PAPER *Special Section on Data Engineering and Information Management*

Geo-Graph-Indistinguishability: Location Privacy on Road Networks with Differential Privacy[*]

Shun TAKAGI[†a)], ***Nonmember***, **Yang CAO**[†], **Yasuhito ASANO**[††], ***and*** **Masatoshi YOSHIKAWA**[†], ***Members***

SUMMARY In recent years, concerns about location privacy are increasing with the spread of location-based services (LBSs). Many methods to protect location privacy have been proposed in the past decades. Especially, perturbation methods based on Geo-Indistinguishability (GeoI), which randomly perturb a true location to a pseudolocation, are getting attention due to its strong privacy guarantee inherited from differential privacy. However, GeoI is based on the Euclidean plane even though many LBSs are based on road networks (e.g. ride-sharing services). This causes unnecessary noise and thus an insufficient tradeoff between utility and privacy for LBSs on road networks. To address this issue, we propose a new privacy notion, Geo-Graph-Indistinguishability (GeoGI), for locations on a road network to achieve a better tradeoff. We propose Graph-Exponential Mechanism (GEM), which satisfies GeoGI. Moreover, we formalize the optimization problem to find the optimal GEM in terms of the tradeoff. However, the computational complexity of a naive method to find the optimal solution is prohibitive, so we propose a greedy algorithm to find an approximate solution in an acceptable amount of time. Finally, our experiments show that our proposed mechanism outperforms GeoI mechanisms, including optimal GeoI mechanism, with respect to the tradeoff.

key words: *location privacy, road network, differential privacy, geo-indistinguishability, local differential privacy*

1. Introduction

In recent years, the spread of smartphones and GPS improvements have led to a growing use of location-based services (LBSs). While such services have provided enormous benefits for individuals and society, their exposure of the users' location raises privacy issues. Using the location information, it is easy to obtain sensitive personal information, such as information pertaining to home and family. In response, many methods have been proposed in the past decade to protect location privacy. These methods involve three main approaches: perturbation, cloaking, and anonymization. Most of these privacy protection methods are based on the Euclidean plane rather than on road networks; however many LBSs such as UBER[**] and Waze[***] are based on road networks to capitalize on their structures [7], [19], [23], resulting in utility loss and privacy leakage when using such methods. Some prior works have revealed this fact [8], [13], [32] and proposed methods that use road networks and are based on cloaking and anonymization. However, cloaking and anonymization also have weaknesses: if an adversary has peripheral knowledge about a true location, such as the range of a user's location, no privacy protection is guaranteed (in detail, we refer to Sect. 7). In this paper, based on differential privacy [9], we consider a perturbation method that does not possess such weakness. First, we review perturbation methods and differential privacy [9], which are the bases of our work; then, we describe the details of our work.

Perturbation methods modify a true location to another location by adding random noise [2], [26] using a mechanism. Shokri et al. [25] defined location privacy introduced by a mechanism, and they constructed a mechanism that minimizes adversaries' inference attack. However, this concept of location privacy depends on specific adversarial knowledge, which cannot guarantee privacy against adversaries with different types of background knowledge.

Differential privacy [9] has received attention as a rigorous privacy notion that guarantees privacy protection against any adversary. Andrés et al. [2] defined a formal notion of location privacy called geo-indistinguishability (GeoI) by extending differential privacy. A mechanism that achieves it guarantees the indistinguishability of a true location from other locations to some extent against any adversary. However, because this method is based on the Euclidean plane, GeoI does not tightly protect the privacy of locations on road networks, which results in a loose tradeoff between utility and privacy. In other words, GeoI protects privacy too much for people on road networks.

GeoI assumes only that the given data is a location from a Cartesian plane, which causes a loose tradeoff between utility and privacy for LBSs over *road networks*. In this paper, we study how to achieve optimal privacy-utility tradeoffs when a user is located on a road network. We model the road network using a graph, and we propose a new privacy definition, called ε-geo-graph-indistinguishability (GeoGI), based on the notion of differential privacy. Additionally, we propose the graph-exponential mechanism (GEM), which satisfies GeoGI. These proposals appeared in the preliminary version [38], [41] of this paper.

This paper extends the preliminary version [38] by considering a mechanism optimized with prior knowledge about

Manuscript received June 26, 2022.
Manuscript revised November 16, 2022.
Manuscript publicized January 16, 2023.

[†] The authors are with the Kyoto University, Kyoto-shi, 606–8501 Japan.

[††] The author is with the Toyo University, Tokyo, 112–8606 Japan.

[*] The preliminary version of this paper appeared in IFIP Annual Conference on Data and Applications Security and Privacy. Springer, Cham, 2019.

a) E-mail: takagi.shun.45a@st.kyoto-u.ac.jp

DOI: 10.1587/transinf.2022DAP0011

[**] https://marketplace.uber.com/matching

[***] https://www.waze.com/ja/

a user and an adversary (Sect. 5 and related experiments of Sect. 6). Existing works [5], [37] proposed optimal mechanisms for GeoI to consider utility and privacy with prior knowledge using Shokri's notion [25], which we call quality loss (Q^{loss}) and adversarial error (AE) while guaranteeing ε-GeoI. Concretely, Yu et al. [37] proposed a privacy guarantee based on AE in addition to ε-GeoI, but its method does not strictly guarantee ε-GeoI as proved by Shun et al. [27]; Bordenabe et al. [5] optimized Q^{loss} while guaranteeing GeoI using δ-spanner to improve the runtime of optimization. We can apply δ-spanner to GeoGI, but the method requires that the number of possible locations input to the mechanism is small (e.g., < 100), which causes utility and privacy loss for a user at a location outside the possible input locations as described in Sect. 6. In addition, none of them considers the effects of road networks on the privacy-utility tradeoffs.

In this paper, we propose a novel method to optimize Q^{loss} and AE while strictly guaranteeing ε-GeoGI by optimizing the output range of GEM. Concretely, although GEM outputs a vertex of a graph that represents a road network, the output range (i.e., set of vertices) is adjustable, which induces the idea that there exists an optimal output range. We analyze the relationship between output range and Shokri's notion. Moreover, we formalize the optimization problem to search the optimal range for AE and Q^{loss}. However, the number of combination of output ranges is $2^{|V|}$, where $|V|$ denotes the size of vertices, which makes it difficult to solve the optimization problem in acceptable time. To this end, we propose a greedy algorithm to find an approximate solution to the optimization problem. The method terminates in an acceptable amount of time comparing with Bordenabe's method [5], which mitigates the utility and privacy loss for a user at a location on a road network outside possible input locations as shown in Sect. 6.

Because our definition tightly considers location privacy on road networks, it results in a better tradeoff between utility and privacy. To demonstrate this aspect, we compare GEM with GeoI mechanisms, including the optimal GeoI mechanisms [5]. In our experiments on three kinds of real-world data, GEM outperforms the baseline w.r.t. the tradeoff between utility and privacy. Moreover, we obtained the prior distribution of a user using a real-world dataset. Then, we show that the privacy protection level of a user who follows the prior distribution can be effectively improved by the optimization.

In summary, our contributions are as follows:

- We propose a privacy definition for locations on road networks, called ε-GeoGI (Sect. 3).
- We propose a graph-exponential mechanism (GEM) that satisfies GeoGI (Sect. 4).
- We analyze the performance of GEM and formalize optimization problems to improve utility and privacy protection (Sect. 5).
- We experimentally show that our proposed mechanism outperforms the mechanisms proposed in [2], [5] w.r.t. the tradeoff between utility and privacy (Sect. 6).

2. Preliminaries and Problem Setting

In this section, we first review the formulations for a perturbation mechanism, empirical privacy gain and utility loss. Next, we describe the concept of differential privacy [9], which is the basis of our proposed privacy notion. Finally, we explain a setting where we define privacy.

2.1 Perturbation Mechanism on the Euclidean Plane

Here, we explain the formulations for a perturbation mechanism, empirical privacy gain and utility loss [26].

2.1.1 User and Adversary

Shokri et al. [26] assumed that user u is located at location $x \in \mathbb{R}^2$ according to a prior distribution $\pi_u(x)$. LBSs are used by people who wants to protect their location privacy but receive high-quality services. The user adopts a perturbation mechanism $M : \mathbb{R}^2 \rightarrow \mathcal{Z}$ that sends a pseudolocation $M(x) = z \in \mathcal{Z}$ instead of his/her true location x where $\mathcal{Z} \subseteq \mathbb{R}^2$. Assume that an adversary a has some knowledge represented as a prior distribution about the user location $\pi_a(x)$ and tries to infer the user's true location from the observed pseudolocation z. In this paper, we assume that the adversary has unbounded computational power and precise prior knowledge, i.e., $\pi_a(x) = \pi_u(x)$. Although this assumption is advantageous for the adversary, protection against such an adversary confers a strong guarantee of privacy.

2.1.2 Empirical Privacy Gain and Utility Loss

The empirical privacy gain obtained by mechanism M is defined as follows, which we call adversarial error (AE).

$$AE(\pi_a, M, h, d_q) = \sum_{\hat{x},x,z} \pi_a(x) \Pr[M(x) = z] \Pr[h(z) = \hat{x}] d_q(\hat{x}, x)$$

where d_q is a distance over $\mathbb{R}^2$ and h is a probability distribution over $\mathbb{R}^2$ that represents the inference of the adversary about the user's location. Thus, intuitively, AE represents the expected distance between the user's true location x and the location $\hat{x}$ inferred by the adversary. Next, we explain the model of an adversary, that is, how an adversary constructs a mechanism h, which is called an optimal inference attack [26]. An adversary who obtains a user's perturbed location z tries to infer the user's true location through an optimal inference attack. In this type of attack, the adversary solves the following mathematical optimization problem to obtain the optimal probability distribution and constructs the optimal inference mechanism h. Then, by applying this mechanism to the input z, the adversary can estimate the user's true location.

$$\begin{aligned} \underset{h}{\text{minimize}} \quad & AE(\pi_a, M, h, d_q) \\ \text{subject to} \quad & \sum_{\hat{x}} \Pr[h(z) = \hat{x}] = 1, \ \forall z, \\ & \Pr[h(z) = \hat{x}] \geq 0, \ \forall z, \hat{x} \end{aligned}$$

For example, if an adversary knows a road network, the domain of his prior π_a consists of locations on that road network. In this setting, the problem is a linear programming problem because $\Pr[h(z) = \hat{x}]$ represents a variable and the other terms are constant; thus, the objective function and the constraints are linear. We solve this problem using CBC (coin-or branch and cut)† solver from the Python PuLP library.

The utility loss caused by mechanism M, called quality loss (Q^{loss}), is defined as follows:

$$Q^{\text{loss}}(\pi_u, M, d_q) = \sum_{x,x'} \pi_u(x) \Pr[M(x) = x'] d_q(x, x')$$

Q^{loss} denotes the expected distance between the user's true location x and the pseudolocation z.

Note that the relationship between Q^{loss} and AE shows the quality of the trade-off between privacy and utility (we refer to Sect. 5.1 for the detail). Our goal is to improve this relationship.

