IEICE TRANS. INF. & SYST., VOL.E106–D, NO.12 DECEMBER 2023

PAPER

Mechanisms to Address Different Privacy Requirements for Users and Locations

Ryota HIRAISHI[†a)], ***Nonmember***, **Masatoshi YOSHIKAWA**[†], ***Fellow***, **Yang CAO**[††], ***Member***, **Sumio FUJITA**[†††], ***Nonmember***, ***and*** **Hidehito GOMI**[†††], ***Member***

SUMMARY The significance of individuals' location information has been increasing recently, and the utilization of such data has become indispensable for businesses and society. The possible uses of location information include personalized services (maps, restaurant searches and weather forecast services) and business decisions (deciding where to open a store). However, considering that the data could be exploited, users should add random noise using their terminals before providing location data to collectors. In numerous instances, the level of privacy protection a user requires depends on their location. Therefore, in our framework, we assume that users can specify different privacy protection requirements for each location utilizing the adversarial error (AE), and the system computes a mechanism to satisfy these requirements. To guarantee some utility for data analysis, the maximum error in outputting the location should also be output. In most privacy frameworks, the mechanism for adding random noise is public; however, in this problem setting, the privacy protection requirements and the mechanism must be confidential because this information includes sensitive information. We propose two mechanisms to address privacy personalization. The first mechanism is the individual exponential mechanism, which uses the exponential mechanism in the differential privacy framework. However, in the individual exponential mechanism, the maximum error for each output can be used to narrow down candidates of the actual location by observing outputs from the same location multiple times. The second mechanism improves on this deficiency and is called the donut mechanism, which uniformly outputs a random location near the location where the distance from the user's actual location is at the user-specified AE distance. Considering the potential attacks against the idea of donut mechanism that utilize the maximum error, we extended the mechanism to counter these attacks. We compare these two mechanisms by experiments using maps constructed from artificial and real world data.

key words: *location privacy, personalization, entropy, error, privacy*

1. Introduction

Recently, the significance of personal location information has increased. Such data can be utilized for weather forecasting services and to locate nearby stores. Provided significant data are compiled and aggregated, it can help make business decisions such as opening new stores or expanding service areas. Furthermore, they can be utilized in applications that provide information on traffic congestion. However, considering the data could be exploited, users should add random noises using their terminals before providing location information to the collectors. When users output location information in such a framework, in many instances, the degree of privacy protection required by each user differs between locations. In this study, we investigate a method for privacy personalization that can manage different privacy requirements for each location.

Two closely corresponding studies have addressed the personalization of location privacy. PIVE and DPIVE [1], [2] guarantee that the adversarial error (AE) in each region is greater than or equal to the required error value of the user. Customizable Robust Geo-Indistinguishability (CORGI) [3] allows a user to specify the degree of privacy protection regarding output range and output granularity for each region. However, in [1], [2], users are not allowed to specify the privacy protection degree for each location, and [3] does not consider the AE or entropy at all. To the best of our knowledge, no studies are exploring methods that allow users to specify the privacy protection degree in the AE for each location. This study is the first work that explores such personalization of privacy.

The remainder of this paper is organized as follows: First, in Sect. 2, we describe the problem setting and framework of the proposed system. In Sect. 3, we introduce the studies on privacy personalization. Section 4 describes the privacy standards used in this study. Subsequently, in Sect. 5, we show two mechanisms proposed to achieve privacy personalization and attacks against the mechanism. Finally, in Sect. 6, we compare the proposed two mechanisms through experiments using an actual map; in Sect. 7, we discuss the limitations of the two mechanisms.

2. Preliminary

2.1 Problem Setting

In this section, we describe the problem settings used in this study.

Users must transmit their location information to service providers or data collectors to utilize location-based services. Before transmitting the location information, the user adds noise to the location using a mechanism. However, users often have different privacy preferences. For example, users are expected to have high privacy protection requirements for areas where their homes, workplaces, or

Manuscript received March 13, 2023.
Manuscript revised July 31, 2023.
Manuscript publicized September 25, 2023.

[†]The authors are with Dept. of Social Informatics, Kyoto University, Kyoto-shi, 606–8501 Japan.
[††]The author is with Graduate School / Faculty of Information Science and Technology, Hokkaido University, Sapporo-shi, 060–0808 Japan.
[†††]The authors are with Yahoo Japan Corporation, Tokyo, 102–8282 Japan.
a) E-mail: hiraishi.ryouta.73m@kyoto-u.jp

DOI: 10.1587/transinf.2023EDP7050

Table 1 Meanings of symbols

Symbol	Meaning
M	A mechanism for adding random noise to data
$\mathcal{X}$	The set of the entire region in a grid
x, x'	One specific region in the grid
$\hat{x}$	An adversary's expected region to the user's actual region
$\Pr(M(x) = x')$	The probability that M outputs x' when x is input
$\Pr(h(x') = \hat{x})$	The probability that $\hat{x}$ is a prediction of the user's actual region when an adversary observes x'
$\pi(x)$	The prior probability that users are in x
$d(x, x')$	Distance between centers of x and x'
$req_{err}(x)$	Requirement for AE specified by the user in x
r_x	Maximum error in output from x
C_x	Set of candidate regions of the user's actual region, refined by an adversary when the user is in x

facilities such as hospitals, are located. Furthermore, they have low requirements for locations where many people gather such as train stations or recreational facilities. Therefore, developing mechanisms that allow users to specify the degree of privacy protection for each location is necessary. We propose methods to achieve this requirement by adjusting the amount of noise for each location.

It should be noted that the system operates only on the user's terminal and does not transmit any data to the server when constructing the mechanism. This is because, users are expected to specify a high privacy protection level for locations where they do not want to be located by adversaries, and thus, their privacy preferences contain sensitive information. Similarly, the mechanism should be confidential because it is computed based on the user's privacy preferences. However, the mechanism is considered public information in the differential privacy framework. This aspect is the primary difference between our study and existing research.

2.2 Framework

In this section, we introduce our proposed system framework. The symbols used in this paper and their meanings are listed in Table 1.

In this study, we use a grid to represent a map. The location information used for service or data analysis is managed in units of grid cells (regions). The set of all regions in the grid is denoted by $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$. We denote the number of regions by $|\mathcal{X}|$. Let x and x' be the regions containing locations p and p', respectively, then the distance between p and p' is approximated by the Euclidean distance between the centers of the regions x and x', denoted $d(x, x')$.

Before transmitting the data, the user specifies the degree of privacy protection $req_{err}(x)$ for each region x ($\in \mathcal{X}$) using the expected distance between the user's actual location and an adversary's prediction of the user's location after observing the noise-added location (AE). The AE is a more intuitive measure than the privacy parameter ϵ of differential privacy [4]; thus, it is a reasonable value for users to specify. Because it is considered to be burden for the user to specify the degree of privacy protection for all regions, it is also effective to design the system so that the user only specifies AE in regions that require protection. In addition, default values are utilized for other regions. Subsequent to receiving $req_{err}(x)$ for each region x ($\in \mathcal{X}$), the system computes a mechanism M. The user stores M (or the parameters used to develop M) and exploits it, adding noise to the user's location. Data analysts perform various analyses, and adversaries attempt to identify the user's actual region based on the observed location.

3. Related Work

In this section, we present previous work related to this study.

