Trajectory Data Collection with Local Differential Privacy

Yuemin Zhang*
Harbin Engineering University
zhangyuemin@hrbeu.edu.cn

Qingqing Ye
Hong Kong Polytechnic University
qqing.ye@polyu.edu.hk

Rui Chen†
Harbin Engineering University
ruichen@hrbeu.edu.cn

Haibo Hu
Hong Kong Polytechnic University
haibo.hu@polyu.edu.hk

Qilong Han†
Harbin Engineering University
hanqilong@hrbeu.edu.cn

ABSTRACT

Trajectory data collection is a common task with many applications in our daily lives. Analyzing trajectory data enables service providers to enhance their services, which ultimately benefits users. However, directly collecting trajectory data may give rise to privacy-related issues that cannot be ignored. Local differential privacy (LDP), as the *de facto* privacy protection standard in a decentralized setting, enables users to perturb their trajectories locally and provides a provable privacy guarantee. Existing approaches to private trajectory data collection in a local setting typically use relaxed versions of LDP, which cannot provide a strict privacy guarantee, or require some external knowledge that is impractical to obtain and update in a timely manner. To tackle these problems, we propose a novel trajectory perturbation mechanism that relies solely on an underlying location set and satisfies pure ϵ-LDP to provide a stringent privacy guarantee. In the proposed mechanism, each point's adjacent direction information in the trajectory is used in its perturbation process. Such information serves as an effective clue to connect neighboring points and can be used to restrict the possible region of a perturbed point in order to enhance utility. To the best of our knowledge, our study is the first to use direction information for trajectory perturbation under LDP. Furthermore, based on this mechanism, we present an anchor-based method that adaptively restricts the region of each perturbed trajectory, thereby significantly boosting performance without violating the privacy constraint. Extensive experiments on both real-world and synthetic datasets demonstrate the effectiveness of the proposed mechanisms.

PVLDB Reference Format:
Yuemin Zhang, Qingqing Ye, Rui Chen, Haibo Hu, and Qilong Han. Trajectory Data Collection with Local Differential Privacy. PVLDB, 16(10): 2591 - 2604, 2023.
doi:10.14778/3603581.3603597

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/ymzgithub/ymzgithub-traj_LDP_submission_sup.

*Work partially done at The Hong Kong Polytechnic University.
†Corresponding authors.

Proceedings of the VLDB Endowment, Vol. 16, No. 10 ISSN 2150-8097.
doi:10.14778/3603581.3603597

1 INTRODUCTION

Nowadays mobile applications generate a considerable amount of personal private data, such as locations and browsing histories. The collection and analysis of such data help data collectors improve their services, which benefits both data collectors and users. As a common and important type of personal data, trajectory data have a wide range of applications in data mining tasks. For example, taxi companies could help drivers decide where to wait for passengers by analyzing passengers' trajectory data [49]. However, in the absence of proper privacy protection, direct collection of sensitive trajectory data would give rise to serious privacy concerns. Differential privacy (DP) [17] is regarded as a golden standard that provides a rigorous privacy protection. However, it requires a trusted server to which users must upload their raw data. This requirement may not always be possible in practical settings. In contrast, local differential privacy (LDP) [26] enables distributed users to perturb their data locally and send the sanitized data to an untrusted server, which is more suitable for private trajectory data collection.

However, due to the known low utility of LDP perturbation on trajectory release, most of the existing approaches for private trajectory data collection either use some relaxed versions of LDP or require external knowledge that is difficult to obtain and update in practice. The relaxations of LDP, such as geo-indistinguishability [1], cannot provide the strict privacy guarantee of ϵ-LDP. The only approach that satisfies ϵ-LDP is the NGRAM mechanism [14], which uses additional external knowledge, such as business hours and category information, to boost utility. Such information may not be readily available in practice and is often not updated in a timely manner. Therefore, our goal is to introduce a more practical trajectory perturbation mechanism that satisfies ϵ-LDP while relying solely on the essential location set of trajectories.

It is a non-trivial technical challenge to design such a mechanism with high utility. The location points of a trajectory are usually the elements of a finite point set, e.g., grid centers or points of interest (POIs). Each point has its corresponding tuple of latitude and longitude. To perturb a trajectory under ϵ-LDP, we must ensure that each point in the set has a probability of being included in the output. However, the point domain is normally so large that the perturbed point may be far from the real one, which leads to poor utility. On the other hand, a real-world trajectory places a natural restriction that the distance between any two neighboring points cannot be too large. To resolve these problems, a straightforward solution is to restrict the possible perturbed points for each original point in the trajectory, as used in [14]. However, such a solution

requires not only the trajectory and the distribution of all possible locations, but also public external information, from which we would like to free ourselves.

The key observation of this paper is that a user's trajectory is a time-ordered sequence of locations backed by the user's intentions, i.e., she needs to go to a few target places step by step. When she needs to go to a target place, she typically first determines an approximate direction from the current location to the target location. This direction information is the "clue" that connects every pair of adjacent points in a trajectory, which has not been explored in the existing approaches under LDP. Motivated by this observation, in this paper, we propose a **pivot sampling** perturbation mechanism that enhances privacy-aware trajectory modeling by sanitizing a given target point in a trajectory through a two-stage process. We first identify the adjacent points preceding and following a target point in the trajectory as its pivots. A pivot of the target point is treated as an origin so that the direction between itself and the target point can be obtained. In the second stage, we perturb the target point within the region defined by bi-directional information from the two pivots. It is worth noting that our proposed mechanism is applicable to not only trajectories satisfying the above key observation, but also many other types, as long as the points in a trajectory belong to a finite location point set.