2.2 Differential Privacy

Differential privacy [9] is a mathematical definition of the privacy properties of individuals in a statistical dataset. Differential privacy has become a standard privacy definition and is widely accepted as the foundation of a mechanism that provides strong privacy protection. $d \in \mathcal{D}$ denotes a record belonging to an individual and dataset X is a set of n records. When *neighboring datasets* are defined as two datasets which differ by only a single record, then ε-differential privacy is defined as follows.

definition 1 (ε-differential privacy). *Mechanism $M : \mathcal{D} \to \mathcal{S}$ satisfies ε-differential privacy iff $\forall z \in \mathcal{S}, X, X' \in \mathcal{D}$ such that X and X' are neighboring*

$$\Pr[M(X) = z] \leq e^{\varepsilon} \Pr[M(X') = z].$$

ε-differential privacy guarantees that the outputs of mechanism M are similar when the inputs are neighboring. In other words, from the output of algorithm M, it is difficult to infer what a single record is due to the definition of the neighboring datasets. In this study, we apply differential privacy to a setting of a location on a road network.

2.3 Geo-Indistinguishability

Here, we describe the definition of geo-indistinguishability (GeoI) [2]. Let $\mathcal{X}$ be a set of locations. Intuitively, a mechanism M that achieves GeoI guarantees that $M(x)$ and $M(x')$ are similar to a certain degree for any two locations $x, x' \in \mathcal{X}$. This means that even if an adversary obtains an output from this mechanism, a true location will be indistinguishable from other locations to a certain degree. When $\mathcal{X} \subseteq \mathbb{R}^2$, ε-GeoI is defined as follows [2].

definition 2 (ε-geo-indistinguishability [2]). *Let $\mathcal{Z}$ be a set of query outputs. A mechanism $M : \mathcal{X} \to \mathcal{Z}$ satisfies ε-GeoI iff $\forall x, x' \in \mathcal{X}, z \in \mathcal{Z}$:*

$$\Pr[M(x) = z] \leq e^{\varepsilon d_e(x,x')} \Pr[M(x') = z].$$

where d_e is the Euclidean distance.

2.3.1 Mechanism Satisfying ε-GeoI

The authors of [2] introduced a mechanism called the planar Laplace mechanism (PLM) to achieve ε-GeoI. The probability distribution generated by PLM is called the planar Laplace distribution and—as its name suggests—is derived from a two-dimensional version of the Laplace distribution as follows:

$$\Pr[\mathrm{PLM}_{\varepsilon}(x) = z] = \frac{\varepsilon^2}{2\pi} e^{-\varepsilon d_e(x,z)},$$

where $x, z \in \mathcal{X}$.

2.4 Problem Statement

We consider a perturbation mechanism to improve the tradeoff between utility and privacy by taking advantage of road networks. We assume that the LBSs work on road networks (e.g., UBER), that users are located on road networks, and that LBS providers expect to receive a location on a road network.

We model a road network as an undirected weighted graph $G = (V, E)$ and locations on the road network as the vertices V that are on the Euclidean plane $\mathbb{R}^2$. Each edge in E represents a road segment and the weight of the edge is the length of the road segment. Then, the distance is the shortest path length d_s between two nodes. Here, the following inequality holds for any two vertices on $v, v' \in \mathcal{V}$.

$$d_e(v, v') \leq d_s(v, v'), \tag{1}$$

where d_e is the Euclidean distance.

We assume that a user is located at a location on a road network $v \in V$, sends the location once to receive service from an untrusted LBS, and that an adversary knows that the user is on the road network. The user needs to protect his/her privacy on his/her own device using a perturbation mechanism $M : V \to \mathcal{W}$ where $\mathcal{W} \subseteq V$. This is the same setting as the setting of the local differential privacy [18].

Goals of this paper are to formally define privacy of locations on road networks and to achieve a better tradeoff between privacy and utility by considering road networks than existing method [2] based on the Euclidean plane.

The main notations used in this paper are summarized in Table 1.

† https://projects.coin-or.org/Cbc

Table 1 Summary of notation.

Symbol	Meaning
u, a	A user and an adversary.
$\mathbb{R}$	Set of real numbers.
$\mathcal{Z}$	Set of outputs.
$G = (V, E)$	Weighted undirected graph that represents a road network.
V	Set of vertices.
E	Set of edges. A weight is the distance on the road segment connecting two vertices.
$\mathcal{W} \subseteq V$	Set of vertices of outputs.
$v, v', \hat{v}$	On a road network, a true vertex, a perturbed vertex and an inferred vertex.
$x, x', \hat{x}$	On the Euclidean plane, a true location, a perturbed location and an inferred location.
$\pi_u(x)$	The probability that user u is at location x.
$\pi_a(x)$	Adversary a's knowledge about user's location that represents the probability of being at location x.
M	A mechanism. Given a location, M outputs a perturbed location.
$d_e(x, x')$	An Euclidean distance between x and x'.
$d_s(v, v')$	The shortest distance between v and v' on a road network.
h	Inference function that represents inference of an adversary.
f	Post-processing function.

3. Geo-Graph-Indistinguishability

In this section, we propose a new definition of location privacy on road networks, called Geo-Graph-Indistinguishability (GeoGI). We first formally define GeoGI. Then, we clarify the relationship between GeoI and GeoGI. In the following subsections, we describe the reason why GeoGI restricts the output range and characteristics that GeoGI inherits from $d_{\mathcal{X}}$-privacy [16].

3.1 Definition

We assume that a graph $G = (V, E)$ representing a road network is given. Given $\varepsilon \in \mathbb{R}^+$, we define ε-geo-graph-indistinguishability as follows.

definition 3. (ε-geo-graph-indistinguishability) *Mechanism $M : V \to \mathcal{Z}$ satisfies ε-GeoGI iff $\forall v, v' \in V, z \in \mathcal{Z}$,*

$$\Pr[M(v) = z] \leq \mathrm{e}^{\varepsilon d_s(v,v')} \Pr[M(v') = z],$$

where d_s is the shortest path length between two vertices on G.

Intuitively, ε-GeoGI constrains any two outputs of a mechanism to be similar when the two inputs are similar, that is, they will represent close vertices. In other words, two distributions of two outputs are guaranteed to be similar. The degree of similarity of two probability distributions is $\varepsilon d_s(v, v')$. From this property, an adversary who obtains an output of the mechanism cannot distinguish the true input v from other vertices v' according to the value of $\varepsilon d_s(v, v')$. In particular, a vertex close to the true vertex cannot be distinguished.

Also, this definition implies that GeoGI is an instance of $d_{\mathcal{X}}$-privacy [16] proposed by Chatzikokolakis et al. as are GeoI and differential privacy. Chatzikokolakis et al. showed that an instance of $d_{\mathcal{X}}$-privacy guaranteed strong privacy property as shown in Appendix A.

Remark: We empirically found that $\mathcal{Z}$ should be V from perspective of utility-privacy trade-off. That is, the output range should be the same as the input space (i.e., locations on the road network). This is because an adversary can post-process the perturbed location outside the road network to infer the true information using the road network which is public information.

3.2 Relationship between GeoI and GeoGI

GeoI [2] defines location privacy on the Euclidean plane (see Sect. 2.3 for details). Here, we explain the relationship between GeoI and GeoGI. From Inequality (1),

$$\begin{aligned}\sup_{z \in \mathcal{Z}} \left| \log \frac{\Pr[M(v) = z]}{\Pr[M(v') = z]} \right| &\leq \varepsilon d_e(v, v') \\ &\leq \varepsilon d_s(v, v').\end{aligned}$$

Therefore, we can derive the following lemma.

theorem 1. *If a mechanism M satisfies ε-GeoI, M satisfies ε-GeoGI.*

We note that the reverse is not always true. That is, GeoGI is a relaxed version of GeoI through the use of the metric d_s, allowing for us to create a mechanism that outputs a useful location. We refer to Sect. 4.3 for details.

For example, the planar Laplace mechanism (PLM) (Sect. 2.3.1) satisfies ε-GeoI. Because Outputs of PLM consist locations other than locations on a road network, it may cause empirical privacy leaks. From Theorem 1 and the post-processing theorem (Lemma 1 in Appendix A.1), $f \circ$ PLM satisfies ε-GeoGI and prevents this privacy leaks if f is a mapping function to a vertex of a graph. For utility, we can use a mapping function that maps to the nearest vertex; we call this mechanism the Planar Laplace Mechanism on a Graph (PLMG).

4. A Mechanism to Achieve Geo-Graph-Indistinguishability

Here, we assume that a graph $G = (V, E)$, which represents a road network, is given, and we propose a mechanism that satisfies GeoGI, which we call the Graph-Exponential Mechanism (GEM). Second, we explain the implementation of GEM. Third, we describe an advantage and an issue of GEM caused by not satisfying GeoI.

4.1 Graph-Exponential Mechanism

PLMG (Sect. 3.2) satisfies GeoGI, but PLMG does not take advantage of the structures of road networks to output useful locations. Here, we propose a mechanism that considers the structure of road networks so that the mechanism can output more useful locations. Given a parameter $\varepsilon \in \mathbb{R}^+$ and a set of outputs $\mathcal{W} \subseteq V$, GEM_ε is defined as follows.

definition 4. GEM_ε *takes* $v \in V$ *as an input and outputs* $z \in \mathcal{W}$ *with the following probability.*

$$\Pr[\mathrm{GEM}_\varepsilon(v) = z] = \alpha(v)\mathrm{e}^{-\frac{\varepsilon}{2}d_s(v,z)}, \tag{2}$$

where $\alpha(v) = (\sum_{z\in\mathcal{W}} \mathrm{e}^{-\frac{\varepsilon}{2}d_s(v,z)})^{-1}$.

This mechanism employs the idea of an exponential mechanism [22] that is one of the general mechanisms for differential privacy. Because this mechanism capitalizes on the road network structure by using the metric d_s, it can achieve higher utility for LBSs over road networks than can PLMG as shown in Sect. 6.

theorem 2. *GEM_ε satisfies ε-GeoGI.*

We refer readers to Appendix C for the proof.

4.2 Computational Complexity of GEM

Since we assume that LBS providers are untrusted and there is no trusted server, a user needs to create the distribution and sample the perturbed location according to the distribution locally. Here, we explore a method to accomplish this and the issues that can be caused by the number of vertices.

GEM consists of three phases: (i) obtain the shortest path lengths to all vertices from the user's location. (ii) compute the distribution according to Eq. (2). (iii) sample a point from the distribution. We show the pseudocode of GEM in Algorithm 1.

Algorithm 1 Graph-exponential mechanism.

```
Input: Privacy parameter $\varepsilon$, true location $v$, graph $G = (V, E)$, output range
  $\mathcal{W} \subseteq V$.
Output: Perturbed location $w$.
  (i) $d_s(v, \cdot) \Leftarrow Dijkstra(G = (V, E), v)$
  (ii) Compute the distribution:
      for $v$ in $\mathcal{W}$ do
          $\Pr[GEM(v) = w] \Leftarrow \alpha(v)\mathrm{e}^{-\varepsilon d_s(v,w)/2}$
  (iii) $w \sim \Pr[GEM(v) = w]$
  return $w$
```

We next analyze the computational complexity of each phase. For phase (i), GEM computes the shortest path lengths to the other nodes from v. The computational complexity of this operation is $O(|E| + |V|\log|V|)$ by using Fibonacci heap, where $|V|$ is the number of nodes and $|E|$ is the number of edges. This level of computational complexity does not cause a problem, but on road networks, a fast algorithm computing the shortest path length has been studied for large numbers of graph vertices; we refer the reader to [1] that may be applied to our algorithm. Phase (ii) has no computational problem because its computational complexity is $O(|V|)$. In phase (iii), when the number of vertices is much larger than we expect, we may not be able to effectively sample the vertices according to the distribution. This problem has also been studied and is known as consistent weighted sampling (CWS); we refer the reader to [21], [34]. We believe that these studies can be applied to our algorithm and can be computed even when the number of vertices is somewhat large.