3.1 k-Anonymity

k-anonymity is a widely known privacy protection technique. It was originally proposed as a way to guarantee privacy in relational databases. In k-anonymity, there should be at least $k-1$ identical data for each data. This ensures indistinguishability among at least k records. There has been extensive research on the application of k-anonymity to the protection of location information. Gruteser et al. [5] proposed a method to guarantee k-anonymity in space by dividing the region up to the limit where k data exist at the same time. Moreover, they proposed a method to guarantee k-anonymity in time by delaying the output until k data are collected.

3.2 Differential Privacy

In the field of privacy, differential privacy by Dwork [4] has attracted considerable attention. Differential privacy is a privacy criterion that makes it challenging to distinguish between two databases that differ only by a single record when observing query answers. Chatzikokolakis et al. [6] extended the definition of differential privacy to general data, called $d_{\mathcal{X}}$-privacy. If a mechanism M on $\mathcal{X}$ satisfies the following inequality for any $x, x' \in \mathcal{X}, Z \subseteq \mathcal{Z}$, then M satisfies ϵ-$d_{\mathcal{X}}$-privacy.

$$\frac{\Pr[M(x) \in Z]}{\Pr[M(x') \in Z]} \leq \exp\left(\epsilon d_{\mathcal{X}}(x, x')\right) \tag{1}$$

In $d_{\mathcal{X}}$-privacy, the difficulty in identifying any two data depends on the distance between them. For data with a slight distance, the distribution of the outputs of the mechanism is closer, and it is challenging to distinguish which of the two is the actual data by only observing the outputs. The exponential mechanism on $\mathcal{X}$ was proposed as a mechanism that satisfies $d_{\mathcal{X}}$-privacy.

Definition 1 (The Exponential Mechanism on $\mathcal{X}$)**.** For the privacy parameter $\epsilon \in \mathbb{R}^+$, the exponential mechanism $\mathcal{M}^{\epsilon}_{d_{\mathcal{X}}}$ on $\mathcal{X}$ outputs $x' \in \mathcal{X}$ with a probability proportional to $\exp\left(-\frac{\epsilon}{2} d_{\mathcal{X}}(x, x')\right)$ when data $x \in \mathcal{X}$ is input.

Table 2 Comparison with existing studies

	Indistinguishability	User input parameters
[11]	At least k data that are output at the same time	k, spatial and temporal tolerances
[12]	All locations within an output circle	Relevance (relative accuracy loss)
[13]	Any two points in the safe region	Safe region and ϵ in it
[1], [2]	Any two regions in PLS	AE common to all regions
[3]	Any two regions within the output range	Output range and granularity
ours	Regions within r_x from output location	AE per region

$$\Pr\left(\mathcal{M}^{\epsilon}_{d_{\mathcal{X}}}(x) = x'\right) = \frac{\exp\left(-\frac{\epsilon}{2} d_{\mathcal{X}}(x, x')\right)}{\sum_{x'' \in \mathcal{X}} \exp\left(-\frac{\epsilon}{2} d_{\mathcal{X}}(x, x'')\right)}$$

Andrés et al. [7] proposed Geo-Indistinguishability (GeoI), which applied $d_{\mathcal{X}}$-privacy to location information. In this definition, $\mathcal{X}$ is regarded as a set of locations on a two-dimensional plane and $d_{\mathcal{X}}$ as the Euclidean distance.

3.3 Privacy Personalization

In this section, we present studies on the personalization of privacy.

In a study on databases, smooth sensitivity [8] was proposed to vary the amount of noise in each dataset. In addition, personalized differential privacy [9] was proposed to allow users to specify the degree of privacy protection required for databases. Hassan et al. [10] studied threats to the mechanism introduced in fitness-tracking social networks such as Strava, which allows users to specify the location that they want to protect privacy.

Gedik et al. [11] proposed a method to personalize k-anonymity for location. A user specifies k in k-anonymity, which represents the privacy strength, spatial tolerance (d_x, d_y) and temporal tolerance (d_t). d_x and d_t are parameters to guarantee that the $k-1$ locations to be indistinguishable are only d_x in the x direction and d_y in the y direction away from the actual location, respectively. d_t is a parameter to guarantee that $k-1$ data are only d_t apart in time from actual data. These are the utility requirements from the user. A trusted system receives user identifiers, users' actual locations, and their privacy preferences one after another, and performs perturbation that satisfies conditions of user preferences by searching for a clique based on the constraint graph constructed from the preferences.

Ardagna et al. [12] proposed a method to achieve privacy personalization in a setting where users can specify the degree of privacy protection with a metric called "relevance", which is measured by a relative accuracy loss. In [12], the location information is measured in the form of a circle containing the actual location of the user due to the measurement error. The system determines the radius and location of a circle to ensure that "relevance" metric is less than the user specified value, and outputs the circle as the user's location information.

Chen et al. [13] proposed personalized local differential privacy (PLDP), where data is aggregated in the settings of privacy personalization and LDP. In PLDP, a user specifies a safe region, such as "Kyoto Prefecture" or "Sakyo Ward,", and differential privacy is guaranteed to be satisfied at any two points within the safe region. Based on this setup, they proposed a mechanism specialized for aggregating data and obtaining distributions.

Yu et al. [1] stated that in addition to GeoI, developing a mechanism that considers the AE for each region is essential. They proposed PIVE, a method to develop a mechanism such that the AE required by the user (common for the whole map) is satisfied. Zhang et al. [2] demonstrated that the method proposed by [1] may not satisfy differential privacy, and proposed DPIVE, which improves PIVE such that differential privacy is satisfied in all regions.

Pappachan et al. [3] proposed Customizable Robust Geo-Indistinguishability (CORGI), which allows users to specify the degree of privacy protection in detail. In [3], a user-specified three parameters for privacy requirements: the output range, the output granularity, and more detailed preferences for the output (e.g., regions not to be output). The system calculates the general mechanism for GeoI based on the output range and granularity by solving a linear programming problem. The client customizes the mechanism based on the third parameter to realize personalization for each region.

The differences between these related studies and the problem settings we are attempting to address are summarized in Table 2. Two studies that are particularly relevant to this study are DPIVE (PIVE) [1], [2] and CORGI [3]. In [1], [2], users could only specify AE common to all regions and not specify each location's privacy protection degree. In addition, [3] did not consider AE; however, many works [1], [2], [14], [15] have noted that it is insufficient to satisfy differential privacy and it is crucial to consider AE. To the best of our knowledge, no research on the problem settings allows users to specify the degree of privacy protection in the form of AE for each location.

4. Privacy Criterion

In this section, we describe the privacy criteria used in this study. In the problem setting of our study, the mechanism used by a particular user is unknown to the adversaries. Consequently, the left side of $d_{\mathcal{X}}$-privacy (Eq. (1)) is unknown to the adversaries, so satisfying $d_{\mathcal{X}}$-privacy formula is not considered to be an appropriate measure of the strength of privacy. Moreover, we believe that it is difficult to achieve location-specific privacy for various user requirements because of its definition formula (Eq. (1)). Therefore, in this study, we use AE (Eq. (2)) and entropy (Eq. (3)) as

the privacy criterion.

4.1 Adversarial Error (AE)

First, we define adversarial error (AE), a measure of user privacy widely used in privacy studies. Intuitively, AE represents the expected distance in the map between the user's actual region and the region predicted by an adversary observing the user's noisy location using the mechanism M. In this study, we consider $AE(x)$, an error in the adversary's prediction for each region instead of the entire map.