In contrast to pivot sampling leveraging bi-directional information to confine a single point's perturbed result, it is also possible to confine the spatial region of an entire trajectory. According to the first law of geography [37], neighboring points in a trajectory are usually close to each other, and therefore the spatial region of a trajectory is highly likely to be relatively small in comparison with the entire area of the map. Thus, we further introduce an anchor-based method to limit the spatial region of the entire trajectory, and then apply pivot sampling to trajectory perturbation. A simple strategy is to restrict trajectories to different spatial regions with the same fixed size. We theoretically analyze the utility of this strategy under different sizes. Then, to avoid the use of additional hyperparameters (i.e., the region size), we put forward an adaptive strategy to achieve better performance than the mechanism using the best fixed size.

The key contributions of our study are summarized as follows:

- We introduce an original privacy-aware trajectory modeling solution to private trajectory data collection. We model the perturbation of the points in a trajectory as a two-stage process and propose a novel mechanism called pivot sampling, which captures bi-directional information of the points to restrict their regions. We also propose a guideline on choosing the direction granularity. To the best of our knowledge, our study is the first to combine direction information with trajectory perturbation in LDP.
- We also propose an anchor-based pivot sampling mechanism that restricts the spatial regions of trajectories while satisfying ϵ-LDP. We perform a theoretical utility analysis of using a fixed region size, and then propose a strategy to adaptively restrict the spatial region of a trajectory.
- We perform extensive experiments on real-world and synthetic datasets to demonstrate the effectiveness of our mechanism in comparison with existing solutions.

The remainder of this paper is organized as follows. We discuss the related work in Section 2. In Section 3, we provide necessary background information and the problem definition. In Sections 4 and 5, we describe the proposed mechanisms in detail. In Section 6, we present the experimental results. Finally, we conclude our paper in Section 7.

2 RELATED WORK

To prevent privacy leakages, several classic privacy models have been proposed, such as k-anonymity [32, 35], l-diversity [29], and t-closeness [27]. These models make different assumptions about the background knowledge possessed by attackers. In recent years, DP [17] has been considered the golden standard for privacy protection. In contrast with other privacy models, DP does not need to assume background knowledge owned by attackers, and provides a rigorous privacy guarantee for individual user data. It has been widely used in many tasks such as frequent subgraph mining [10, 41], high-dimensional data publishing [9, 11, 33, 50], and sequential data sanitization [7]. In the typical setting of DP, data owners upload their data to a trusted data collector, which then perturbs and shares the querying results. However, it comes with deficiencies in some practical settings—on the one hand, the DP model requires a trusted server to collect the raw data from users; on the other hand, users are unlikely to agree to directly upload their sensitive data to the collector in real life. To avoid these problems, LDP [26] was proposed. Many studies [46, 47] have focused on applying LDP to protect users' local data, and several companies [13] have already developed LDP in ther products, such as Google [19], Apple [2], and Microsoft [15]. In the LDP model, data owners must perturb their sensitive data locally and then send the perturbed version to an untrusted data collector.

DP/LDP generally considers a trade-off between utility and privacy. Several relaxations of DP/LDP have been proposed to obtain better data utility for various data analysis tasks at the cost of weakened privacy protection. For example, $d_{\mathcal{X}}$-privacy [6] assumes that the indistinguishability of two inputs is inversely proportional to the distance between them, whereas in pure LDP the indistinguishability remains the same and does not depend on the distance. Geo-indistinguishability [1] is a variant of $d_{\mathcal{X}}$-privacy in the context of location privacy. Personalized LDP [8] is a variant of LDP that enables different users to use different privacy levels determined by themselves.

Although many studies have investigated location privacy, few studies have examined private trajectory data collection in the local setting. Most studies have focused on trajectory privacy in the centralized setting [7, 21–23, 34, 45]. Chen *et al.* [7] utilize the prefix tree structure to group sequences with identical prefixes into the same branch. He *et al.* [23] develop a mechanism for trajectory data synthesis, which discretizes raw trajectories using hierarchical reference systems to capture individual movements at different speeds. They also propose a post-processing strategy to restore directionality while sampling synthetic trajectories from the noisy model based on the Markovianity of the trajectory. We would like to emphasize that, unlike our idea, their solution does not utilize the adjacent direction information to restrict the perturbation domain of a point. Furthermore, their solution is designed for DP and

cannot be applied to LDP directly. Jiang *et al.* [24] also consider utilizing direction information to publish trajectories under DP. They assume that the start and end points of a trajectory are known and that the distance between each pair of neighboring locations in a trajectory is bounded by a constant value. In contrast, we focus on LDP and aim to remove strong assumptions to make our solution more practical.

Most of the related studies in the local setting have been designed to satisfy the relaxations of LDP. To the best of our knowledge, the n-gram-based method NGRAM [14] is the only solution to private trajectory data collection that satisfies pure LDP. NGRAM makes use of public external knowledge, such as business hours and POI categories, for POI trajectory perturbation. However, it suffers from some limitations in real-world applications where the required public external information is often not readily available or cannot be updated in a timely manner. Therefore, in this paper we are motivated to remove the dependency on such public knowledge for better applicability.