4.3 Privacy with Respect to Euclidean Distance

As described in Sect. 3.2, PLMG satisfies ε-GeoI and ε-GeoGI, but GEM satisfies only ε-GeoGI. This is because GeoGI is a relaxed definition of GeoI that allows a mechanism to output a more useful perturbed location. Therefore, GEM shows better utility as shown in experiments of Sect. 6. It is worth investigating whether this relaxation weakens the privacy protection guarantees. In short, GeoGI has no privacy protection guarantees with respect to Euclidean distance; thus, if a user is using a mechanism that satisfies GeoGI to location privacy, the adversary may easily be able to distinguish the user's location from other locations even when those other locations are close to the user's location based on Euclidean distance. In what follows, we demonstrate this fact using the notion of true probability (TP). The probability that an adversary can distinguish a user's location is

$$\begin{aligned} &TP(\pi_u, M, h) \\ &= \sum_{v,\hat{v}\in\mathcal{V},o\in\mathcal{W}} \pi_u(v)\Pr[M(v) = o]\Pr[h(o) = v']\delta(v, \hat{v}) \end{aligned}$$

where $\delta(\hat{v}, v)$ is a function that returns 1 if $\hat{v} = v$ holds; otherwise, it returns 0. TP is the expected probability with which an adversary can remap a perturbed location to the true location.

We assume a set of graphs, each of which has only two vertices. The Euclidean distances between the vertices are the same for all the graphs, but weights of the edges between them are different for each graph (Fig. 1). Next, we assume that each prior of a user's location is a uniform distribution on two vertices of this graph, and we compute TP of PLMG and GEM. Figure 2 shows the change in TP when the weight (that is, the shortest path length) changes. Due to the guarantee of the Euclidean distance of GeoI, PLM does not degrade TP even when the shortest path length changes, however, since GeoGI does not have a guarantee of the Euclidean distance, GEM significantly degrades TP, which means that the adversary can discover the user's true location.

A mechanism satisfying ε-GeoGI can achieve better

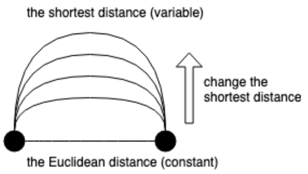

the shortest distance (variable)
change the shortest distance
the Euclidean distance (constant)

Fig. 1 Each graph has a different shortest path length with the same Euclidean distance.

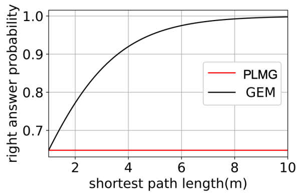


Fig. 2 TP according to GEM and PLMG.

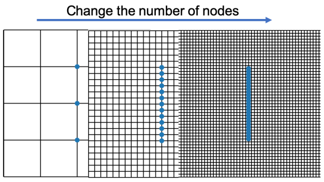


Fig. 3 Points represent graphs nodes, which we use as the input and output of mechanisms. There are edges between neighboring nodes. The side length of each square is 1000 m.

utility than can a mechanism satisfying GeoI by guaranteeing privacy protection in terms of the shortest distance on road networks instead of the Euclidean distance. This idea comes from the interpretation of privacy; in this paper, we assume that privacy can be interpreted as the shortest distance on road networks. Therefore, GeoGI may not be suitable for protecting location privacy when the privacy needs to be interpreted as Euclidean distance, e.g., weather conditions, where a wide range of locations need to be protected.

4.4 Utility Comparison with PLMG

Both GEM$_\varepsilon$ and PLMG$_\varepsilon$ satisfy ε-GeoGI, which means that both guarantee the same indistinguishability. However, outputs of GEM and PLMG are created from different distributions: the continuous distribution with post-processing and the discrete distribution, respectively. Here, we explore the change in utility yielded by their difference; consequently, we use synthetic graphs (Blue points in Fig. 3) whose shortest path lengths and Euclidean distances between two nodes are identical to exclude the difference caused by the variations in the adopted metrics—that is, graphs that have the shape of a straight line on a Euclidean plane. We prepare several graphs by changing the number of nodes while fixing the length of the entire graph. Figure 4 shows the utility loss (i.e., Q^{loss}) of GEM and PLMG with $\varepsilon = 0.01$ for each graph. As shown, the Q^{loss} of GEM increases as the number of nodes increases, while the Q^{loss} of PLMG decreases. This is also the result with other ε values. PLMG is post-processed by mapping to the nearest node, so when few nodes exists near the output of PLM, PLMG cannot output a useful location because the mapping to the location may be distant from the input. Conversely, GEM cannot efficiently output a useful location when there are many nodes because GEM needs to distribute the probabilities to distant nodes. This problem of GEM is solved in the next section. We will also show the effectiveness of GEM compared with PLMG according to the utility in the real-world road networks. We refer to Sect. 6 for details.

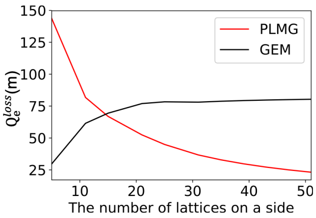


Fig. 4 Utility loss when changing the number of nodes with $\varepsilon = 0.01$.

5. Analyzing the Performance of GEM and Optimizing Range

GEM requires output z to be on a road network but require nothing else for the output range. This means that an optimal output range exists for privacy and utility. In this section, first we apply Q^{loss} and AE to a location setting on road networks. Then, we propose the performance criteria (PC) which represents the tradeoff between the privacy and the utility. Next, we formalize an optimization problem for the PC. Finally, we propose a greedy algorithm to solve the optimization problem in an acceptable amount of time.

5.1 Performance of a Mechanism on a Road Network

While the ε of GeoGI indicates the degree of indistinguishability between a real and perturbed location, it does not indicate the performance of a mechanism w.r.t its utility for some user and empirical privacy against some adversary. Therefore, we introduce the two notions Q_s^{loss} and AE$_s$ by applying Q^{loss} and AE (Sect. 2.1.2) to the setting of road networks. We provide their definitions below.

$$Q_s^{\text{loss}}(\pi_u, M) = Q^{\text{loss}}(\pi_u, M, d_s)$$
$$AE_s(\pi_a, M, h) = AE(\pi_a, M, h, d_s)$$

Intuitively, Q$_s^{\text{loss}}$ is the expected distance on road networks between the true locations and perturbed locations, while AE$_s$ is the expected distance on road networks between the true locations and the locations inferred by an adversary. In the following, we let Q^{loss} and AE denote Q$_s^{\text{loss}}$ and AE$_s$, respectively. We note that, as opposed to ε, AE changes according to the assumed adversary (i.e., the specific attack method and prior distribution). However, because AE increases as Q^{loss} increases (e.g., a mechanism that outputs a distant location will result in high AE but also high Q^{loss}), using only AE as a performance criterion for a mechanism is not appropriate. Then, we define a new criterion to measure the performance of a mechanism against an assumed adversary, which we call the performance criterion (PC).

$$PC = AE/Q^{\text{loss}}$$

Intuitively, against an assumed adversary, PC represents the size of AE with respect to the Q^{loss}. In other words, PC measures the utility/privacy tradeoff. For example, if an adversary with an optimal attack [26] cannot infer the true location at all (i.e., the adversary infers the pseudolocation as the true location), the mechanism can be considered as having the highest performance ($PC = 1$). Conversely, the mechanism performs worst ($PC = 0$) if the adversary can always infer the true location.

5.2 Objective Functions

Here, we propose an objective function to find the optimal output range of GEM with respect to the performance. We assume that the prior distribution of a user is given and adversary knows the prior distribution. An example of this is shown in Sect. 6.2.1. If the prior distribution is not give, we can use uniform distribution for the general user.

Then, we can compute AE and Q^{loss} by assuming an inference function (we refer to Sect. 2.1.1 for detail). We use a posterior distribution $p(\hat{v}|o)$ given the pseudolocation o as the inference function h (that is, $h(o) \sim p(\cdot|o)$). Then, given an output range $\mathcal{W}$, the PC of GEM with the output range $\mathcal{W}$ is formulated as follows:

$$\frac{\sum_{v,\hat{v}\in V,o\in\mathcal{W}} \pi_u(v) \Pr[GEM_{\mathcal{W}}(v) = o] p(\hat{v}|o) d_s(v, \hat{v})}{\sum_{v\in V,o\in\mathcal{W}} \pi_u(v) \Pr[GEM_{\mathcal{W}}(v) = o] d_s(v, o)}$$

where GEM$_{\mathcal{W}}$ denotes GEM with the output range $\mathcal{W}$. Then, the objective function against the adversary can be formulated as follows.

$$\underset{\mathcal{W}\subseteq V}{\text{maximize}} \quad PC_{\mathcal{W}}$$

where PC$_{\mathcal{W}}$ is the PC of GEM$_{\mathcal{W}}$. Here, GEM with the optimized output range is considered to show the best tradeoff against the adversary, but it can fail to be useful (i.e. large Q^{loss}) because Q^{loss} has no constraints; consequently we add the following constraint to Q^{loss}.

$$\begin{aligned} &\underset{\mathcal{W}\subseteq V}{\text{maximize}} && PC_{\mathcal{W}} \\ &\text{subject to} && Q^{\text{loss}}_{\mathcal{W}} \leq \theta \end{aligned}$$

where Q$^{loss}_{\mathcal{W}}$ is the Q^{loss} of GEM$_{\mathcal{W}}$. The optimal GEM shows the best tradeoff in GEM with an output range that shows a better Q^{loss} than θ. We set Q$^{loss}_{\mathcal{W}_0}$ to θ so that the utility does not degrade by the optimization.

5.3 Algorithm to Find an Approximate Solution

Because the number of combinations for the output range is $2^{|V|}$, we cannot compute all combinations to find the optimal solution for the optimized problem in an acceptable amount of time; therefore, we propose a greedy algorithm that instead finds approximate solutions. The pseudocode for this algorithm is listed in Algorithm 2. The constraint function is a function that returns a value indicating whether the constraint holds or does not hold.