Definition 2 (AE per Region ($AE(x)$))**.** When a user is in the region $x(\in \mathcal{X})$, the AE per region ($AE(x)$) can be calculated using the following equation:

$$AE(x) = \sum_{x', \hat{x} \in \mathcal{X}} \Pr(M(x) = x') \Pr(h(x') = \hat{x}) d_p(\hat{x}, x)$$

where h is a function of the adversary's prediction, and $\Pr(h(x') = \hat{x})$ is the probability that the adversary's prediction is $\hat{x}$ when observing the noisy region x'.

To obtain h, an attack using Bayesian inference or the optimal attack [16] to minimize the average AE was considered. However, because these attacks use the actual mechanism utilized by a user, an adversary oblivious to the mechanism attacks by assuming that the user's output region is the user's actual region, hereinafter referred to as the naive attack. In this case, $\Pr(h(x) = x) = 1$, and thus $AE(x)$ can be obtained using the following equation:

$$AE(x) = \sum_{x' \in \mathcal{X}} \Pr(M(x) = x') d(x, x') \tag{2}$$

From the perspective of data users, AE is an metric directly related to the utility of the data. Therefore, $AE(x)$ should be as small as possible while satisfying the user's required AE to ensure utility.

4.2 Entropy

In this section, we introduce the second privacy criterion, entropy. In [17], it was argued that GeoI and AE are insufficient to protect privacy, and that privacy criteria based on entropy should also be considered. In this study, we utilize a privacy criterion based on entropy. When the output of the mechanism is x', entropy is expressed by the following equation:

$$H(x|x') = -\sum_{x \in \mathcal{X}} \Pr(x|x') \log\left(\Pr(x|x')\right)$$

where $\Pr(x|x')$ denotes the posterior probability that the user's actual location is x when the user's output is x'. In this study, we adapt this entropy to the problem settings and use entropy per region as a privacy criterion.

Definition 3 (Entropy per region)**.** If the user's actual region is x and the set of candidates for the user's actual region predicted by an adversary is denoted as C_x $(\subseteq \mathcal{X})$, the entropy for region x is calculated by the following formula:

$$H(x) = -\sum_{x' \in C_x} \frac{\pi(x')}{\sum_{x'' \in C_x} \pi(x'')} \log\left(\frac{\pi(x')}{\sum_{x'' \in C_x} \pi(x'')}\right) \tag{3}$$

Intuitively, this criterion indicates how adequately an adversary can narrow down candidates of the user's actual region when observing the user's output. Determining whether the prediction is accurate is irrelevant. Specifically, this criterion takes the maximum value $\log|\mathcal{X}|$ when the posterior probability is equal for all regions and the minimum value of zero when a region's posterior probability is one.

5. Proposed Mechanisms

In this section, we propose two mechanisms for the problem in this study.

In the differential privacy framework, the average error between the user's actual location and the output location is moderately guaranteed to the data users because the specific mechanism used by the user is obtained. However, in the problem settings of this study, mechanisms are concealed; thus, there is no guarantee to the data users regarding the utility of the data. Therefore, to guarantee that the output region x' is at most r_x away from the user's actual region x, we restrict the range of x' to within a certain radius r_x. When outputting the location, the user transmits the r_x and x' pair (x', r_x) to the server. Considering the perspective of data utility, a smaller r_x is adequate; however, it must sufficiently large to satisfy the AE requirement of the user.

Subsequently, we propose the individual exponential mechanism and donut mechanism to allow users to specify the degree of privacy protection for each location. These two mechanisms ensure that AE in each region x is greater than or equal to $req_{err}(x)$ specified by the user.

5.1 Individual Exponential Mechanism

The first mechanism is called the individual exponential mechanism (IExpM), which uses the exponential mechanism in Definition 1 for privacy personalization. As mentioned above, the exponential mechanism allows adjusting the output data accuracy by changing the parameter ϵ. Using this property, we can obtain ϵ by solving the following equation when a region x is input, such that AE requirement $req_{err}(x)$ for x is satisfied.

$$\sum_{x' \in \mathcal{X}} \Pr(\mathcal{M}_d^{\epsilon}(x) = x') d(x, x') = req_{err}(x) \tag{4}$$

In Eq. (4), the left side is AE due to the naive attack in Eq. (2). However, the maximum error would become immense, depending on the map's size, which results in utility degradation. Therefore, we limit the output range of the mechanism to a radius r_x and output this r_x as the maximum error. A smaller r_x is preferable for data users.

Therefore, we determine the smallest possible r_x that

Table 3 Attacks against DONM

Attack method	Assumptions at the time of attack	Outcome for an adversary
Maximum error attack	Multiple outputs from region x.	Identification of the set C_x of candidate regions x.
Maximum error aggregation attack	Multiple outputs from any regions on the map	Identification of an area where the user has specified a large AE
Output range reference attack	Multiple outputs from region x and mechanism construction method	Further refinement of C_x in maximum error attack

satisfies the AE requirement. This r_x can be discovered using a simple method, such as a binary search. In this case, by determining whether a solution to Eq. (4), we can inspect whether there exists ϵ that satisfies the user requirement in the current r_x.

However, in IExpM, the actual region of the user can be exposed by observing the outputs from the same region multiple times. Specifically, the candidates for the user's actual region can be narrowed down by collecting data that are considered output from the same region based on the maximum error and consider regions common set within a radius r_x from each output region. Note that such an attack succeeds when the adversary can identify outputs from the same region. If the user specifies the same degree of privacy requirements for regions' proximity, the attack will not necessarily be succeed.

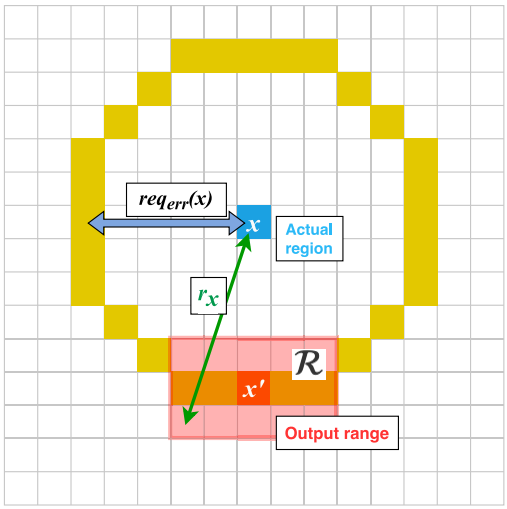

Fig. 1 Donut mechanism.

5.2 Basic Idea of Donut Mechanism

The second mechanism is called the donut mechanism (DONM). In IExpM, output range depends only on the user's actual region and it renders the mechanism vulnerable to the attack mentioned in Sect. 5.1. Therefore, in DONM, the mechanism is designed such that the output range depends not on the actual region but on regions determined from the actual region (which may be selected even when the actual region is another region). Specifically, the output range of the mechanism is a rectangle whose "center" is a region of distance approximately $req_{err}(x)$ in a randomly selected direction. The user randomly outputs a region within the output range. First, we outline the algorithm for developing this mechanism based on Fig. 1.

The algorithm considers as input the user's actual region x and the privacy requirement $req_{err}(x)$ for x, and outputs a rectangle $\mathcal{R}$ representing the output range from x. First, we randomly select x' from among the regions whose distance from x is approximately $req_{err}(x)$ (the yellow regions in Fig. 1). The best rectangle that includes x' is then selected as the output range. In Fig. 1, a peach-colored rectangle $\mathcal{R}$ marked as "Output range" is selected. Subsequently, a region selected uniformly at random from the regions in $\mathcal{R}$ is output as the user's noisy location, concurrently with the maximum distance r_x between x and the region in $\mathcal{X}$. In this mechanism, once a rectangle is determined, it is fixed afterwards, and no new output range is calculated.