3 PRELIMINARIES AND PROBLEM FORMULATION

3.1 Local Differential Privacy

As a variant of differential privacy (DP) [17], LDP [26] does not rely on a trusted data collector, and thus becomes a practical privacy notion for many applications in the distributed setting, such as frequency and mean estimation [16, 38, 39, 42], heavy hitter identification [4, 31], and preference ranking analysis [44]. A formal definition of LDP is given as follows.

Definition 3.1 (ϵ-LDP). Given a privacy budget $\epsilon > 0$, a randomized mechanism $\mathcal{M} : \mathcal{X} \to \mathcal{Y}$ provides ϵ-LDP if and only if, for any two inputs $x, x' \in \mathcal{X}$ and any possible output $y \in \mathcal{Y}$, the following inequality holds

$$\Pr[\mathcal{M}(x) = y] \leq e^{\epsilon} \times \Pr[\mathcal{M}(x') = y]. \tag{1}$$

LDP enables users to locally perturb their data in order to guarantee plausible deniability, which is controlled by privacy budget ϵ. As with DP, LDP enjoys the desired properties of sequential composition and post-processing [18].

Theorem 3.2. *(Sequential Composition) Let each $\mathcal{M}_i (1 \leq i \leq n)$ denote a mechanism satisfying ϵ_i-LDP. Then the sequential combination of these mechanisms satisfies ϵ-LDP, where $\epsilon = \sum_{i=1}^{n} \epsilon_i$.*

Theorem 3.3. *(Post-Processing) Let $\mathcal{M} : \mathcal{X} \to \mathcal{Y}$ be a mechanism that satisfies ϵ-LDP, and $f : \mathcal{Y} \to \mathcal{Y}'$ be an arbitrary randomized mapping. Then $f \circ \mathcal{M} : \mathcal{X} \to \mathcal{Y}'$ satisfies ϵ-LDP, where $\circ$ denotes the composition of f and $\mathcal{M}$.*

From the sequential composition property, the amount of the privacy budget consumed will increase linearly when we apply the mechanisms sequentially; from the post-processing property, there is no additional privacy cost when perturbed data are used for further processing.

3.2 Perturbation Mechanisms

Some perturbation mechanisms that satisfy ϵ-LDP have been proposed in the literature. In this subsection, we briefly introduce three classic mechanisms adopted in our study: k-ary randomized response (k-RR) [25, 40], exponential mechanism (EM) [30] and square-wave mechanism (SW) [28].

Definition 3.4 (k-RR). Given a privacy budget ϵ, k-RR mechanism $\mathcal{M} : \mathcal{X} \to \mathcal{X}$ satisfies ϵ-LDP, if, for any input $x \in \mathcal{X}$, the output $y \in \mathcal{X}$ is sampled from the following distribution:

$$\Pr[\mathcal{M}(x) = y] = \begin{cases} \frac{e^{\epsilon}}{|\mathcal{X}|-1+e^{\epsilon}} & \text{if } y = x, \\ \frac{1}{|\mathcal{X}|-1+e^{\epsilon}} & \text{otherwise.} \end{cases} \tag{2}$$

By Equation 2, k-RR always reports the original value (i.e., $y = x$) with a larger probability; otherwise, the mechanism returns one of the other values uniformly. It has been proven that k-RR achieves good performance when the domain size k (i.e., $|\mathcal{X}|$) is not overly large (i.e., when $k < 3e^{\epsilon} + 2$) [39].

Definition 3.5 (Exponential Mechanism). The exponential mechanism (EM) $\mathcal{M}$ preserves ϵ-LDP, if, for any input $x \in \mathcal{X}$, the probability of any output $r \in \mathcal{R}$ is

$$\Pr[\mathcal{M}(x) = r] = \frac{\exp(\frac{\epsilon u(x,r)}{2\Delta u})}{\sum_{r' \in \mathcal{R}} \exp(\frac{\epsilon u(x,r')}{2\Delta u})}, \tag{3}$$

where $u(x, r)$ denotes the utility function of x and r, and $\Delta u = \max_{x,x' \in \mathcal{X},\, r \in \mathcal{R}} |u(x, r) - u(x', r)|$ is the sensitivity of the utility score.

The EM is a natural building block of answering queries with arbitrary utility scores, which makes it possible for a perturbed answer to be as close to the truth as possible (controlled by the utility score and privacy budget).

While both k-RR and the EM are used for categorical value perturbation, the Laplace mechanism [17] and Square-Wave Mechanism (SW) are typically designed for numerical values. Due to unbounded noise and high variance in the case of the Laplace mechanism, we alternatively adopt the SW mechanism [28] for numerical value perturbation.

Definition 3.6 (Square-Wave Mechanism). The square-wave mechanism $\mathcal{M}$ satisfies ϵ-LDP, if, for any input $x \in [0, 1]$ and output $y \in [-b, b + 1]$, the output probability of y is:

$$\Pr[\mathcal{M}(x) = y] = \begin{cases} \frac{e^{\epsilon}}{2be^{\epsilon}+1} & \text{if } |x - y| \leq b, \\ \frac{1}{2be^{\epsilon}+1} & \text{otherwise,} \end{cases} \tag{4}$$

where $b = \frac{\epsilon e^{\epsilon} - e^{\epsilon} + 1}{2e^{\epsilon}(e^{\epsilon} - 1 - \epsilon)}$.