Algorithm 2 Finding a local solution.

```
Input: Privacy parameter $\varepsilon$, graph $G = (V, E)$ objective function $f$, constraint function $c$, initial output range $\mathcal{W}_0$.
Output: Output range $\mathcal{W}$.
 1: while True do
 2:     $obj \Leftarrow f(GEM_{\mathcal{W}_o})$
 3:     for $v$ in $V$ do
 4:         $\mathcal{W}' \Leftarrow \mathcal{W} \setminus \{v\}$
 5:         $obj' \Leftarrow f(GEM_{\mathcal{W}'})$
 6:         $cons \Leftarrow c(GEM_{\mathcal{W}'})$
 7:         if $obj' - obj < 0$ and cons then
 8:             $\mathcal{W} \Leftarrow W'$
 9:             $obj \Leftarrow obj'$
10:     if $\mathcal{W}_0 = \mathcal{W}$ then
11:         break
12: return $\mathcal{W}$
```

First, we start with an initial output range $\mathcal{W}_0$, which is given by the next section. Next, we compute a value of the objective function of the output range with one node removed. We remove that node if the objective function improves and the constraint holds. We repeat this procedure until the objective function converges, which has a computational complexity of $O(|\mathcal{W}_0|^2)$ in the worst case when the computational complexity of the objective function is $O(1)$. As a rule of thumb, the main loop (line 2 of Algorithm 2) likely completes in only a small number of iterations. However, the computational complexity of PC is $O(|V|^2|\mathcal{W}_0|)$, so the overall computational $O(|V|^2|\mathcal{W}_0|^3)$. Therefore, when $|\mathcal{W}_0|$ is large, this computational complexity is not acceptable. In the following, we propose a way of providing $\mathcal{W}_0$.

5.3.1 Initialization of $\mathcal{W}$

PC increases when Q^{loss} decreases, so we propose to first optimize output range according to Q^{loss}, which is computed in the small computational complexity. The optimization problem is as follows:

$$\underset{\mathcal{W}\subseteq V}{\text{minimize}} \quad Q^{\text{loss}}_{\mathcal{W}}$$

Q$^{\text{loss}}_{\mathcal{W}\setminus v}$ can be computed using Q$^{\text{loss}}_{\mathcal{W}}$ in the computational complexity of $O(|V|)$. Therefore, we can obtain an approximate solution according to this optimization problem using Algorithm 2 with the initial output range V in the computational complexity of $O(|V|^3)$ in the worst case. As described above, the main loop likely completes in only a small number of iterations, so we can complete this algorithm in the computational complexity of $O(|V|^2)$ in the most case, and this is acceptable even when $|V|$ is somewhat large. We use this output range as the initial output range of Algorithm 2.

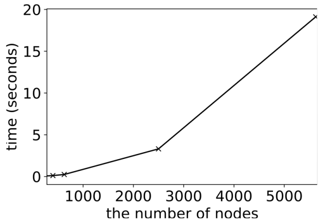


Fig. 5 The relationship between the number of nodes and time required for the optimization.

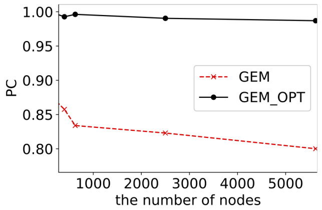


Fig. 6 The relationship between PC and the number of nodes.

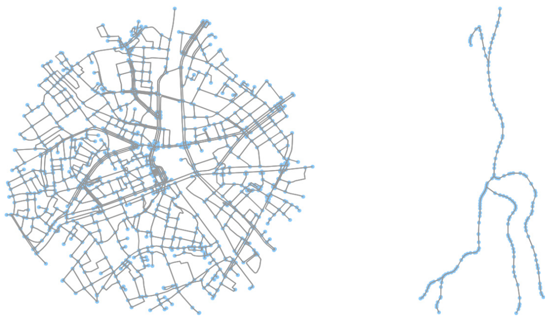


Fig. 7 On the left is a map of Tokyo, while the right shows a map of Akita.

5.4 Optimization Examples

Here, we show examples of the optimization using the synthetic map. First, we explore the relationship between the number of nodes and the time required for the optimization (including initialization of $\mathcal{W}$). We use several lattices with different numbers of nodes (Fig. 3). We use Python 3.7, an Ubuntu 15.10 OS, and 1 core of Intel core i7 6770k CPU with 64 GB of memory as the computational environment. The results are shown in Fig. 5 and Fig. 6, where we can see that even when the number of nodes is large (e.g., > 5000), the algorithm completes under 1 minute and the PC improves by the optimization. This time is acceptable because we can execute the algorithm to calculate future perturbations in advance. As examples of the number of nodes, the two graphs in Fig. 7 whose ranges are 1000 m from the center contain 1,155 and 168 nodes, respectively. Even when a graph is quite large, by separating it into the small graphs such as those in Fig. 7, we can execute the algorithm in an acceptable time. Our implementation for the optimization is publicly available[†].

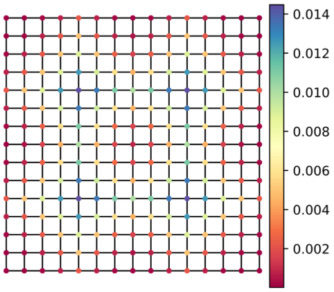


Fig. 8 Synthetic map whose side length is 1500 m. Axis represents the prior probability.

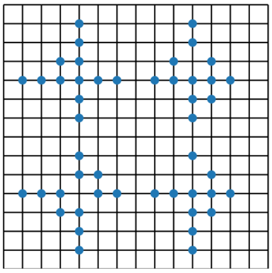


Fig. 9 The example of the solution of the output range.

Next, we executed the algorithm using the synthetic map in Fig. 8 under the assumption of the prior distribution. We assume that there are four places where the prior probability is high, as shown in Fig. 8 and a user who follows this prior probability uses GEM with $\varepsilon = 0.01$ and an adversary has knowledge of the prior distribution. In this case, Q^{loss} is 328 m and PC is 0.9 when we use $\mathcal{W}$ as all nodes. A solution of the Algorithm 2 is as shown in Fig. 9. By restricting output in the place where the prior probability is high, lower utility loss ($Q^{\mathrm{loss}} = 290$ m) and a higher tradeoff ($PC = 0.98$) can be achieved. The adversary infers that the pseudolocation is the true location, which means that the mechanism has effectively perturbed the true location.

Note that this is the expected result; one specific user may suffer utility loss due to the restriction. For example, a user on a corner of Fig. 8 must output distant location because there is no locations in output range near the true location. We can solve this problem by adding constraints, but this incurs the significant additional computational time. This is interesting future direction to guarantee fair utility.

6. Experiments with Real-World Data

Here, we conduct experiments with real-world data. First, we compare GEM with a baseline mechanism, which shows that GEM outperforms the baseline w.r.t. utility-privacy trade-off. Next, we evaluate the optimization method proposed in Sect. 5 with a prior distribution that simulates a real-world use-case.

[†] https://github.com/tkgsn/GG-I

Table 2 The average differences between the Euclidean distances and the shortest path distances.

	Akita	Tokyo	Random
difference (m)	1821.1	666.0	980.8

6.1 Comparison of GEM with the Baseline

Here, we compare GEM with the baseline mechanism described below. We use the output range of GEM obtained by Algorithm 2 with this prior distribution.

(1) The baseline mechanism

We choose OptGeoI [5] and PLMG as the baseline mechanism. OptGeoI has a parameter δ, which balances utility and computational time; we choose $\delta = 1.0$ when the number of possible locations is less than 150, $\delta = 1.2$ when the number is less than 200, and $\delta = 1.4$ when the number is less than 250 (refer to Appendix E for the reason of the choices).

Note that when the number of nodes is more than 200, PLMG is better than OptGeoI with respect to utility because we experimentally found utility of PLMG is better than OptGeoI with $\delta > 1.4$. In this case, we have two options; using PLMG or truncating inputs so that the number of nodes is less than or equal to 200. In this experiment, we adopt both.

(2) Evaluation

We use the following two measures for evaluation.

- PC to evaluate empirical privacy gain
- Q^{loss} to evaluate utility

We compare them with the same ε. Note that ε-OptGeoI and ε-GEM both satisfy ε-GeoGI.

(3) The setting of maps

We use three kinds of maps, Tokyo, Akita (Fig. 7) and Random whose ranges are 2000 m from the center. Here, the experimental results on Random are the average results of the experiments on the randomly pre-chosen 10 maps to explore the more general result. That is, the performance of a user who is located at randomly chosen 10 maps is simulated by the Random map. In Table 2, We show the average difference between the Euclidean distance and the shortest path distance on the road network (i.e., $\frac{1}{\sum_{v,v'} 1}\sum_{v,v'}(d_s(v,v') - d_e(v,v'))$).

We randomly choose n nodes as possible locations to adjust the number. Users are located at each node with the same probability. As described above, the baseline mechanism varies when the number of possible locations exceeds 200. Therefore, we conduct experiments on the following two settings.

- small input: n is less than or equal to 200 (OptGeoI without truncation)
- large input: n is more than 200 (PLMG or OptGeoI with truncation)

6.1.1 The Case of Small Input

Here, we compare GEM with OptGeoI without truncation.

(1) Comparison of PC

Here, we evaluate PC (i.e., AE/Q^{loss}) to fairly compare empirical privacy gain (i.e., AE) of GEM with that of the baseline with the same ε. Figure 10 shows the results when varying n. The number after the name of a map represents the number of possible locations n. We can see GEM achieves the almost optimal value[†] regardless of a chosen map. OptGeoI also achieves near-optimal value for the Tokyo map with small inputs (i.e., 50, 100), but it degrades the performance as the number of nodes increases because δ becomes larger. Also, the results of the Akita map are far from the optimal value. This is because OptGeoI does not consider the road networks; when the difference between the Euclidean distances and the shortest path distances are larger, our mechanism becomes better than OptGeoI.

(2) Comparison of Q^{loss}

Here, we evaluate utility Q^{loss} to compare utility of GEM with that of the baseline. Figure 11 shows the results. We can see that GEM has a higher utility (i.e., smaller Q^{loss} at the same ε) than OptGeoI on the map of Akita. This is due to the large difference between the Euclidean distance and the shortest path distance.

When n is small (e.g., 50, 100), OptGeoI sometimes outperforms GEM on the Tokyo map. This is because the difference between the Euclidean distance and the shortest path distance is small and OptGeoI can adopt $\delta = 1$. This setting is advantageous for OptGeoI, and we can see below that GEM outperforms OptGeoI in the more general setting.

6.1.2 The Case of Large Input

Here, we consider the case where the number of possible locations n is 2000[††]. When n is larger than 200, OptGeoI requires truncation as desribed in Sect. 6.1 (1). Therefore, if a user is located at a location outside the removed inputs, he/she moves the location to the nearest feasible location to use OptGeoI. This breaks ε-GeoI and ε-GeoGI, but here we neglect the affect for simplicity (refer to Appendix D for detail); we focus on the empirical measures (Q^{loss} and PC) when changing the reduced number.

Figure 12 and Fig. 13 show the results. OptGeoI_n represents OptGeoI with truncation to reduce the inputs to n from 2000. OptGeoI_n is better when n is larger even though δ is larger. This is because larger n alleviates the loss due to truncating. PLMG is better than OptGeoI in many cases because PLMG does not require truncation. GEM also does

[†] PC = 1 is the optimal value as described in Sect. 5.1.

[††] Here, we used this number to show the worst case. Even if we increase n from 2000, we confirmed that the results do not change so much. If we decrease n from 2000, we confirmed that the results approaches the results of the case of small input.