5.3 Attacks against DONM

In this section, we introduce possible attacks on the DONM, called maximum error attack, maximum error aggregation attack and output range reference attack. Table 3 summarizes the conditions necessary for these three attacks and the information the adversary can obtain after an attack. In this study, we assume that the maximum error aggregation attack is the attack made under the strongest assumption, and does not consider adversaries with more information than this attack.

5.3.1 Maximum Error Attack

A maximum error attack is an attack that predicts a user's actual region by integrating the multiple outputs from the same region. We call this the maximum error attack because it is performed using the maximum error output. A conceptual diagram of maximum error attack against DONM is shown in Fig. 2. The specific attack procedure is as follows:

1. Assuming that the outputs with equal maximum error r_x are from the same region, select all outputs from the same region (let x be that region).
2. Considering outputs selected in 1, identify the output range rectangle $\mathcal{R}$ from x (the orange area of Fig. 2)
3. Find the set C_x of regions whose maximum distance from the region in $\mathcal{R}$ is r_x (the pink area of Fig. 2).

C_x obtained in Procedure 3 is a set of candidates for the actual region of the user. While no output-limited regions exists, such as oceans, the set contains multiple regions and cannot be reduced. We can estimate this attack's performance using the entropy exhibited in Eq. (3). Note that this maximum error attack assumes the worst-case in which an adversary can identify outputs from the same region based

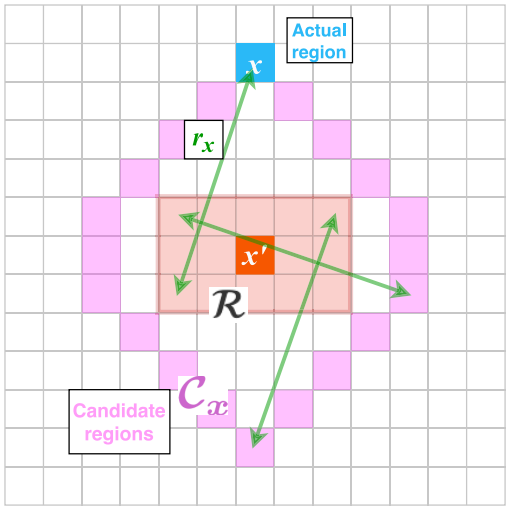

Fig. 2 Maximum error attack.

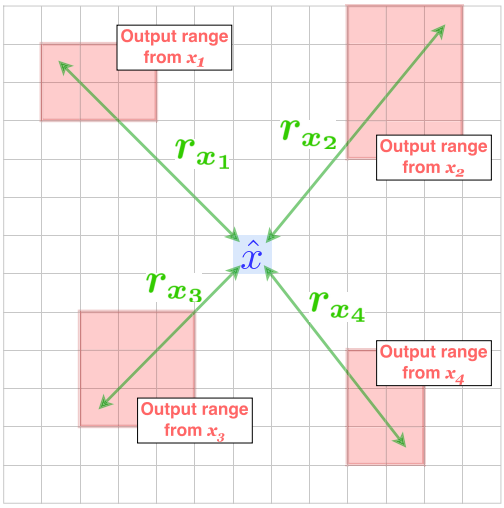

Fig. 3 Maximum error aggregation attack example.

on the maximum error, and an attack on IExpM explained in Sect. 5.1.

5.3.2 Maximum Error Aggregation Attack

A maximum error aggregation attack is an attack that integrates multiple outputs from any region in a map by a user and roughly predicts the region where the user specifies strong privacy protection. Specifically, this attack generalizes the maximum error attack described in Sect. 5.3.1 and considers all outputs from the same user. The specific attack procedure is as follows:

1. Extract all outputs from a user and execute maximum error attack to predict the output range rectangle from each region.
2. In addition, create a set that stores the regions on distance r_x from the output range for each region x.
3. Based on the set created in Procedure 2, compute the average of the maximum error when included in the candidates for each region.
4. The region with a significant average of the maximum error calculated in Procedure 3 is likely to be a sensitive region where the user has specified a high degree of privacy protection.

Because Procedure 3 is a bit complicated, we use a simple example to illustrate it.

Figure 3 exhibits an example of performing Procedure 3. $\hat{x}$ is a region whose maximum distance from the output range for the four regions x_1, x_2, x_3, x_4 is $r_{x_1}, r_{x_2}, r_{x_3}, r_{x_4}$, respectively; and is a candidate for the user's actual region, which is limited using each of the four output ranges. Usually, the immense the AE specified by the user, the larger the maximum error. Therefore, the average maximum error $(r_{x_1} + r_{x_2} + r_{x_3} + r_{x_4})/4$ for $\hat{x}$ can be utilized as a criterion for the user-specified AE in $\hat{x}$. Wherever, the user-specified AE are considerable in the surrounding regions, the average maximum error value may be significant in $\hat{x}$, which does not indicate that the user strongly wants to protect $\hat{x}$. However, considering this case, the adversary can realize that the user strongly wants to protect the region surrounding $\hat{x}$.

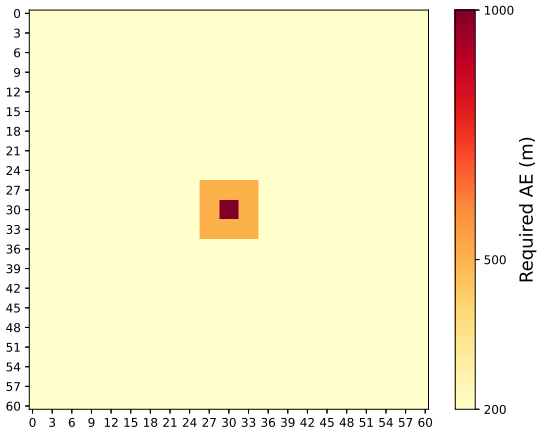

Fig. 4 The central area should be strongly protected.

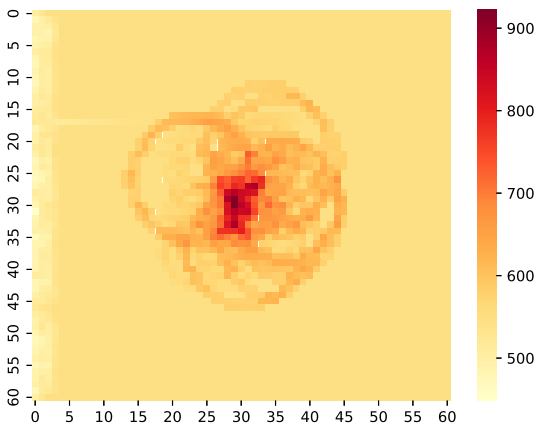

Fig. 5 Result of applying Procedure 3 to Fig. 4.

Results of this attack are shown in Fig. 4 and Fig. 5. In this example, for simplicity, we assume that the user se-

lects AE requirements between 200m, 500m, and 1,000m. Figure 4 exhibits the privacy protection requirements for the case where a user expects to strongly protect the center area in a 61×61 grid. The result until procedure 3 of maximum error aggregation attack for this requirement is shown in Fig. 5. The average maximum error is considerable near the center of the grid, and the adversary can determine that the grid's center area comprises sensitive user information.