The SW mechanism takes a value in $[0, 1]$ as input and returns a bounded output. The output range will have a higher concentration around the input value as the privacy budget increases, which is beneficial for utility enhancement.

3.3 Problem Statement

We consider a trajectory as a time-ordered sequence of points. Each user u in the population possesses a sensitive trajectory $\tau = \{p_1, p_2, \cdots, p_{|\tau|}\}$, where $p_i \in \mathcal{P}$ denotes a location in the form of a two-dimensional (2D) point with latitude and longitude, and $\mathcal{P}$ is a finite point set. We assume that the location point set $\mathcal{P}$ is public and accessible to the data collector and all users. This is a common requirement for location/trajectory privacy preservation [1, 5], and

$\mathcal{P}$ is readily available in practice. Given two points, we use the Haversine distance as the distance measure, which calculates their great-circle distance, namely, the shortest distance over the surface of earth.

For the sake of privacy, each user locally perturbs her trajectory τ into $\hat{\tau} = \{\hat{p}_1, \hat{p}_2, \cdots, \hat{p}_{|\tau|}\}$ while satisfying ϵ-LDP. Following Definition 3.1, we aim to provide *user-level* ϵ-LDP for each user by perturbing *every* point in her trajectory. Hence, the perturbed trajectory ensures that an adversary cannot infer *any* point (i.e., location) she visited in the original trajectory with high confidence.

Upon receiving the perturbed trajectories from all users, some trajectory data analysis tasks (e.g., preservation range queries) can be performed. The goal of our study is to preserve as much trajectory information as possible, while achieving a rigorous ϵ-LDP privacy guarantee.

4 METHODOLOGY

In this section, we first introduce our design rationale for private trajectory data collection with LDP and provide an overview of the workflow in Section 4.1. Then, we present the implementation details in Sections 4.2 and 4.3. Finally, we establish the privacy guarantee for the proposed mechanism in Section 4.4.

4.1 Design Rationale and Overview

In the task of private trajectory data collection, the distance between any two points (i.e., locations) is the most intuitive information that can be used for perturbation, e.g., applying EM [30] to perturb each point [36, 48]. However, as the domain of each point in a trajectory is a large region, the perturbed point is likely to be far from the original point. To address this challenge, a natural idea is to restrict the domain of a perturbed point. A good source of restriction is the direction between two neighboring points—for a given point, once the direction of the next point is determined, the size of the perturbation domain will be reduced significantly. Therefore, we leverage both distance and direction information for trajectory perturbation.

An overview of the proposed mechanism is shown in Figure 1. Each point in the trajectory is perturbed in a two-stage process. First, the direction between the target point (i.e., the point to be perturbed) and its neighbor is perturbed, and then this point is perturbed under the restriction of the perturbed direction. To this end, the original trajectory is duplicated to create two copies, from which pivot points (i.e., the pink ones) are selected alternately to ensure that each point in the trajectory has private access to its adjacent direction information. Then, the direction between each pair of neighboring points in the two trajectories is perturbed. Under the perturbed direction constraints, each non-pivot point is perturbed using a considerably smaller point domain. Finally, the optimal perturbed result is determined based on the two independently perturbed trajectories.

4.2 Implementation of Proposed Mechanism

In this subsection, we propose the pivot sampling (TP) mechanism as the implementation of the above perturbation process. It exploits a novel strategy called pivot perturbation to capture additional direction information. An illustration of the TP mechanism

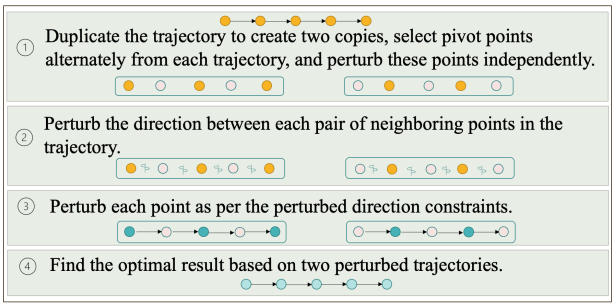


Figure 1: An overview of the proposed method (best viewed in color).

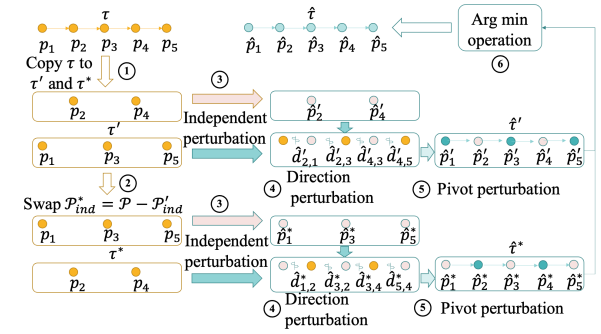
is shown in Figure 2a. The process consists of three steps: independent perturbation, pivot perturbation, and the optimal perturbed trajectory determination. To utilize direction information to restrict each point's spatial region while avoiding privacy leakage, we need to identify the pivots of each point. A pivot of a target point is treated as an origin so that the direction between the target point and itself can be calculated. The neighboring points of each point in the original trajectory are suitable candidates for pivots. However, such points are sensitive and, therefore, cannot be directly used. Furthermore, directly perturbing points in the original trajectory in a sequential manner would accumulate excessive noise. We duplicate the original trajectory to create two copies, and approximately half of the points in each trajectory can be selected as pivots in an alternate manner (i.e., by alternating between the two copies). Thus, the other non-pivot points that are not selected can access perturbed direction(s) between the neighboring pivot(s) and themselves to reduce their perturbation domains. Then, we alternately select points as the pivots in the two copies to obtain two perturbed trajectories and generate the optimal result based on them. In this way, each point in the final perturbed trajectory can enjoy both direction and distance information privately.

