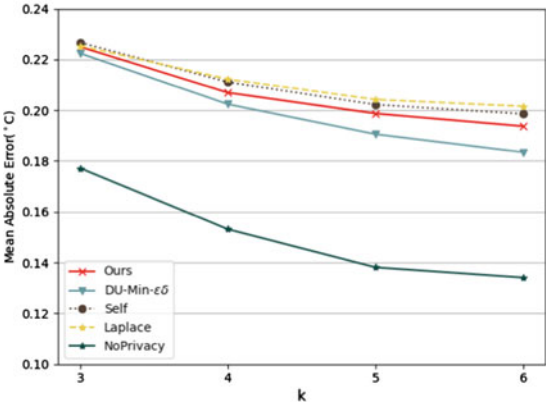
Figure 5 shows how the target data quality varies with the number of sensor data k collected in each cycle. From Fig. 5, we can see that the target data quality loss will decrease with the increase of k, even if we change the number of cycles c for the user to perform sensing tasks. In addition, due to the data noise caused by LPPM, the data quality loss decreases more and more slowly and cannot reach the level of no privacy. Moreover, our algorithm is superior to $Laplace$ and $Self$ in terms of data loss. Compared with DU-Min-$\epsilon\delta$, the error is also controlled within 10^{-2}. More importantly, we provide more comprehensive location privacy protection.

Figure 6 shows the loss of data quality when the number of cycles c of the user performing the sensing task is 2 and 3. Since the number of constraints in the optimization problem for generating the optimal location mapping probability matrix is the factorial of c, the complexity of the problem will become higher when c is larger, so we only calculated the cases where c is 2 and 3. However, to ensure the quality of the data in the SparseMCS environment, we usually do not need the same user to perform multiple sensing cycles. In future work, we will further reduce the complexity of the optimization problem to adapt to more scenarios.

In order to verify the impact of the basic sampling weight on the data quality loss, we change the weight and calculate the data quality loss under different privacy budgets when k and c are the default values. Figure 7 shows the results of the experiment. Under different privacy budgets, we find that the data error is the smallest when ω_0 is 0.75.



(a) $c = 2$



(b) $c = 3$

Fig. 5 MAE changes with the amount of sensing data collected by participants in each cycle

7 Conclusions

This paper presents a spatiotemporal and differential location privacy protection mechanism for 5G-enabled sparse mobile crowdsensing. It considers the level of location privacy protection required by users, the protection of travel modes, the ability to resist attacks from attackers with prior solid knowledge, and the loss of data quality due to location mapping. In particular, users can use this framework to develop personalized location privacy protection based on their travel mode. Experiments based on real data verify the effectiveness of the framework.