Note that this attack is based on hypothesis that a user outputs location information multiple times from any area: it is occasionally impossible to locate a location that retails sensitive user information. Therefore, this paper only describes the attack method, and proposing countermeasures against this attack is a future work.

5.3.3 Output Range Reference Attack

An output range reference attack is an attack in that an adversary refers to output range information from all region in a grid. The adversary is assumed to have prior knowledge of the output range in DONM for a given combination of the region and AE specification in that region. While the actual mechanism utilized by a specific user is hidden: how the mechanism is developed can be revealed. Therefore, adversaries with prior knowledge must be considered. In output range reference attack, an adversary uses maximum error attack to discover an output range rectangle $\mathcal{R}$ and then locates a set of candidate regions C_x. By selecting regions from $\mathcal{R}$ where the output range can be C_x, the adversary can limit the candidates by more than C_x.

5.4 Algorithm for DONM

For the naive idea of DONM, described in Sect. 5.2, attacks described in Sect. 5.3 are expected, especially it is necessary to adopt countermeasures against the maximum error attack and output range reference attack. We should consider the following points when developing the mechanism to counter each attack.

- Maximum error attack: Round the maximum error value to output.
- Output range reference attack: When developing the mechanism from the region narrowed down by an adversary, the output range should be the same rectangle.

Before analyzing the specific algorithm description, we provide insight into the differences from Sect. 5.2 by using Fig. 1 and Fig. 2. First, the output maximum error value is rounded (rounded up) and the granularity varies. For example, if the maximum error is output in units of 100m, the maximum error is output as 1,100m, whether it is 1,001m or 1,050m. Therefore, when adversaries select candidates for actual region of the user, it is as shown in Fig. 2, they require selecting regions between 1,001m and 1,100m from an output region, which increases the number of candidate regions. This outcome increases the entropy and counters the maximum error attack. Meanwhile, a large granularity of the maximum error decreases the utility of the data; therefore, it is necessary to set an appropriate value in sequence with the usage. Second, when selecting the output range, we consider the average errors caused by the naive attack in the candidate regions limited by the maximum error attack. The adversary uses the maximum error attack to limit the candidates of the user's actual region to the peach-colored region in Fig. 2. When the same $req_{err}(x)$ is utilized to develop the DONM from each of these candidate regions, and the same region x' is assumed to be selected as the center of the output range; the output range must also be $\mathcal{R}$. The selection of the output range rectangle is unrelated to the user's actual region but depends on x'. This aspect is to counteract the output range reference attack. Specifically, when calculating the metric for selecting rectangle $\mathcal{R}$ as the output range, we use the average of the naive errors from all peach-colored regions to $\mathcal{R}$.

Based on the above explanation, the algorithm for developing donut mechanism is presented in Algorithm 1. Note that $d_{max}(\hat{x})$ in Algorithm 1 represents the maximum distance from $\hat{x}$ ($\in \mathcal{X}$) to the region inside the rectangle $\mathcal{R}'$ on the grid. The function compute_naive_error calculates the naive error using Eq. (2). The compute_entropy function calculates the entropy of the adversary's prediction using Eq. (3), and compute_metric function calculates a metric

Algorithm 1 Compositional algorithm of donut mechanismfor a single region

Input: User's actual region x, Privacy protection requirement for x $req_{err}(x)$, The set of all regions $\mathcal{X}$, prior distribution π for each region, Selection range of output center δ, Rectangular search range radius r, The maximum error output unit err_{unit}
Output: Rectangle $\mathcal{R}$ that will be the output range

$\mathcal{O} \Leftarrow$ Randomly sorted list of regions consisting of x' that satisfy $req_{err}(x) \leq d(x, x') \leq req_{err}(x) + \delta$
for x' **in** $\mathcal{O}$ **do**
  $R \Leftarrow$ The list of rectangles containing x' and inside a circle of radius r from x'
  $metric\_list \Leftarrow []$
  for $\mathcal{R}'$ **in** R **do**
    $err_{naive} \Leftarrow$ compute_naive_error$(x, \mathcal{R}')$
    if $err_{naive} \leq req_{err}(x)$ **then**
      $metric\_list.add(0)$
      continue
    end if
    $r_x \Leftarrow$ Maximum distance from x to a region in $\mathcal{R}'$
    $err_{max} \Leftarrow$ The smallest value that is a multiple of err_{unit} and greater than or equal to r_x.
    $C \Leftarrow \{\hat{x} \in \mathcal{X} | err_{max} - err_{unit} \leq d_{max}(\hat{x}) \leq err_{max} \wedge$ compute_naive_error$(\hat{x}, \mathcal{R}') \geq req_{err}(x)\}$
    $naive\_error\_list \Leftarrow$ [compute_naive_error$(c, \mathcal{R}')$ **for** c **in** C]
    $m \Leftarrow$ mean$(naive_error_list)$
    $e \Leftarrow$ compute_entropy(C)
    $metric_list.add($compute_metric$(m, e, err_{max}, \pi))$
  end for
  if max$(metric_list) \neq 0$ **then**
    $i \Leftarrow$ arg max$(metric_list)$
    $\mathcal{R} \Leftarrow R[i]$
    return $\mathcal{R}$
  end if
end for
return None

that expresses the "advantages" of a rectangle $\mathcal{R}'$ as the output range. The function mean is a function that extracts the mean of the elements in the list of arguments.

The algorithm considers as input the user's actual region x, the privacy protection requirement for the region $req_{err}(x)$ and other parameters, and outputs a rectangle $\mathcal{R}$ representing the output range from the region x. First, we develop a set of regions whose distance from x is approximately $req_{err}(x)$ and then randomly sort the set into O in line 1. A parameter δ can adjust the maximum distance between the candidate region and $req_{err}(x)$. A large delta leads to an increase in the likelihood that a location with a distance from the user's region x greater than $req_{err}(x)$ will be selected as x'. Consequently, the possibility that a rectangle of the output range is constructed in the area with a large distance from x is increased, and the utility is reduced. However, it is difficult to determine δ theoretically, and considered that δ is often determined empirically.

Next, we inspect whether we can develop an output range that satisfies the user's privacy requirements for each element x' of O in lines 2 to 24. In line 3, we create a list R of all rectangles that are output range candidates from x. In lines 5 to 18, for each rectangle $\mathcal{R}'$ ($\in R$), we compute the metric of "advantages" to be the output range. In line 7, we determine whether the error of the naive attack err_{naive} satisfies the user's requirements. If it does not satisfy the user's requirement, the metric value is set to zero. If it is satisfied, we prepare to compute the metric in line 11 and thereafter. In line 13, we discover candidate regions by the maximum error attack and select a set of regions C where the naive errors are more than $req_{err}(x)$ when the output range from the candidate region is $\mathcal{R}'$. This is because, in region $\hat{x}$ ($\in C$), if the naive error is less than $req_{err}(x)$, then the $\mathcal{R}'$ is never selected as the output range from $\hat{x}$. In lines 14 and 15, we compute the naive error when each region in C is input and the average value of naive errors is calculated, which is utilized when calculating the metric in line 17. If only the naive error from the actual region x is used to calculate the metric, the "best" rectangle may differ from that of the region c ($\in C$). In this case, c is excluded from the candidates for the adversary's prediction by the output range reference attack. Therefore, we require a measure of the naive error that is independent of x. This outcome makes the output range reference attack impossible. Finally, the rectangle with the highest metric is selected as the algorithm's output in lines 19 to 23. Note that if any x' and any rectangle $\mathcal{R}'$ cannot satisfy the user requirement, the output range cannot be determined; therefore none is output.