Let us take trajectory $\tau = \{p_i \in \mathcal{P} | 1 \leq i \leq |\tau|, |\tau| = 5\}$ in Figure 2a as an example. The perturbation process is similar if $|\tau|$ is even. First, the trajectory τ is duplicated as τ'=τ^*=τ. Then, the points in τ' (resp. τ^*) are divided into two sets — a set of pivot points to be perturbed, and a set of non-pivot points getting access to the direction information. The pivot points are selected alternately from τ' (resp. τ^*), each of which will be perturbed independently. Then the perturbation result is used to perturb non-pivot points in τ' (resp. τ^*), by utilizing the directional information between the target points and perturbed neighboring pivots, enabling each target point to enjoy a smaller perturbation domain.

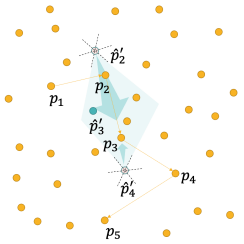
Specifically, for τ', we first select a set of pivot points (e.g., p_2 and p_4 in Figure 2a) denoted as $\mathcal{P}_{ind}^{\tau'} = \{p_2, p_4\}$. These points are perturbed using EM:

$$Pr[\hat{p} = r] = \frac{exp(\frac{\epsilon u(p,r)}{2\Delta u})}{\sum_{r' \in \mathcal{P}} exp(\frac{\epsilon u(p,r')}{2\Delta u})}, \tag{5}$$

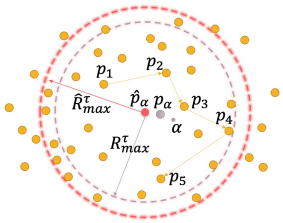
where $p \in \mathcal{P}_{ind}^{\tau'}$, $r \in \mathcal{P}$, $u(p,r) = -dist(p,r)$, $dist(\cdot)$ is the Haversine distance, and the sensitivity $\Delta u = \max_{p,r,r' \in \mathcal{P}} |u(p,r) - u(p,r')|$. For instance, the pink perturbed points $\hat{p}_2'$ and $\hat{p}_4'$ in Figure 2a are



(a) An illustration of the TP mechanism.



(b) Determine the perturbation domain of p_3.



(c) Restrict trajectory region.

Figure 2: Illustrations of the (A)TP mechanism.

the pivots in τ', denoted as $\hat{\mathcal{P}}^{\tau'}_{ind} = \{\hat{p}'_2, \hat{p}'_4\}$. Then, we can determine the perturbed directions between each point in $\mathcal{P}^{\tau'} - \hat{\mathcal{P}}^{\tau'}_{ind} = \{p_1, p_3, p_5\}$ and its neighboring pivots, i.e., the previous and the next perturbed neighboring points in τ'.

To obtain a radian interval that covers the original direction with a high probability, randomized response [25, 40] is adopted to perturb the directions. We must discretize the directions first, whose optimal granularity will be determined in Section 4.3. We take a neighboring pivot as the origin and the direction of the target point (i.e., the non-pivot point to be perturbed under the direction constraint) as the reference for the original discrete direction. Then the radian interval of a discrete direction with index $d \in \mathcal{D}$ is $[(2d-1)\frac{\pi}{|\mathcal{D}|}, (2d+1)\frac{\pi}{|\mathcal{D}|}]$, where $\mathcal{D} = \{0, 1, 2, \cdots, |\mathcal{D}|-1\}$ denotes the discrete direction universe under a given granularity. As shown in Figure 2b, when $|\mathcal{D}| = 6$ and p_3 is the target point, $\hat{p}_2$ and $\hat{p}_4$ are designated as the origins for discretizing the directions, and the direction between $\hat{p}_2$ or $\hat{p}_4$ and p_3 serves as the median value of the central angle of the corresponding original discrete direction. Let $\mathcal{D}^{\tau'}$ denote the discrete directions to be perturbed in τ'. Then, in Figure 2a, we need to perturb the directions between each point in $\hat{\mathcal{P}}^{\tau'}_{ind}$ and its perturbed neighboring pivots in τ', namely, $\mathcal{D}^{\tau'} = \{d_{2,1}, d_{2,3}, d_{4,3}, d_{4,5}\}$, where $d_{i,j}$ represents the index of the discrete direction from the i-th perturbed point to the j-th point in trajectory τ'. For instance, $d_{2,3}$ denotes the index of the discrete direction, the median value of whose central angle is towards the target point p_3

Algorithm 1: GetPointDomain

Input: Target point p_i, point set $\mathcal{P}$, trajectory τ, independently perturbed points $\hat{\mathcal{P}}^{\tau}_{ind}$ and perturbed directions $\hat{\mathcal{D}}^{\tau}$
Output: The perturbation domain $\mathcal{P}^i$ of the target point