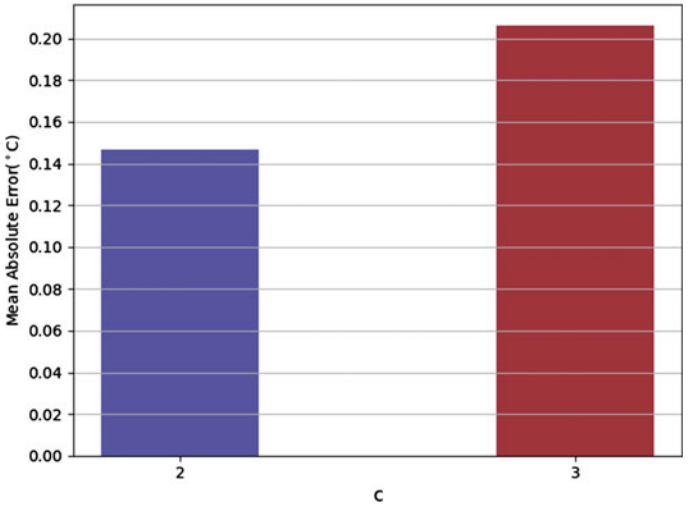


Fig. 6 MAE changes with c

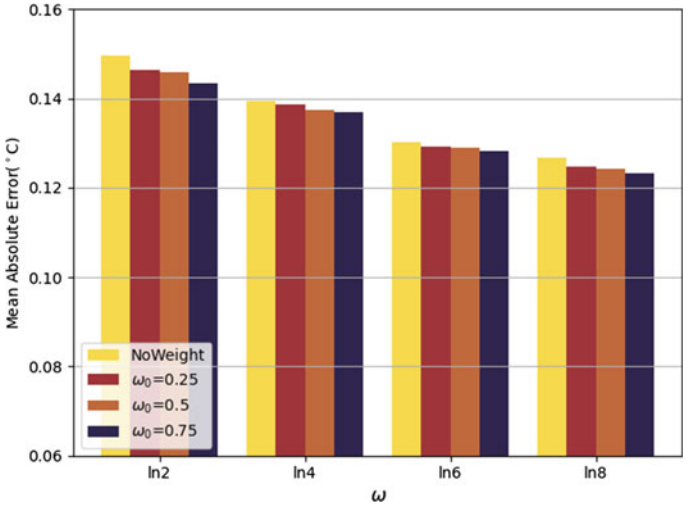


Fig. 7 MAE changes with ω

Acknowledgements This work was support from the National Science Foundation of China (Grant No. 61877007), Fundamental Research Funds for the Central Universities (Grant No. DUT20GJ205), and the Shandong National Science Foundation of China (Grant No. ZR202103040468).

References

1. L. Tan, H. Xiao, K. Yu, M. Aloqaily, Y. Jararweh, A blockchain-empowered crowdsourcing system for 5G-enabled smart cities. Comput. Stand. Interfaces **76**, 103517 (2021). [Online]. https://doi.org/10.1016/j.csi.2021.103517
2. S.B. Shah, C. Zhe, F. Yin, I.U. Khan, S. Begum, M. Faheem, F.A. Khan, 3D weighted centroid algorithm & RSSI ranging model strategy for node localization in WSN based on smart devices. Sustain. Cities Soc. **39**, 298–308 (2018). [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2210670717312982
3. S.B.H. Shah, L. Wang, M.E. Haque, M.J. Islam, A. Carie, N. Kumar, Lifetime improvements of smart sensors maintenance protocol in prospect of IoT-based Rampal power plant, in *2020 16th International Conference on Mobility, Sensing and Networking (MSN)* (2020), pp. 260–267
4. S.B.H. Shah, Z. Chen, S.H. Ahmed, F. Yin, M. Faheem, S. Begum, Depth based routing protocol using smart clustered sensor nodes in underwater WSN, in *Proceedings of the 2nd International Conference on Future Networks and Distributed Systems*, Amman, Jordan, 26–27 June 2018, ed. by A. Abuarqoub, B. Adebisi, M. Hammoudeh, S. Murad, M. Arioua (ACM, 2018), pp. 53:1–53:7. [Online]. https://doi.org/10.1145/3231053.3231119
5. M. Faheem, R.A. Butt, B. Raza, M.W. Ashraf, M.A. Ngadi, V.C. Gungor, A multi-channel distributed routing scheme for smart grid real-time critical event monitoring applications in the perspective of industry 4.0. Int. J. Ad Hoc Ubiquitous Comput. **32**(4), 236–256 (2019). [Online]. https://doi.org/10.1504/IJAHUC.2019.103264
6. M. Faheem, R.A. Butt, R. Ali, B. Raza, M.A. Ngadi, V.C. Gungor, CBI4.0: a cross-layer approach for big data gathering for active monitoring and maintenance in the manufacturing industry 4.0. J. Ind. Inf. Integr. **24**, 100236 (2021). [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2452414X21000364
7. Y. Zhu, Z. Li, H. Zhu, M. Li, Q. Zhang, A compressive sensing approach to urban traffic estimation with probe vehicles. IEEE Trans. Mob. Comput. **12**(11), 2289–2302 (2013)
8. R.K. Rana, C.T. Chou, S.S. Kanhere, N. Bulusu, W. Hu, Ear-phone: an end-to-end participatory urban noise mapping system, in *Proceedings of the 9th International Conference on Information Processing in Sensor Networks, IPSN 2010*, Stockholm, Sweden, 12–16 Apr 2010, ed. by T.F. Abdelzaher, T. Voigt, A. Wolisz (ACM, 2010), pp. 105–116. [Online]. https://doi.org/10.1145/1791212.1791226
9. D. Hasenfratz, O. Saukh, S. Sturzenegger, L. Thiele, Participatory air pollution monitoring using smartphones. Mob. Sens. (2012)
10. L. Wang, D. Zhang, Y. Wang, C. Chen, X. Han, A. M'hamed, Sparse mobile crowdsensing: challenges and opportunities. IEEE Commun. Mag. **54**(7), 161–167 (2016)
11. J.E. Dobson, P.F. Fisher, Geoslavery. IEEE Technol. Soc. Mag. **22**(1), 47–52 (2003). [Online]. https://doi.org/10.1109/MTAS.2003.1188276
12. J. Krumm, A survey of computational location privacy. Pers. Ubiquitous Comput. **13**(6), 391–399 (2009)
13. M.E. Andrés, N.E. Bordenabe, K. Chatzikokolakis, C. Palamidessi, Geo-indistinguishability: differential privacy for location-based systems, in *2013 ACM SIGSAC Conference on Computer and Communications Security, CCS'13*, Berlin, Germany, 4–8 Nov 2013, ed. by A. Sadeghi, V.D. Gligor, M. Yung (ACM, 2013), pp. 901–914
14. C. Dwork, Differential privacy, in *33rd International Colloquium on Automata, Languages and Programming, ICALP 2006*, Proceedings, Part II, Venice, Italy, 10–14 July 2006, ed. by M. Bugliesi, B. Preneel, V. Sassone, I. Wegener. Lecture Notes in Computer Science, vol. 4052 (Springer, 2006), pp. 1–12
15. R. Shokri, Privacy games: optimal user-centric data obfuscation, in Proc. Priv. Enh. Technol. **2015**(2), 299–315 (2015). [Online]. https://doi.org/10.1515/popets-2015-0024
16. Y. Cao, Y. Xiao, L. Xiong, L. Bai, PriSTE: from location privacy to spatiotemporal event privacy, in *35th IEEE International Conference on Data Engineering, ICDE 2019*, Macao, China, 8–11 Apr 2019 (IEEE, 2019), pp. 1606–1609