6. Experiments

In this section, we compare the effectiveness of IExpM and DONM using two experiments. One compares IExpM and DONM regarding the naive error, maximum error, and execution time. The other investigates the change in entropy by changing the output granularity of the maximum error in DONM, as described in Sect. 5.4. The experiments in this section were performed on a MacBook Pro with an Apple M1 chip and 16GB of memory.

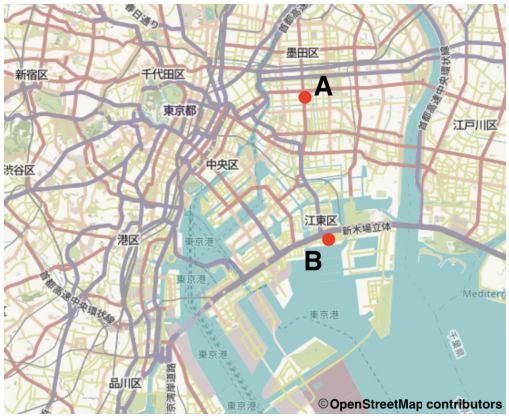

Fig. 6 Map of the area used in the experiment.

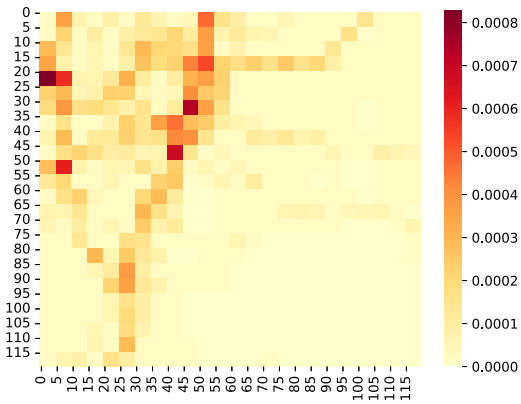

Fig. 7 Prior probability of the area in Fig. 6.

6.1 Settings of Experiments

In the experiments, we used an artificial map with uniform prior distribution, and an actual map of Tokyo with non-uniform prior distribution. The actual map of the region used in the experiment is depicted in Fig. 6. These two maps are represented by a 120×120 grid, such that each region is a rectangle with a height of 115.625m and 141.5m in width. The size is determined by the mesh[†], used by the Japanese government to organize statistical data.

We calculated the prior probabilities for regions in an actual map in units of five $\times$ five regions using the "People Flow 2008 Tokyo Metropolitan Area" by the People Flow Project [18][††]. This dataset is a collection of trajectory data for people in the Tokyo metropolitan area collected on October 1, 2008. The results of the prior probability calculations are in Fig. 7 using a heat map.

The output range is the area excluding the sea in the IExpM and the area within the rectangle excluding the sea in the DONM (with such constraints, $\mathcal{R}$ and ϵ are calculated),

[†]https://www.stat.go.jp/data/mesh/pdf/gaiyo1.pdf
[††]https://pflow.csis.u-tokyo.ac.jp/

to avoid outputting over the sea.

In the first experiment, we vary the AE required by a user for a region in the center of the artificial map (denoted as region C), and for two regions in the map of Tokyo that are sufficiently far from the sea (region A) and close to the sea (region B). The approximate locations of the regions A and B on the map of Tokyo are indicated by red dots in Fig. 6). We apply IExpM and DONM in three regions and compare the naive error, maximum error, and execution time for each mechanism. We utilize 200m, 500m, and 1,000m as the values of AE. In the DONM, from the perspective of privacy, only those with entropy (Eq. (3)) greater than 2.5 when $req(x) = 200$(m), 3 when $req(x) = 500$(m), and 3.5 when $req(x) = 1,000$(m), respectively.

Because the construction of DONM is stochastic, the naive and the maximum errors are calculated as the average of the results of 10 runs. In both mechanisms, the running time is compared with the average of the results of 10 runs. The parameters required to develop DONM are set to $\delta = 182.73$(m), $r = req_{err}(x)$, $err_{unit} = 182.73$(m). The value 182.73(m) corresponds to the length of the diagonal of one region. By setting this value to δ, it is expected that regions in O are in every direction from x and each of which is not too far from x. In addition, in the development of DONM, Eq. (5) is used to select the best rectangle in line 17 of Algorithm 1.

$$\begin{aligned}&\text{compute\_metric}(m, e, err_{max})\\ &\quad = -w_m m^{norm} - w_{err_{max}} err_{max}^{norm} + w_e e^{norm} \end{aligned} \tag{5}$$

In Eq. (5), m^{norm}, e^{norm}, and err_{max}^{norm} are m, e, and err_{max} normalized using mean and variance, respectively. $w_m, w_e, w_{err_{max}}$ are the weights for m^{norm}, e^{norm}, and err_{max}^{norm}, respectively. By varying each weight w_m, w_e, and $w_{err_{max}}$, we can specify which privacy and utility metric is important. When w_m is larger than the other weights, the rectangle of the output range is selected with emphasis on the small average error (high utility). When w_e is larger than the other weights, the rectangle with large entropy will be more likely to be selected (high privacy strength). When $w_{err_{max}}$ is larger than the other weights, the rectangle with small maximum error will be selected (high utility). It is also possible to extend the system in such a way that the user specifies these values in some way. In the experiment, we use $w_m = w_e = w_{err_{max}} = 1$.

In the second experiment, we investigate the effect on the entropy of varying the output granularity of the maximum error in DONM. We search for rectangles that are candidates for the output range for all regions in DONM and calculate the distribution of entropy when the output range is a rectangle that satisfies the user's requirements. This outcomes enabled us to examine the difficulty of developing a mechanism with high entropy for different output granularities. In this experiment, we consider the cases when the output granularity of the maximum error is 0m, 50m, 182.73m, 500m, and 2,000m for region A in the first experiment. Generally, when the output granularity of the maximum error becomes coarse, it is challenging for an adversary to distinguish whether the outputs are from the same region in the maximum error attack. However, this experiment examines the worst-case scenario, where adversary can identify outputs from the same region. Note that when the output granularity is 2,000m, we assume the maximum error output from all regions is 2,000m. This is a unique case where the adversary cannot identify which outputs are from the same region; entropy can be computed by assuming that all regions within a radius of 2,000m are candidate regions. We utilize $req_{err}(x) = 1,000$(m) and $r = 500$(m) as the experimental parameters. We utilize the same values as those in the first experiment for the other parameters.

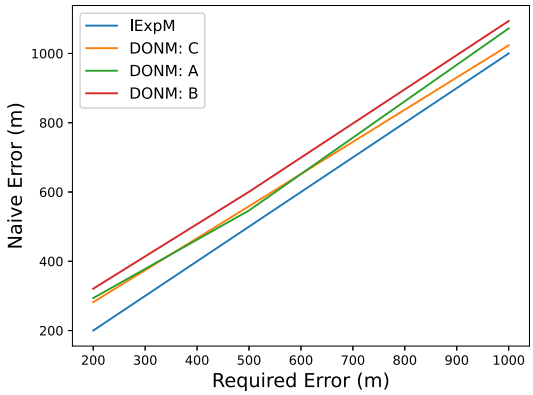

Fig. 8 Comparison of naive error.