1 $\mathcal{P}^i = \emptyset$;
2 **if** $1 < i < |\tau|$ **then**
3 $\quad \mathcal{P}^i = GetPointSet(\hat{p}_{i-1}, \hat{d}_{i-1,i}) \cap GetPointSet(\hat{p}_{i+1}, \hat{d}_{i+1,i})$;
4 $\quad$ **if** $\mathcal{P}^i = \emptyset$ **then**
5 $\quad\quad \mathcal{P}^i = \mathcal{P}$;
6 $\quad \mathcal{P}^i = \{p_i\} \cup \hat{\mathcal{P}}^i$;
7 **else if** $i = 1$ **then**
8 $\quad \mathcal{P}^i = \{p_i\} \cup GetPointSet(\hat{p}_{i+1}, \hat{d}_{i+1,i})$;
9 **else**
10 $\quad \mathcal{P}^i = \{p_i\} \cup GetPointSet(\hat{p}_{i-1}, \hat{d}_{i-1,i})$;
11 **return** $\mathcal{P}^i$;

from the origin $\hat{p}'_2$. The discrete direction is perturbed by k-RR:

$$Pr[\mathcal{K}(d) = d'] = \begin{cases} \frac{e^\epsilon}{|\mathcal{D}|-1+e^\epsilon} & \text{if } d = d', \\ \frac{1}{|\mathcal{D}|-1+e^\epsilon} & \text{if } d \neq d'. \end{cases} \tag{6}$$

Let $\hat{\mathcal{D}}^{\tau'}$ be the perturbed directions in τ'. Based on these perturbed directions and independently perturbed pivots, the perturbation domain of each remaining point in τ' can be determined by Algorithm 1. Note that the function $GetPointSet(\hat{p}_j, \hat{d}_{j,k})$ in Line 3 returns the points located in the input radian interval $\hat{d}_{j,k} \in \hat{\mathcal{D}}^{\tau}$ of the point $\hat{p}_j \in \hat{\mathcal{P}}^{\tau}_{ind}$. For instance, in Figure 2b, p_2 and p_4 are perturbed independently to $\hat{p}'_2$ and $\hat{p}'_4$ (i.e., two pink points) as two neighboring pivots of the target point p_3. The green arrows represent the perturbed discrete directions from the two pivots as origins (i.e., $\hat{p}'_2$ and $\hat{p}'_4$) to the target point p_3 (assuming that two correct discrete directions are reported). Then the set of points located in the corresponding central angle are selected as the outputs, i.e., $GetPointSet(\hat{p}'_2, \hat{d}_{2,3})$ and $GetPointSet(\hat{p}'_4, \hat{d}_{4,3})$. Finally, the intersection of the two point sets is obtained as the output of Algorithm 1, i.e., the perturbation domain of p_3, which consists of the points located in the green area shown in Figure 2b. By applying Algorithm 1, the perturbation domains of p_1, p_3 and p_5 can be obtained as $\mathcal{P}^1$, $\mathcal{P}^3$ and $\mathcal{P}^5$, respectively. These smaller perturbation domains will be increasingly accurate as ϵ increases, thus facilitating the selection of better perturbed points. Next, the other points in τ' can be perturbed with the corresponding perturbation domains using EM, that is, for each $p_j \in \mathcal{P} - \mathcal{P}^{\tau'}_{ind}$, $\mathcal{P}$ is substituted with $\mathcal{P}^j$ in Equation 5 to obtain the perturbed $\hat{p}'_j$.

The other copy τ^* can be processed in a similar way. This time, $\mathcal{P}^{\tau'}_{ind}$ and $\mathcal{P} - \mathcal{P}^{\tau'}_{ind}$ are swapped, i.e., for τ^*, we set $\mathcal{P}^{\tau^*}_{ind} = \mathcal{P}^{\tau'} - \mathcal{P}^{\tau'}_{ind}$ and $\mathcal{P}^{\tau^*} - \mathcal{P}^{\tau^*}_{ind} = \mathcal{P}^{\tau'}_{ind}$. For example, in Figure 2a, $\mathcal{P}^{\tau^*}_{ind} = \{p_1, p_3, p_5\}$ is perturbed into $\hat{\mathcal{P}}^{\tau^*}_{ind} = \{\hat{p}^*_1, \hat{p}^*_3, \hat{p}^*_5\}$ by Equation 5. Then, $\mathcal{D}^{\tau^*} = \{d_{1,2}, d_{3,2}, d_{3,4}, d_{5,4}\}$ is perturbed to obtain $\hat{\mathcal{D}}^{\tau^*} = \{\hat{d}_{1,2}, \hat{d}_{3,2}, \hat{d}_{3,4}, \hat{d}_{5,4}\}$. Based on these perturbed discrete directions, the other non-pivot points in τ^* are perturbed by EM to obtain $\hat{\tau}^*$.