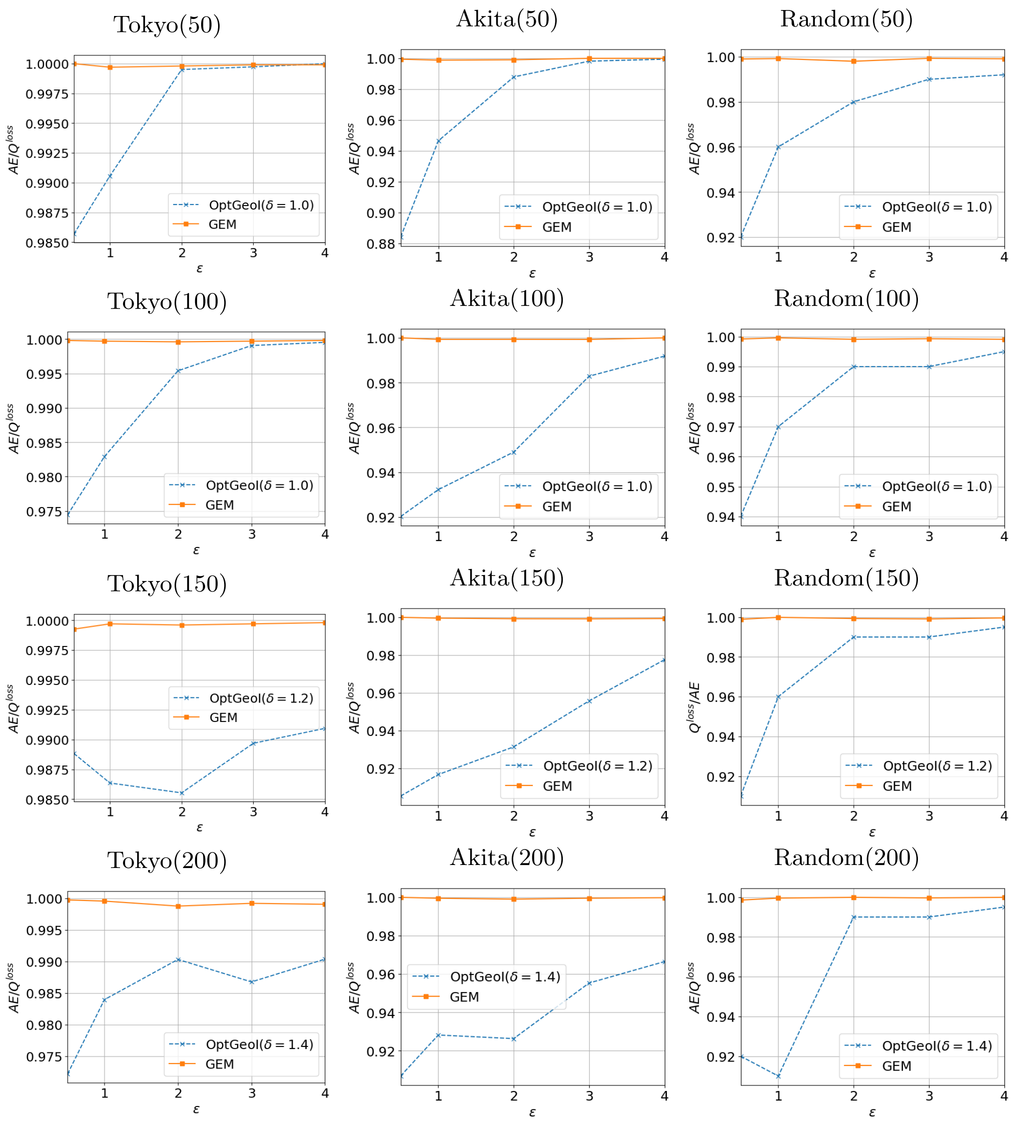


Fig. 10 The comparison of PC on the setting of the small inputs.

not cause the loss. Moreover, GEM considers road networks, which results in much better Q^{loss} and PC than these of the baseline.

6.1.3 Discussion: Effect to GeoI

We showed above that the utility of our mechanism outperformed that of the baseline; this is due to the fact that our mechanism replaces ε-GeoI with ε-GeoGI. Therefore, our mechanism guarantees constant GeoGI regardless a used map but does not guarantee any GeoI, which degrades privacy with respect to Euclidean distance as described in Sect. 4.3. To visualize this effect, we formulate the local GeoI of x and x' as:

$$\varepsilon^{\text{lgeoi}}(x, x') = \frac{1}{d_e(x, x')} \sup_{z \in \mathcal{Z}} \ln \left| \frac{\Pr[M(x) = z]}{\Pr[M(x') = z]} \right|.$$

If M satisfies ε-GeoI, it holds that $\forall x, x'$, $\varepsilon^{\text{lgeoi}}(x, x') \leq \varepsilon$. That is, this is the local guarantee of ε-GeoI between two nodes x and x'. Here, we explore this local value between specific two nodes illustrated on Fig. 14.

We plot the local GeoI of the two specific nodes when we use a 1-GeoGI mechanism and change c with $a = 1000$ on Fig. 15. We can see that GeoI linearly degrades when c increases. For example, when $c = 666$ (i.e., the representative value of the Tokyo map), it degrades to 1.666 from 1. When $c = 1821$ (i.e., the representative value of the Akita map), it degrades 2.821 from 1. This is the key that GEM is much better than OptGeoI in the Akita map. As this, GEM achieves better utility by degrading GeoI and instead keeping GeoGI.

6.2 Evaluation of the Effectiveness of Optimization

Figure 10 shows that the optimization works well in the case of uniform prior distribution. Here, we assume more realis-

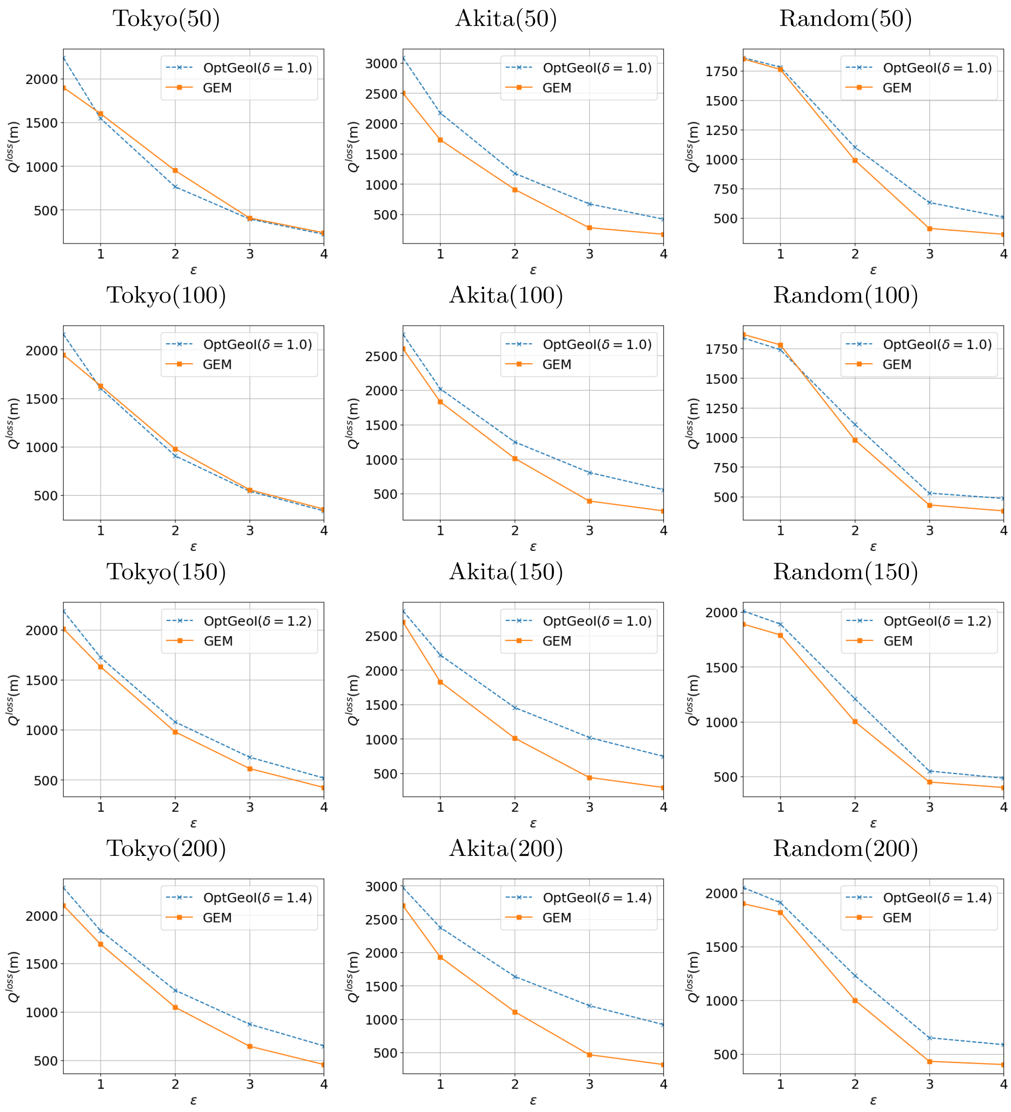


Fig. 11 The comparison of Q^{loss} with the small inputs.

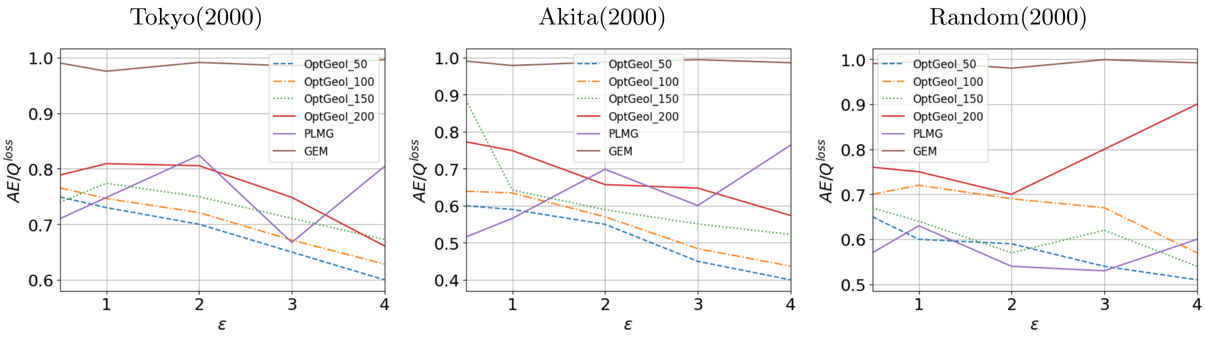


Fig. 12 The comparison of PC on the setting of large inputs.

tic prior distribution and show the effectiveness of the optimization.