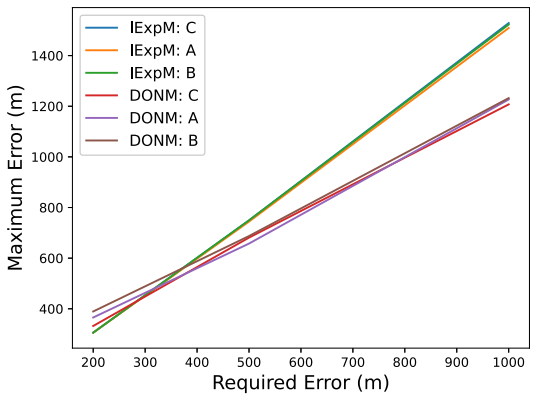

Fig. 9 Comparison of maximum error.

6.2 Results

In this section, we present the results of the experiments described in Sect. 6.1.

The results of the experiments comparing IExpM and DONM are shown in Fig. 8 to Fig. 10. Figure 8 and Fig. 9 illustrate the results of plotting naive errors and maximum errors for each combination of mechanisms and regions A, B, and C. The smaller these values are above AE required by the user, the better they are from the perspective of the data utility.

From Fig. 8, we note that IExpM is better regarding naive errors. This aspect is because, in IExpM, the mechanism is obtained by solving Eq. (4) so that the naive error

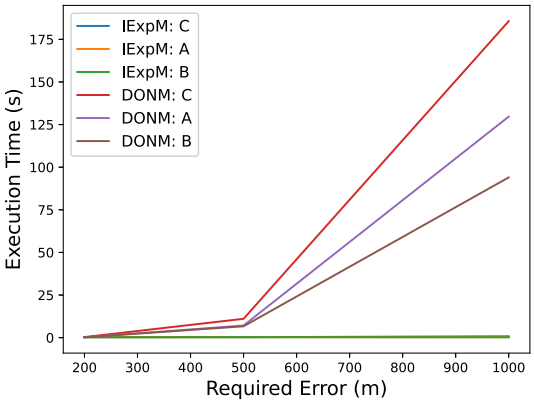

Fig. 10 Comparison of execution time.

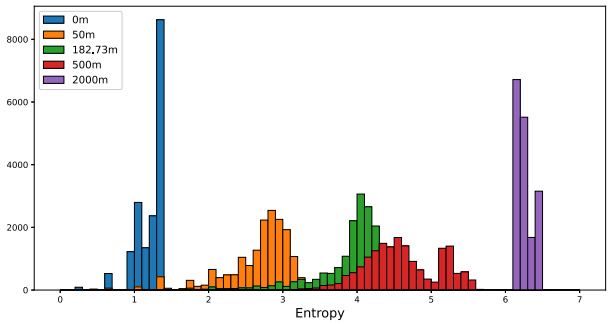

Fig. 11 Comparison of entropy distributions for varying output granularity of maximum error (region A)

by the adversary is equal to the user's requirement. However, in DONM, the output range is determined in units of rectangles; therefore, it is difficult to adjust the naive error. From Fig. 9, we note that the maximum error can be reduced by using DONM. This result is because IExpM outputs the user's actual region x and its surroundings; therefore, it is necessary to output the region farther away to satisfy the user's requirement. Meanwhile, DONM only outputs around regions such that the distance from x is $req_{err}(x)$.

Figure 10 depicts the execution time for each combination of mechanisms, and regions A, B, and C. It is evident from the figure that IExpM is much better than DONM regarding execution time. This feature is because, IExpM locates the mechanism by solving unitary equations or binary searches, which are computationally inexpensive, whereas DONM searches for all candidate rectangles and selects the most suitable. The reason why the execution time of region B is shorter than that of the other two regions is considered that the presence of the ocean in the surrounding area reduces the number of candidates for rectangle search. In addition, because the biased prior distribution in region A allows the elimination of candidate rectangles with small entropy before computing the metric, the computation time in region A is shorter than that in region C, which assumes a uniform prior distribution.

Figure 11 presents the results of the second experiment. As depicted in Fig. 11, the output granularity of the maximum error becomes coarser (i.e., the output unit becomes immense), the number of rectangles that can guarantee a high entropy increases. Therefore, it can be noted that increasing the output unit of the maximum error is an effective countermeasure against the maximum error attack.

7. Limitations of Proposed Mechanisms

In this section, we examine the limitations of these two proposed mechanisms. IExpM is vulnerable to the maximum error attack and has difficulty outputting the maximum error r_x from the actual region each time it outputs the location. Thus, it can only output the maximum error for the entire region, and it is challenging data users to realize the amount of noise added to the output. This outcome makes it challenging to handle the data.

In the DONM, the naive error can be adjusted only on a regional basis: this may cause the actual naive error value to be more considerable than the value required by the user, thereby reducing the utility of the data. Although the mechanism itself can be pre-computed and does not require recomputation each time the location information is output. The calculation of the mechanism is computationally more expensive than IExpM because it requires searching all candidates of rectangles to find the optimal output range.

A common limitation is that when using errors to guarantee data utility for data users, attacks such as the maximum error attack, maximum error aggregation attack, output range reference attack are possible. In this study, we primarily output the maximum error. However, the same argument can be applied to the case where the average distance between the user's actual region and the region within the output range is the output. Generally, we consider that outputting additional information to guarantee data utility inevitably provides opportunities for such attacks.

8. Conclusion

In this study, we proposed a framework that allows users to specify the degree of privacy protection for each region as AE. To satisfy this requirement, we propose two mechanisms. One is IExpM, which utilizes the exponential mechanism, and the other is a mechanism that selects a region in a randomly selected direction and designates the rectangle, including the region as the output range. We subsequently describe possible attack methods against the latter mechanism and define a mechanism incorporating approaches to counter these attacks. Finally, we demonstrate that privacy personalization can be realized using the two mechanisms; that IExpM is superior regarding the naive error; DONM is superior regarding the maximum error. We also demonstrate that it is possible to counter maximum error attack by changing the granularity of the maximum error output.

There are two potential topics for future research. First, we should enable IExpM to output the maximum error each time location information is transmitted. The second is to extend the output range of DONM to other than rectangular.

Future research should address these two challenges.

Acknowledgments

This work was supported by JST CREST (No. JPMJCR21M2), JST SICORP (No. JPMJSC2107), and JSPS KAKENHI (No. 21J23090, 21K19767, 22H03595).

References

[1] L. Yu, L. Liu, and C. Pu, "Dynamic Differential Location Privacy with Personalized Error Bounds," Proceedings 2017 Network and Distributed System Security Symposium, San Diego, CA, Internet Society, 2017.

[2] S. Zhang, B. Duan, Z. Chen, T. Ni, and H. Zhong, "Regionalized location obfuscation mechanism with personalized privacy levels," Jan. 2022. arXiv:2102.00654 [cs].

[3] P. Pappachan, C. Qiu, A. Squicciarini, and V.S.H. Manjunath, "User Customizable and Robust Geo-Indistinguishability for Location Privacy," Oct. 2022. arXiv:2206.08396 [cs] version: 2.

[4] C. Dwork, "Differential Privacy: A Survey of Results," in Theory and Applications of Models of Computation, pp.1–19, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[5] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking," Proceedings of the 1st International Conference on Mobile Systems, Applications and Services, MobiSys '03, New York, NY, USA, pp.31–42, Association for Computing Machinery, 2003.

[6] K. Chatzikokolakis, M.E. Andrés, N.E. Bordenabe, and C. Palamidessi, "Broadening the Scope of Differential Privacy Using Metrics," Privacy Enhancing Technologies, vol.7981, pp.82–102, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[7] M.E. Andrés, N.E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Geo-indistinguishability: differential privacy for location-based systems," Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security - CCS '13, Berlin, Germany, pp.901–914, ACM Press, 2013.