By applying independent perturbation and pivot perturbation independently on both τ' and τ^* (as shown in Algorithm 3, which

Algorithm 2: Independent and Pivot Perturbation (Pivot)

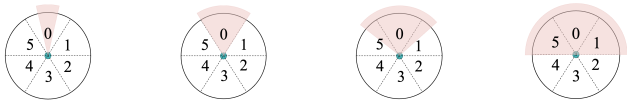
Input: Trajectory τ, point domain $\mathcal{P}$, privacy budget ϵ, discrete direction set $\mathcal{D}$, flag $\mathcal{F}$
Output: Perturbed trajectory $\hat{\tau}$

Initialize $\hat{\mathcal{P}}^{\tau}_{ind}$, $\hat{\mathcal{D}}^{\tau}$, $\mathcal{P}_{res}$ and $\hat{\mathcal{P}}_{res}$ to $\emptyset$;
$\epsilon_d = 0.75\epsilon$, $\epsilon_{ind} = (\epsilon - \epsilon_d)/2$, $\epsilon_{rest} = (\epsilon - \epsilon_d)/2$;
for $1 \le i \le |\tau|$ **do**
  `// Assuming |τ| is odd`
  if *i%2==*$\mathcal{F}$ **then**
    Use Equation 5 with $\epsilon_{ind}/|\tau|$ to perturb the i-th point in τ and obtain $\hat{p}_i$;
    $\hat{\mathcal{P}}^{\tau}_{ind} = \hat{\mathcal{P}}^{\tau}_{ind} \cup \{\hat{p}_i\}$;
  else
    $\mathcal{P}_{res} = \mathcal{P}_{res} \cup \{p_i\}$;
for p_j *in* $\mathcal{P}_{res}$ **do**
  Use Equation 6 with equal budget $\epsilon_d/2(|\tau| - 1)$ to perturb the discrete directions $d_{i-1,i}$ and $d_{i+1,i}$;
  $\hat{\mathcal{D}}^{\tau} = \hat{\mathcal{D}}^{\tau} \cup \{\hat{d}_{i,i-1}, \hat{d}_{i,i+1}\}$;
  $\mathcal{P}^j = GetPointDomain(p_j, \mathcal{P}, \tau, \hat{\mathcal{P}}^{\tau}_{ind}, \hat{\mathcal{D}}^{\tau})$;
  Use Equation 5 with $\epsilon_{rest}/|\tau|$ by substituting $\mathcal{P}$ with $\mathcal{P}^j$ to obtain $\hat{p}_j$;
  $\hat{\mathcal{P}}_{res} = \hat{\mathcal{P}}_{res} \cup \{\hat{p}_j\}$;
$\hat{\tau} = OrderedByOriginalIdx(\hat{\mathcal{P}}^{\tau}_{ind} \cup \hat{\mathcal{P}}_{res})$;
return $\hat{\tau}$;

Algorithm 3: TP Mechanism (TP)

Input: Trajectory τ, point set $\mathcal{P}$, privacy budget ϵ
Output: Perturbed trajectory $\hat{\tau}$

$\tau' = \tau$, $\tau^* = \tau$;
Solve Equation 8 to obtain the discrete direction set $\mathcal{D}$;
`// Pivot perturbation, assuming |τ| is odd`
$\hat{\tau}^* = Pivot(\tau^*, \mathcal{P}, \epsilon/2, \mathcal{D}, \mathcal{F} = 1)$;
$\hat{\tau}' = Pivot(\tau', \mathcal{P}, \epsilon/2, \mathcal{D}, \mathcal{F} = 0)$;
`// Find the optimal perturbed trajectory`
Solve Equation 7 to obtain $\hat{\tau}$ based on $\hat{\tau}^*$ and $\hat{\tau}'$;
return $\hat{\tau}$;



(a) $\theta = \pi/12$ **(b)** $\theta = \pi/6$ **(c)** $\theta = \pi/4$ **(d)** $\theta = \pi/2$

Figure 3: Calculation of the average direction-preserving success probability. The pink area denotes the query range θ , with sector 0 serving as the input for k-RR, representing the original discrete direction, the median value of whose central angle is towards a target point from the green origin.

invokes Algorithm 2 twice with flag $\mathcal{F} = 1$ and 0, respectively), two perturbed points are obtained for each point in τ: one based on only the distance factor of the original point, and the other perturbed under the direction restriction of neighboring pivot(s). Then, these two trajectories are combined to obtain the optimal perturbed trajectory by solving the following equation:

$$\arg\min_{\hat{\tau}} \sum_{i=1}^{|\hat{\tau}|} dist(\hat{p}_i, \hat{p}'_i) + dist(\hat{p}_i, \hat{p}^*_i), \tag{7}$$

where $\hat{p}_i \in \mathcal{P}$ is the i-th point in $\hat{\tau}$, $\hat{p}'_i \in \hat{\tau}'$, $\hat{p}^*_i \in \hat{\tau}^*$, and $dist(\cdot)$ denotes the Haversine distance. Then, the final perturbed trajectory benefits from bi-directional information. However, the challenge remains on how to determine the optimal direction granularity.

4.3 Choosing the Direction Granularity

From the perspective of the utility of the proposed mechanism, a coarse-grained direction would cover a greater number of points, leading to a higher recall rate for the chosen points. Furthermore, a smaller $|\mathcal{D}|$ increases the accuracy of the k-RR mechanism. Unfortunately, the coverage of a greater number of noisy points may result in a considerably large perturbation domain when EM is used to perturb points under the perturbed direction constraint. In contrast, a fine-grained direction can lead to a smaller perturbation domain, but faces more challenges to select the exact direction due to the large $|\mathcal{D}|$. In this case, even though it is difficult to identify the original discrete direction, there is chances to obtain discrete directions close to the original one. As ϵ increases, the probability of obtaining such a close discrete direction increases.