6.2.1 Scenario

First, we show that the approximate solution for the proposed objective function effectively improves the tradeoff between utility and privacy. We use the following real-world scenario: a bus rider who uses LBSs. In other words, the user has a higher probability of being located near a bus stop. We create a prior distribution following this scenario by using a real-world dataset, Kyoto Open Data[†], which in-

[†] https://data.city.kyoto.lg.jp/28

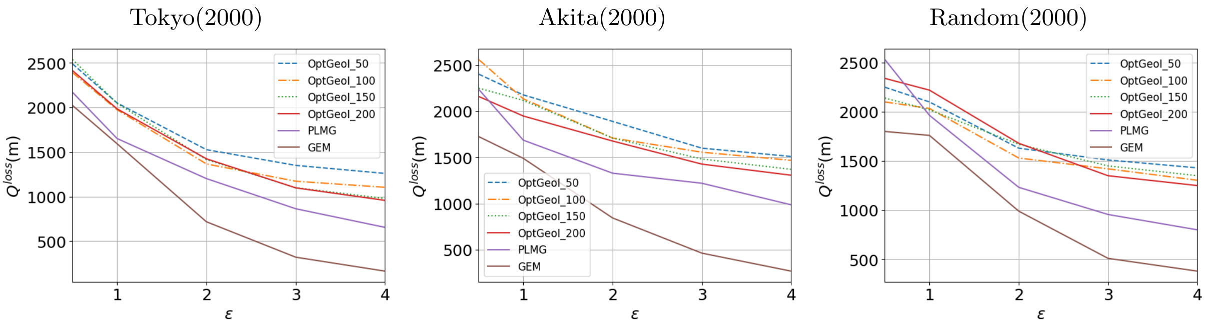


Fig. 13 The comparison of Q^{loss} on the setting of large inputs.

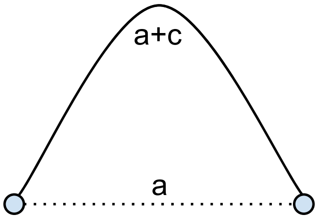


Fig. 14 The case where the difference between the Euclidean distance (a) and the shortest path distance ($a + c$) is cm. We consider the indistinguishability between the two nodes.

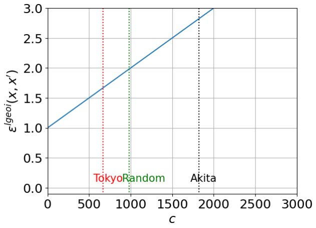


Fig. 15 The local GeoI of the two nodes when a mechanism satisfies 1-GeoGI.

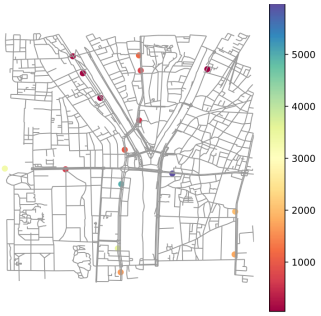


Fig. 16 Each point represents a bus stop, and the y-axis represents the number of people who enter and exit buses at that stop.

cludes the number of people who enter and exit buses at each bus stop per day. Figure 16 shows the data, and Fig. 17 represents the prior distribution made by distributing node probability based on the shortest distance from that node to a bus stop and the number of people who enter and exit buses at that bus stop. We assume that a user who follows this prior distribution uses an LBS with GEM and that an adversary knows the prior distribution. In this setting, we run Algorithm 2 and obtain an approximate solution. Figure 18 shows the example of an approximate solution. We can see that the nodes around the place with higher prior probability remain.

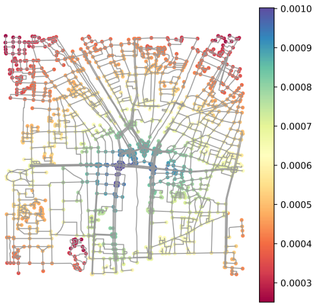


Fig. 17 Prior distribution created from Kyoto Open Data.

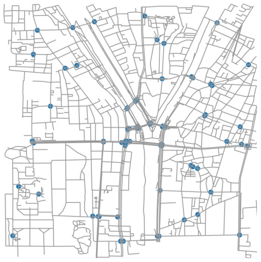


Fig. 18 The solution for the objective function against the adversary with $\varepsilon = 0.01$.

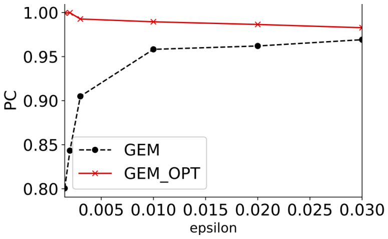


Fig. 19 PC with respect to ε.

6.2.2 Evaluation of Optimized Range

First, we evaluate the PC of GEM with an optimized output range under the same ε as shown in Fig. 19. The result shows that a user can effectively perturb their true location for any realistic value of ε by using the optimized range. When the value of ε is small, the distribution of GEM has a gentle spread. In this case, the output of the mechanism does not contain useful information; thus, the adversary must use his/her prior knowledge, which results in a worse PC in the case of the baseline. However, as these results show, by optimizing the output range according to the prior knowledge of the adversary, we can prevent this type of privacy leak.

7. Related Works

7.1 Cloaking

Cloaking methods [8] obscure a true location by outputting an area instead of the true location. These methods are based on k-anonymity [10] which guarantees that at least k users are in the same area, which prevents an attacker from inferring which user is querying the service provider. This privacy definition is practical, but there are some concerns [20] regarding the rigorousness of the privacy guarantee because k-anonymity does not guarantee privacy against an adversary with some knowledge. If the adversary has peripheral knowledge regarding a user's location, such as range of the user's location, the obscured location can violate privacy. By considering the side knowledge of an adversary [36], the privacy against that particular adversary can be guaranteed, but generally, protecting privacy against one type of adversary is insufficient. Additionally, introducing a cloaking method incurs additional costs for the service provider because the user sends an area rather than a location.

7.2 Anonymization

Anonymization methods [11] separate a user's identifier from that user's location by assigning a pseudonym. Because tracking a single user pseudonym can leak privacy, the user must change the pseudonym periodically. Beresford et al. [3] proposed a way to change pseudonyms using a place called mix zones. However, anonymization does not guarantee privacy because an adversary can sometimes identify a user by linking other information. One of the latest anonymization methods uses Blockchain [14], [24]. The method can guarantee k-anonymity with the guarantee of impossibility of alteration. The other latest method uses the social network services such as FaceBook to guarantee k-anonymity [12].

7.3 Location Privacy on Road Networks

To the best of our knowledge, this is the first study to propose a perturbation method with the differential privacy approach over road networks. However, several studies explored location privacy on road networks.

Tyagi et al. [31] studied location privacy over road networks for VANET users and showed that no comprehensive privacy-preserving techniques or frameworks cover all privacy requirements or issues while still maintaining a desired location privacy level.

Wang et al. [32] and Wen et al. [33] proposed a method of privacy protection for users who wish to receive location-based services while traveling over road networks. The authors used k-anonymity as the protection method and took advantage of the road network constraints.

Tan et al. [28] proposed to use a road network structure in Private Information Retrieval (PIR). Their method improves the consumption of communication costs by considering the road network structure.

Qiu et al. [42] proposed the optimization problem for vehicle-based spatial crowdsourcing and its solution based on GeoI. Bi et al. [4] proposed to use a Voronoi diagram on a road network in a mechanism satisfying local differential privacy. However, the method requires large noise due to the constraint of local differential privacy. A series of key features distinguish our solution from these studies: a) we use the differential privacy approach; consequently, our solution guarantees privacy protection against any attacker to some extent and b) we assume that no trusted server and additional computational power exists. We highlight these two points as advantages of our proposed method.

7.4 State-of-the-Art Privacy Models

Cao et al. [39] proposed the generalized version of GeoI using a policy graph to enable us to customize the privacy of GeoI. They demonstrate the flexibility of privacy to achieve better utility in the COVID-19 case [40]. Achieving this flexibility in GeoGI is interesting direction.

Since GeoI [2] was published, many related applications have been proposed. To et al. [29] developed an online framework for a privacy-preserving spatial crowdsourcing service using GeoI. Tong et al. [30] proposed a framework for a privacy-preserving ridesharing service based on GeoI and the differential privacy approach. It may be possible to improve these applications by using GeoGI instead of GeoI. Additionally, Bordenabe et al. [5] proposed an optimized mechanism that satisfied GeoI, and it may be possible to apply this method to GEM.

According to [2], using a mechanism satisfying GeoI multiple times causes privacy degradation due to correlations in the data; this same scenario also applies to GeoGI. This issue remains a difficult and intensely investigated problem in the field of differential privacy. Two kinds of approaches have been applied in attempts to solve this problem. The first is to develop a mechanism for multiple perturbations that satisfies existing notions, such as differential privacy and GeoI [15], [17]. Kairouz et al. [17] studied the composition theorem and proposed a mechanism that upgrades the privacy guarantee. Chatzikokolakis et al. [15] proposed a method of controlling privacy using GeoI when the locations are correlated. The second approach is to propose a new privacy notion for correlated data [6], [35]. Xiao et al. [35] proposed δ-location set privacy to protect each location in a trajectory when a moving user sends locations. Cao et al. [6] proposed PriSTE, a framework for protecting spatiotemporal event privacy. We believe that these methods can also be applied to our work.

8. Conclusion and Future Work

In this paper, we proposed a new notion of location privacy on road networks, GeoGI, based on differential privacy. GeoGI provides a guarantee of the indistinguishability of a true location on road networks. We revealed that GeoGI is a relaxed version of GeoI. Our experiments showed that this relaxation allows a mechanism to output more useful locations with the same privacy level for LBSs that function over road networks. By introducing the notions of empirical privacy gain AE and utility loss Q^{loss} in addition to indistinguishability ε, we formalized the objective function and proposed an algorithm to find an approximate solution. We showed that this algorithm has an acceptable execution time and that even an approximate solution results in improved performance.

We represented a road network as a undirected graph; this means that our solution has no directionality even though one-way roads exist, which may degrade its utility. In this paper, the target being protected is a location, but if additional information (such as which hospital the user is in) also needs to be protected, our proposed method does not work well: the hospital could be distinguished. This problem can be solved by introducing another metric space that represents the targets to protect instead of the road network graph. Moreover, we need to consider the fact that multiple perturbations of correlated data, such as trajectory data, may degrade the level of protection even if the mechanism satisfies GeoGI as in the case of GeoI and differential privacy. This topic has been intensely studied, and we believe that the results can be applied to GeoGI.

Acknowledgements

This work is partially supported by JST CREST Grant Number JPMJCR21M2, JST SICORP Grant Number JPMJSC2107, the Japan Society for the Promotion of Science (JSPS) Grant-in-Aid for Scientific Research (S) No. 17H06099, (A) No. 18H04093, (B) No. 22H03595, (C) No. 18K11314, Grant-in-Aid for Challenging Research (Exploratory) No. 21K19767, and Early-Career Scientists No. 19K20269.

References

[1] T. Akiba, Y. Iwata, K. Kawarabayashi, and Y. Kawata, "Fast shortest-path distance queries on road networks by pruned highway labeling," Proc. Sixteenth Workshop on Algorithm Engineering and Experiments (ALENEX), pp.147–154, 2014.

[2] M.E. Andrés, N.E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Geo-indistinguishability: Differential privacy for location-based systems," Proc. 2013 ACM SIGSAC Conference on Computer and Communications Security, pp.901–914, 2013.

[3] A.R. Beresford and F. Stajano, "Location privacy in pervasive computing," IEEE Pervasive Comput., vol.2, no.1, pp.46–55, 2003.

[4] M. Bi, Y. Wang, Z. Cai, and X. Tong, "A privacy-preserving mechanism based on local differential privacy in edge computing," China Communications, vol.17, no.9, pp.50–65, 2020.