17. V. Primault, A. Boutet, S.B. Mokhtar, L. Brunie, The long road to computational location privacy: a survey. IEEE Commun. Surv. Tutor. **21**(3), 2772–2793 (2019)
18. L. Pournajaf, D.A. Garcia-Ulloa, L. Xiong, V.S. Sunderam, Participant privacy in mobile crowd sensing task management: a survey of methods and challenges. SIGMOD Rec. **44**(4), 23–34 (2015)
19. K.T. Putra, H. Chen, Prayitno, M.R. Ogiela, C. Chou, C. Weng, Z. Shae, Federated compressed learning edge computing framework with ensuring data privacy for PM2.5 prediction in smart city sensing applications. Sensors **21**(13), 4586 (2021). [Online]. https://doi.org/10.3390/s21134586
20. M. Li, Y. Li, L. Fang, ELPPS: an enhanced location privacy preserving scheme in mobile crowd-sensing network based on edge computing, in *19th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom 2020*, Guangzhou, China, 29 Dec 2020–1 Jan 2021, ed. by G. Wang, R.K.L. Ko, M.Z.A. Bhuiyan, Y. Pan (IEEE, 2020), pp. 475–482. [Online]. https://doi.org/10.1109/TrustCom50675.2020.00071
21. L. Wang, D. Zhang, D. Yang, B.Y. Lim, X. Han, X. Ma, Sparse mobile crowdsensing with differential and distortion location privacy. IEEE Trans. Inf. Forensics Secur. **15**, 2735–2749 (2020)
22. L.T. Nguyen, J. Kim, B. Shim, Low-rank matrix completion: a contemporary survey. IEEE Access **7**, 94215–94237 (2019)
23. E.J. Candès, Y. Plan, Matrix completion with noise. Proc. IEEE **98**(6), 925–936 (2010)
24. D. Yang, D. Zhang, Z. Yu, Z. Yu, Fine-grained preference-aware location search leveraging crowdsourced digital footprints from LBSNs, in *The 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing, UbiComp '13*, Zurich, Switzerland, 8–12 Sept 2013, ed. by F. Mattern, S. Santini, J.F. Canny, M. Langheinrich, J. Rekimoto (ACM, 2013), pp. 479–488. [Online]. https://doi.org/10.1145/2493432.2493464
25. Y. Cao, Y. Xiao, L. Xiong, L. Bai, M. Yoshikawa, PriSTE: protecting spatiotemporal event privacy in continuous location-based services. Proc. VLDB Endow. **12**(12), 1866–1869 (2019)
26. S. Agrawal, J.R. Haritsa, A framework for high-accuracy privacy-preserving mining, in *Proceedings of the 21st International Conference on Data Engineering, ICDE 2005*, Tokyo, Japan, 5–8 Apr 2005, ed. by K. Aberer, M.J. Franklin, S. Nishio (IEEE Computer Society, 2005), pp. 193–204
27. F. Ingelrest, G. Barrenetxea, G. Schaefer, M. Vetterli, O. Couach, M. Parlange, Sensorscope: application-specific sensor network for environmental monitoring. ACM Trans. Sens. Netw. **6**(2), 17:1–17:32 (2010)