[8] K. Nissim, S. Raskhodnikova, and A. Smith, "Smooth sensitivity and sampling in private data analysis," Proceedings of the thirty-ninth annual ACM symposium on Theory of computing - STOC '07, San Diego, California, USA, pp.75–84, ACM Press, 2007.

[9] Z. Jorgensen, T. Yu, and G. Cormode, "Conservative or liberal? personalized differential privacy," 2015 IEEE 31St international conference on data engineering, pp.1023–1034, IEEE, 2015.

[10] W.U. Hassan, S. Hussain, and A. Bates, "Analysis of privacy protections in fitness tracking social networks-or-you can run, but can you hide?," 27th USENIX Security Symposium (USENIX Security 18), pp.497–512, 2018.

[11] B. Gedik and L. Liu, "Protecting location privacy with personalized k-anonymity: Architecture and algorithms," IEEE Transactions on Mobile Computing, vol.7, no.1, pp.1–18, 2008.

[12] C.A. Ardagna, M. Cremonini, S.D.D Vimercati, and P. Samarati, "An obfuscation-based approach for protecting location privacy," IEEE Transactions on Dependable and Secure Computing, vol.8, no.1, pp.13–27, 2011.

[13] R. Chen, H. Li, A.K. Qin, S.P. Kasiviswanathan, and H. Jin, "Private spatial data aggregation in the local setting," 2016 IEEE 32nd International Conference on Data Engineering (ICDE), Helsinki, Finland, pp.289–300, IEEE, May 2016.

[14] S. Takagi, Y. Cao, Y. Asano, and M. Yoshikawa, "Geo-Graph-Indistinguishability: Protecting Location Privacy for LBS over Road Networks," Data and Applications Security and Privacy XXXIII, vol.11559, pp.143–163, Springer International Publishing, Cham, 2019.

[15] S. Oya, C. Troncoso, and F. Pérez-González, "Is Geo-Indistinguishability What You Are Looking for?," Proceedings of the 2017 on Workshop on Privacy in the Electronic Society, Dallas Texas USA, pp.137–140, ACM, Oct. 2017.

[16] R. Shokri, G. Theodorakopoulos, C. Troncoso, J.-P. Hubaux, and J.-Y.L. Boudec, "Protecting location privacy: optimal strategy against localization attacks," Proceedings of the 2012 ACM conference on Computer and communications security - CCS '12, Raleigh, North Carolina, USA, pp.617–627, ACM Press, 2012.

[17] S. Oya, C. Troncoso, and F. Pérez-González, "Back to the Drawing Board: Revisiting the Design of Optimal Location Privacy-preserving Mechanisms," Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, Dallas Texas USA, pp.1959–1972, ACM, Oct. 2017.

[18] Y. Sekimoto, R. Shibasaki, H. Kanasugi, T. Usui, and Y. Shimazaki, "PFlow: Reconstructing People Flow Recycling Large-Scale Social Survey Data," IEEE Pervasive Computing, vol.10, no.4, pp.27–35, April 2011.

Appendix: Cases Where Users' Requirements are Not Given in AE

In this study, the user specifies a lower bound of AE (distance) for each region. This aspect has the advantage of being more intuitively understandable for users than parameters such as ϵ used in differential privacy. In addition, it is a natural problem setting because AE is widely used to estimate the degree of user's privacy protection. However, there is an aspect that mechanisms are restricted by providing the degree of privacy protection users require regarding AE. For example, in DONM proposed in Sect. 5.2, the output range is limited to regions around x' where the distance from the actual region x is approximately at the user requirement $req_{err}(x)$.

We compare DONM with the following more flexible mechanism: First, x' in DONM is randomly chosen within a radius r of the distance from x (r is a predefined parameter). As in the case of DONM, we develop a rectangle that includes x' as the output range. The maximum distance r_x that is output simultaneously with the region after adding noise is the maximum distance between the rectangles developed by selecting each x' from the region within radius r. Although in many cases, this mechanism will not satisfy the user's requirement specified as AE; it is expected to improve the entropy. This aspect is because the number of candidates of the actual region for which the rectangle may be the output range is more considerable than that in the case of DONM. In addition, because the maximum error r_x is almost similar to that of the DONM, the perceived utility for the data user does not change significantly. However, the actual utility of the data increases because the probabilities of outputting regions close to x are increased.

In addition to the fact that privacy protection requirements differ between persons and regions, there may be preferences on how to specify privacy protection requirements, such as the requirement that AE is above a certain level or the possibility of being identified by an adversary is reduced. Thus, extensions that allow users to select how to specify privacy protection requirements are also considered topics for future work.

Ryota Hiraishi is an masters student in Graduate School of Social Informatics at Kyoto University. He received his BE from Undergraduate School of Information and Mathematical Science at Kyoto University.

Masatoshi Yoshikawa received his BE, ME, and Dr. Eng. degrees from the Department of Information Science, Kyoto University, in 1980, 1982, and 1985, respectively. From 1985 to 1993, he was with Kyoto Sangyo University. In 1993, he joined the Nara Institute of Science and Technology as an associate professor. From 2002 to 2006, he served as a professor at Nagoya University. Since 2006, he has been a professor at Kyoto University. His general research interests are in the area of databases. His current research interest includes privacy protection technologies and personal data market.

Yang Cao is an Associate Professor in the Division of Computer Science and Information Technology at Hokkaido University. He earned his Ph.D. from the Graduate School of Informatics, Kyoto University, in 2017. His research interests lie in the intersections between databases, security, and machine learning. He has published many papers in these areas, including top venues such as VLDB, SIGMOD, ICDE, AAAI, TKDE, and USENIX Security. Two of his papers were selected as one of the best paper finalists in ICDE 2017 and ICME 2020. He is a recipient of the IEEE Computer Society Japan Chapter Young Author Award 2019, Database Society of Japan Kambayashi Young Researcher Award 2021.

Sumio Fujita worked for Computer Institute of Japan Ltd. (Kanagawa, Japan) 1985–1995, studied for DEA at Université de Paris 7 1988–1989, worked as research associate at University of Manchester Institute of Science and Technology 1993–1994, worked as chief researcher at Justsystem Corporation (Tokushima, Japan) 1995–2002, worked as research scientist at Claritech Corporation (Pittsburgh, PA, USA) 1998, and worked as senior research scientist at Patolis Corporation (Tokyo, Japan) 2002–2004. He joined Yahoo Japan Corporation (Tokyo, Japan) in 2005, participated in the foundation of Yahoo! JAPAN Research in 2007 and worked as Senior chief researcher. He currently works as project researcher on information retrieval, web mining and related areas at Yahoo! JAPAN Research. He is a member of ACM, SIGIR and IPSJ.

Hidehito Gomi received the M.Eng. degree from the Division of Applied Systems Science, Faculty of Engineering and the Ph.D. degree in informatics from the Graduate School of Informatics, Kyoto University, Kyoto, Japan, in 1996 and 2012, respectively. From 1996 to 2007, he was a researcher in the laboratories of NEC Corporation. From 2001 to 2003, he was a visiting researcher in the Computer Science Department of Stanford University, CA, U.S.A. In 2007, he joined Yahoo! JAPAN Research, where he is continuing his research on security, privacy, and trust. He is a member of the IEEE, ACM, IPSJ, and IEICE.