[5] N.E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Optimal geo-indistinguishable mechanisms for location privacy," Proc. 2014 ACM SIGSAC Conference on Computer and Communications Security, New York, NY, USA, pp.251–262, 2014.

[6] Y. Cao, Y. Xiao, L. Xiong, and L. Bai, "Priste: From location privacy to spatiotemporal event privacy," arXiv preprint arXiv:1810.09152, 2018.

[7] H.J. Cho and C.W. Chung, "An efficient and scalable approach to CNN queries in a road network," Proc. 31st international conference on Very large data bases, pp.865–876, 2005.

[8] M. Duckham and L. Kulik, "A formal model of obfuscation and negotiation for location privacy," International Conference on Pervasive Comput., vol.3468, pp.152–170, Springer, 2005.

[9] C. Dwork, "Differential privacy," Encyclopedia of Cryptography and Security, pp.338–340, 2011.

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

[11] B. Gedik and L. Liu, "Location privacy in mobile systems: A personalized anonymization model," 25th IEEE International Conference on Distributed Computing Systems (ICDCS'05), pp.620–629, IEEE, 2005.

[12] M. Han, L. Li, Y. Xie, J. Wang, Z. Duan, J. Li, and M. Yan, "Cognitive approach for location privacy protection," IEEE Access, vol.6, pp.13466–13477, 2018.

[13] A.-A. Hossain, A. Hossain, H.-K. Yoo, and J.-W. Chang, "H-star: Hilbert-order based star network expansion cloaking algorithm in road networks," 2011 14th IEEE International Conference on Computational Science and Engineering, pp.81–88, IEEE, 2011.

[14] Y. Ji, J. Zhang, J. Ma, C. Yang, and X. Yao, "Bmpls: blockchain-based multi-level privacy-preserving location sharing scheme for telecare medical information systems," Journal of medical systems, vol.42, no.8, pp.1–13, 2018.

[15] K. Chatzikokolakis, C. Palamidessi, and M. Stronati, "A predictive differentially-private mechanism for mobility traces," Privacy Enhancing Technologies, vol.8555, pp.21–41, 2014.

[16] 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, 2013.

[17] P. Kairouz, S. Oh, and P. Viswanath, "The composition theorem for differential privacy," IEEE Trans. Inf. Theory, vol.63, no.6, pp.4037–4049, 2017.

[18] S.P. Kasiviswanathan, H.K. Lee, K. Nissim, S. Raskhodnikova, and A. Smith, "What can we learn privately?," SIAM Journal on Computing, vol.40, no.3, pp.793–826, 2011.

[19] M. Kolahdouzan and C. Shahabi, "Voronoi-based k nearest neighbor search for spatial network databases," Proc. Thirtieth international conference on Very large data bases, vol.30, pp.840–851, 2004.

[20] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," 22nd International Conference on Data Engineering (ICDE'06), pp.24–24, IEEE, 2006.

[21] M. Manasse, F. McSherry, and K. Talwar, "Consistent weighted sampling," Unpublished technical report, June 2010.

[22] F. McSherry and K. Talwar, "Mechanism design via differential privacy," 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS), pp.94–103, Oct. 2007.

[23] D. Papadias, J. Zhang, N. Mamoulis, and Y. Tao, "Query processing in spatial network databases," Proc. 29th international conference on Very large data bases, pp.802–813, 2003.

[24] Y. Qiu, Y. Liu, X. Li, and J. Chen, "A novel location privacy-preserving approach based on blockchain," Sensors, vol.20, no.12, 3519,

2020.

[25] R. Shokri, G. Theodorakopoulos, J.-Y.L. Boudec, and J.-P. Hubaux, "Quantifying location privacy," Proc. IEEE symposium on security and privacy, pp.247–262, 2011.

[26] R. Shokri, G. Theodorakopoulos, C. Troncoso, J.-P. Hubaux, and J.-Y.L. Boudec, "Protecting location privacy: optimal strategy against localization attacks," Proc. 2012 ACM conference on Computer and communications security, pp.617–627, 2012.

[27] Z. Shun, D. Benfei, C. Zhili, and Z. Hong, "On the differential privacy of dynamic location obfuscation with personalized error bounds," arXiv preprint arXiv:2101.12602, 2021.

[28] Z. Tan, C. Wang, C. Yan, M. Zhou, and C. Jiang, "Protecting privacy of location-based services in road networks," IEEE Trans. Intell. Transp. Syst., vol.22, no.10, pp.6435–6448, 2021.

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

[30] W. Tong, J. Hua, and S. Zhong, "A jointly differentially private scheduling protocol for ridesharing services," IEEE Trans. Inf. Forensics Security, vol.12, no.10, pp.2444–2456, 2017.

[31] A.K. Tyagi and N. Sreenath, "Location privacy preserving techniques for location based services over road networks," Proc. International Conference on Communications and Signal Processing (ICCSP), pp.1319–1326, April 2015.

[32] T. Wang and L. Liu, "Privacy-aware mobile services over road networks," Proc. VLDB Endowment, vol.2, no.1, pp.1042–1053, 2009.

[33] J. Wen and Z. Li, "A method of location privacy protection in road network environment," 2018 International Conference on Smart Materials, Intelligent Manufacturing and Automation (SMIMA), vol.173, 03048.

[34] W. Wu, B. Li, L. Chen, C. Zhang, and P.S. Yu, "Improved Consistent Weighted Sampling Revisited," arXiv:1706.01172 [cs], June 2017.

[35] Y. Xiao and L. Xiong, "Protecting locations with differential privacy under temporal correlations," Proc. 22nd ACM SIGSAC Conference on Computer and Communications Security - CCS '15, pp.1298–1309, 2015.

[36] M. Xue, P. Kalnis, and H.K. Pung, "Location diversity: Enhanced privacy protection in location based services," International Symposium on Location-and Context-Awareness, vol.5561, pp.70–87, Springer, 2009.

[37] L. Yu, L. Liu, and C. Pu, "Dynamic differential location privacy with personalized error bounds," Proc. Netw. Distrib. Syst. Security Symp. (NDSS), 2017.

[38] S. Takagi, Y. Cao, Y. Asano, and M. Yoshikawa, "Geo-graph-indistinguishability: Protecting location privacy for LBS over road networks," IFIP Annual Conference on Data and Applications Security and Privacy, vol.11559, pp.143–163, Springer, Cham, 2019.

[39] Y. Cao, Y. Xiao, S. Takagi, L. Xiong, M. Yoshikawa, Y. Shen, J. Liu, H. Jin, and X. Xu, "PGLP: Customizable and Rigorous Location Privacy through Policy Graph," European Symposium on Research in Computer Security, vol.12308, pp.655–676, Springer, Cham, 2020.

[40] Y. Cao, S. Takagi, Y. Xiao, L. Xiong, and M. Yoshikawa, "PANDA: policy-aware location privacy for epidemic surveillance," Proc. VLDB Endowment, vol.13, no.12, pp.3001–3004, 2020. DOI: 10.14778/3415478.3415529

[41] S. Takagi, Y. Cao, Y. Asano, and M. Yoshikawa, "POSTER: Protecting Location Privacy on Road Networks," Proc. 15th ACM Asia Conference on Computer and Communications Security, pp.913–915, 2020.

[42] C. Qiu, A. Squicciarini, C. Pang, N. Wang, and B. Wu, "Location privacy protection in vehicle-based spatial crowdsourcing via geo-indistinguishability," IEEE Trans. Mobile Comput., vol.21, no.7, pp.2436–2450, 2022.

Appendix A: Characteristics of $d_{\mathcal{X}}$-Privacy

GeoGI is an instance of $d_{\mathcal{X}}$-privacy [16], which is a generalization of differential privacy with the following two characteristics that show strong privacy protection.

A.1 Post-Processing Theorem

Inherited from DP, $d_{\mathcal{X}}$-privacy also satisfies post-processing theorem.

lemma 1 (Post-processing theorem of $d_{\mathcal{X}}$-privacy.). *If a mechanism $M : \mathcal{X} \to \mathcal{Z}$ satisfies ε-$d_{\mathcal{X}}$-privacy, a post-processed mechanism $f \circ M$ also satisfies ε-$d_{\mathcal{X}}$-privacy for any function $f : \mathcal{Z} \to \mathcal{Z}'$.*

Proof. Given function $f : \mathcal{Z} \to \mathcal{Z}'$, the following inequality holds for any two locations $x, x' \in \mathcal{X}$ and $z \in \mathcal{Z}'$. We let T denote $\{z \in \mathcal{Z} : f(z) \in S\}$; then, we have:

$$\begin{aligned}\Pr[f(M(x)) \in S] &= \Pr[M(x) \in T]\\ &\leq \mathrm{e}^{\varepsilon d_{\mathcal{X}}(x,x')} \Pr[M(x') \in T]\\ &= \mathrm{e}^{\varepsilon d_{\mathcal{X}}(x,x')} \Pr[f(M(x')) \in S]\end{aligned}$$

This means that:

$$\log \left| \frac{\Pr[f(M(x)) \in S]}{\Pr[f(M(x')) \in S]} \right| \leq \varepsilon d_{\mathcal{X}}$$

Q.E.D. □

A.2 Hiding Function

The first characteristic uses the concept of a hiding function $\phi : V \to V$, which hide a secret location by mapping to the other location. For any hiding function and a secret location $v \in V$, when an attacker who has a prior distribution that includes information about the user's location obtains each output $o = M(v)$ and $o' = M(\phi(v))$ of a mechanism that satisfies ε-GeoGI, the following inequality holds for each posterior distribution:

$$\left| \log \frac{\Pr[v|o]}{\Pr[v|o']} \right| \leq 2\varepsilon \sup_{v \in V} d_s(v, \phi(v))$$

This inequality guarantees that the adversary's conclusions are the same (up to $2\varepsilon \sup_{v\in V} d_s(v, \phi(v))$) regardless of whether ϕ has been applied to the secret location.

A.2.1 Informed Attacker

The other characteristic can be shown by the ratio of a prior distribution and posterior distribution, which is derived by obtaining an output of the mechanism. By measuring this value, we can determine how much the adversary has learned about the secret. We assume that an adversary (informed attacker) knows that the secret location is in $N \subseteq V$. When the adversary obtains an output of the mechanism, the following inequality holds for the ratio of his prior distribution $\pi_{|N}(v) = \pi(v|N)$ and its posterior distribution $p_{|N}(v|o) = p(v|o, N)$:

$$\log \frac{\pi_{|N}(v)}{p_{|N}(v|o)} \leq \varepsilon \max_{v,v' \in N} d_s(v, v')$$

Intuitively, this means that the more the adversary knows about the actual location, the less he will be able to learn about the location from an output of the mechanism.

Appendix B: Output Range from a Privacy Perspective

There are two reasons why output range should be vertices of the graph. First, LBSs that operate over road networks expect to receive a location on a road network as described in Sect. 2.4, so if a user sends a location outside the road networks, the utility decreases.

Second, because road networks are public information, outputting a location outside the road network may cause empirical privacy leaks. We empirically show that an adversary who knows the road network can perform a more accurate attack than can one who does not know the road network; a post-processed mechanism protects privacy from this type of attack. To show this, we evaluate the empirical privacy gain AE of two kinds of mechanisms PLM and PLMG against the two kinds of adversaries at the same utility Q^{loss}.

For simplicity, we use a simple synthetic map illustrated in Fig. A· 1. This map consists of 1,600 squares each of which has a side length of 100 m; that is, the area dimensions are 4000 m $*$ 4000 m, and each lattice point has a coordinate. The centerline represents a road where a user is able to be located, and the other areas represent locations where a user must not be, such as the sea. In this map, we evaluate the empirical privacy gain AE of the two mechanisms against two kinds of adversaries with the same utility loss Q^{loss}. We use Euclidean distance as the metric of AE and Q^{loss}, denoted by AE$_e$ and Q$_e^{loss}$, respectively.

Figure A· 2 shows the results. PLM represents AE against an adversary who knows the road network, while PLM* represents AE against an adversary who dose not know the road network. In this figure, if AE is close to Q^{loss}, the mechanism strongly protects privacy (see Sect. 2.1.2 for detail) against the adversary. Comparing PLM with PLM*, AE of PLM is smaller than PLM*, which means that the adversary can more accurately infer the true location by considering the road network (i.e., weaker privacy protection) even when users are suffering the same utility loss. AE of PLMG is closer to Q^{loss} than that of PLM and almost identical to AE of PLM*, which means stronger privacy protection of PLMG than PLM against the same adversary who knows the road network. By restricting the output to locations on the road network, the adversary cannot improve the inference of the true location because no additional information exists. In other words, post-processing to a location on road networks strengthens the empirical privacy level against an adversary who knows the road network.

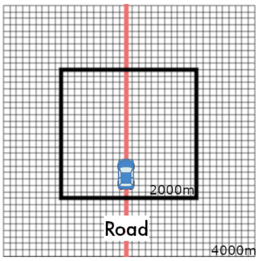


Fig. A· 1 A synthetic map. The red line represents a road, and a user is located inside the black frame.

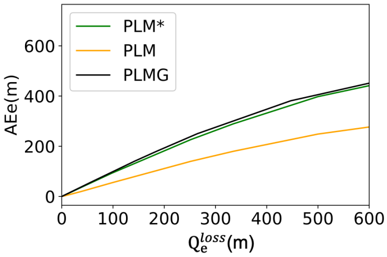


Fig. A· 2 AE of each mechanism with respect to Q^{loss} with the Euclidean distance, that is AE$_e$ and Q$_e^{loss}$. PLM* represents AE of PLMG against an adversary who does not know the road network.

Appendix C: Omitted Proof

theorem 3. *Given a graph $G = (V, E)$, GEM_ε satisfies ε-GeoGI.*

Proof. We prove that the following inequality holds for any two locations on road networks $v, v' \in V$ and $S \subseteq \mathcal{W}$:

$$\frac{\Pr[GEM(v) \in S]}{\Pr[GEM(v') \in S]} \leq \exp(\varepsilon d_s(v, v'))$$

The following inequality holds for any $S \subseteq \mathcal{W}$ and $v, v' \in V$ from the triangle inequality:

$$d_s(v, v') + d_s(v', w) \geq d_s(v, w) \text{ and}$$
$$d_s(v, w) + d_s(v, v') \geq d_s(v', w)$$

Then, the left side of the inequality is transformed as follows:

$$\frac{\Pr[GEM(v) \in S]}{\Pr[GEM(v') \in S]} = \frac{\alpha(v)}{\alpha(v')} \frac{\sum_{w \in S} \exp(-\varepsilon d_s(v, w)/2)}{\sum_{w \in S} \exp(-\varepsilon d_s(v', w)/2)}$$

We show $\frac{\sum_{w \in S} \exp(-\varepsilon d_s(v,w)/2)}{\sum_{w \in S} \exp(-\varepsilon d_s(v',w)/2)} \leq \mathrm{e}^{\varepsilon d_s(v,v')/2}$ and $\frac{\alpha(v)}{\alpha(v')} \leq \mathrm{e}^{\varepsilon d_s(v,v')/2}$, respectively.

$$\begin{aligned}\frac{\sum_{w \in S} \mathrm{e}^{-\varepsilon d_s(v,w)/2}}{\sum_{w \in S} \mathrm{e}^{-\varepsilon d_s(v',w)/2}} &= \frac{\sum_{w \in S} \mathrm{e}^{-\frac{\varepsilon}{2} d_s(v,w)}}{\mathrm{e}^{-\frac{\varepsilon}{2} d_s(v,v')} \sum_{w \in S} \mathrm{e}^{-\frac{\varepsilon}{2}(d_s(v',w) - d_s(v,v'))}} \\ &\leq \mathrm{e}^{\frac{\varepsilon}{2} d_s(v,v')} \frac{\sum_{w \in S} \mathrm{e}^{-\frac{\varepsilon}{2} d_s(v,w)}}{\sum_{w \in S} \mathrm{e}^{-\frac{\varepsilon}{2} d_s(v,w)}} \\ &= \mathrm{e}^{\frac{\varepsilon}{2} d_s(v,v')}\end{aligned}$$

Table A· 1 The privacy guarantees of OptGeoI (ε) and OptGeoI with the truncation (ε'). Here, OptGeoI$_{\varepsilon'}$ and GEM$_{\varepsilon'}$ represent OptGeoI$_{\varepsilon}$ with the truncation and GEM using ε', respectively.

location	ε	ε'	OptGeoI$_{\varepsilon'}$ Q^{loss}	GEM$_{\varepsilon'}$ Q^{loss}
Tokyo	0.5	90.645	4670.9	52.447
	1.0	192.35	2788.0	11.562
	2.0	374.19	2156.1	2.7207
Akita	0.5	26.432	9549.5	84.810
	1.0	86.203	5144.9	32.551
	2.0	123.03	3807.1	13.335
Random	0.5	51.244	5515.6	112.94
	1.0	109.13	3314.0	23.823
	2.0	153.73	2648.2	10.426

The inequality is from the triangle inequality. Since

$$\frac{\alpha(v)}{\alpha(v')} = \frac{\sum_{w\in\mathcal{W}} \exp(-\varepsilon d_s(v',w)/2)}{\sum_{w\in\mathcal{W}} \exp(-\varepsilon d_s(v,w)/2)},$$

in the similar vein we have

$$\begin{aligned}\frac{\sum_{w\in\mathcal{W}} \mathrm{e}^{-\varepsilon d_s(v',w)/2}}{\sum_{w\in\mathcal{W}} \mathrm{e}^{-\varepsilon d_s(v,w)/2}} &= \mathrm{e}^{\frac{\varepsilon}{2}d_s(v,v')} \frac{\sum_{w\in\mathcal{W}} \mathrm{e}^{-\frac{\varepsilon}{2}(d_s(v',w)+d_s(v,v'))}}{\sum_{w\in\mathcal{W}} \mathrm{e}^{-\frac{\varepsilon}{2}d_s(v,w)}}\\ &\leq \mathrm{e}^{\frac{\varepsilon}{2}d_s(v,v')} \frac{\sum_{w\in\mathcal{W}} \mathrm{e}^{-\frac{\varepsilon}{2}d_s(v,w)}}{\sum_{w\in\mathcal{W}} \mathrm{e}^{-\frac{\varepsilon}{2}d_s(v,w)}}\\ &= \mathrm{e}^{\frac{\varepsilon}{2}d_s(v,v')}.\end{aligned}$$

That is,

$$\frac{\alpha(v)}{\alpha(v')} \leq \mathrm{e}^{\frac{\varepsilon}{2}d_s(v,v')}.$$

Therefore,

$$\frac{\Pr[GEM(v)\in S]}{\Pr[GEM(v')\in S]} \leq \mathrm{e}^{\frac{\varepsilon}{2}d_s(v,v')} \cdot \mathrm{e}^{\frac{\varepsilon}{2}d_s(v,v')} = \mathrm{e}^{\varepsilon d_s(v,v')}.$$

Q.E.D. □

Appendix D: Influence of the Small Number of Possible Locations

OptGeoI truncates possible input locations due to the constraint of optimization. This influences GeoGI for users at the removed locations. Here, we experimentally show the influences of OptGeoI with comparing to GEM. We use the extended map of Fig. 7, whose size is 10,000 m from the center to see the influences. Here, we use random 5000 nodes for all possible locations and truncate them to 50 for OptGeoI. Users pre-process to map their location to the nearest possible location that is applicable to OptGeoI.

We compute GeoGI of OptGeoI with truncation according to Definition 2. Table A· 1 shows the impact of the pre-processing where ε' means the privacy level of OptGeoI with truncation. We can see that even if we use ε for OptGeoI, the privacy level (ε') considerably decreases due to the truncation. We note that GEM does not require the truncation, so the privacy level does not decrease. As shown in Table A· 1, if we use GEM with ε', we can achieve much better Q^{loss}.

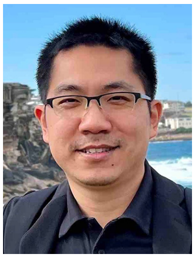
Table A· 2 The computational times given δ and the number of possible locations.

n nodes\\δ	1.0	1.1	1.2	1.3	1.4
50	**1m**	20s	8s	6s	6s
100	**3h**	1h	3m	3m	1m
150	24h+	24h+	**6h**	3h	1h
200	24h+	24h+	24h+	24h+	**8h**
250	24h+	24h+	24h+	24h+	24h+

Appendix E: The Choice of The Baseline Mechanism

Here, we explain the choice of δ of OptGeoI. OptGeoI has a parameter δ, which balances utility and computational time. Bordenabe et al. [5] did not show how to choose the value of δ, so we conducted an experiment to empirically choose the appropriate value of δ. We show the computational time when changing δ and the number of possible locations on the Tokyo map. We set the acceptable computational time as 24 hours for the convenience of experiments, but this computational time is reasonable because it exponentially increases as the number of possible locations increases. δ should be as small as the computational time allows to achieve better utility. Therefore, we choose $\delta = 1.0$ when the number is less than 150, $\delta = 1.2$ when the number is less than 200, and $\delta = 1.4$ when the number is less than 250.

Shun Takagi received the BS degree in information science from University of Kyoto in 2019. Since 2019, he has been a master student at Kyoto university. His research interests include privacy protection technologies, specifically, theoretical analysis of differential privacy and local differential privacy with respect to locations and machine learning.



Yang Cao is an Associate Professor at the Division of Computer Science and Information Technology at Hokkaido University. He earned his Ph.D. from Kyoto University, in 2017. His research interests lie in the intersections between database, security, and machine learning. He has published many papers in these areas including top venues such as SIGMOD, VLDB, ICDE, AAAI, TKDE. Two of his papers were selected as 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.

Yasuhito Asano received the BS, MS, and DS degrees in information science from the University of Tokyo in 1998, 2000, and 2003 respectively. In 2003–2005, he was a research associate in Tohoku University. In 2006–2007, he was an assistant professor in Tokyo Denki University. He joined Kyoto University in 2008. In 2009–2018, he was an associate professor in the Graduate School of Informatics, Kyoto University. Since 2019, he has been a professor at Faculty of Information Networking for Innovation and Design, Toyo University. His research interests include web mining and network algorithms.

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.