In a nutshell, a proper choice of the direction granularity represents a critical trade-off between the preservation of direction and the perturbation domain size of EM. The influence of the perturbation domain size depends not only on the chosen direction granularity but also on the distribution of points on the map. The analysis of the preservation probability of the target point to be perturbed is difficult because the distribution varies for each point. Therefore, we opt to design a general strategy to guide the choice of the direction granularity independent of a specific dataset. We focus on only the direction-preserving performance under different granularities based on the k-RR mechanism and different ϵ values.

Since the k-RR mechanism always returns the original discrete direction with the highest probability, evaluating the direction-preserving performance within just one constant range may not be the fairest approach. The granularity with the highest probability is always the one that is closest to the constant range. Therefore, we choose to compare the average direction-preserving success probability across different radian intervals of different granularities under a given ϵ value. To this end, we choose the granularity $|\mathcal{D}|$ as follows:

$$\arg\max_{g \in \mathcal{D}_c} \sum\nolimits_{\theta_j \in \Theta} \sum\nolimits_{d_i \in \mathcal{R}(g)} \varphi(d_i; \theta_j) \cdot \lambda(d_i; \epsilon; g)/|\Theta|, \tag{8}$$

$$\varphi(d_i; \theta_j) = \frac{|[(2d_i - 1)\frac{\pi}{g}, (2d_i + 1)\frac{\pi}{g}] \cap [-\theta_j, \theta_j]|}{2\pi/g},$$

$$\lambda(d_i; \epsilon; g) = \begin{cases} \frac{e^{\epsilon}}{g-1+e^{\epsilon}} & \text{if } d_i = d, \\ \frac{1}{g-1+e^{\epsilon}} & \text{otherwise,} \end{cases}$$

where $\mathcal{D}_c$ denotes the candidate direction granularities, $\mathcal{R}(g) = \{0, 1, 2, \cdots, g-1\}$ denotes the discrete directions based on a candidate granularity g, $\Theta = \{\theta_j | \theta_j = \pi/g, g \in \mathcal{D}_c\}$ denotes the set of direction-preserving query ranges, and d represents the correct

Algorithm 4: ATP Mechanism

Input: Trajectory τ, point set $\mathcal{P}$, privacy budget ϵ, discrete direction set $\mathcal{D}$
Output: Perturbed trajectory $\hat{\tau}$

// Copy the trajectory, assuming $|\tau|$ is odd
$\tau' = \tau, \tau^* = \tau$;
// Independent and pivot perturbation
$\epsilon_R = 0.25\epsilon, \epsilon_3 = \epsilon - \epsilon_R$;
$\hat{\mathcal{P}}^*_\alpha = RTR(\tau^*, \mathcal{P}, \epsilon_R/2)$;
$\hat{\mathcal{P}}'_\alpha = RTR(\tau', \mathcal{P}, \epsilon_R/2)$;
$\hat{\tau}^* = Pivot(\tau^*, \hat{\mathcal{P}}^*_\alpha, \epsilon_3/2, \mathcal{D}, \mathcal{F} = 1)$;
$\hat{\tau}' = Pivot(\tau', \hat{\mathcal{P}}'_\alpha, \epsilon_3/2, \mathcal{D}, \mathcal{F} = 0)$;
// Find the optimal perturbed trajectory
Solve Equation 7 to obtain $\hat{\tau}$ based on $\hat{\tau}^*$ and $\hat{\tau}'$;
return $\hat{\tau}$;

discrete direction. The calculation of the average success probability can be treated as a weighted average of those results from direction-preserving range queries, i.e., the probabilities of preserving the correct radian intervals under different $g \in \mathcal{D}_c$ values when perturbed by k-RR with a given ϵ value.

For example, in Figure 3, assume that the current granularity $g = 6 \in \mathcal{D}_c = \{2, 4, 6, 12\}$, 0 is the original discrete direction, $\Theta = \{\pi/2, \pi/4, \pi/6, \pi/12\}$, and $q = \frac{e^\epsilon}{g-1+e^\epsilon}$. The query values in Θ correspond to candidate granularities. When the query is $\theta = \pi/12$, as shown in Figure 3a, $\lambda(d_i = 0; \epsilon; g = 6)$ returns q, and the weight $\varphi(d_i = 0; \theta = \pi/12)$ is the proportion of the pink radian interval to the entire radian interval of the sector that is denoted as 0, i.e., $\frac{|[-\frac{\pi}{12}, \frac{\pi}{12}] \cap [-\frac{\pi}{12}, \frac{\pi}{12}]|}{2\pi/6} = \frac{\pi/12-(-\pi/12)}{2\pi/6} = \frac{1}{2}$. The success probability is calculated in a similar way when the query is changed, as shown in Figure 3b-3d. The selection of g is independent of a specific point, but relies on the privacy budget and candidate direction granularities. In particular, g should be between 2 and 12; otherwise, the region under the perturbed direction constraint would be too small to cover the correct points. When ϵ increases, the perturbed direction is more accurate, and the benefits of the fine-grained direction becomes more significant. Through extensive experiments under different ϵ values, we observe that the granularity chosen by comparing the average direction-preserving success probability is close to the granularity with the best performance.

4.4 Privacy Analysis

In this subsection, we show that the proposed TP mechanism preserves differential privacy.

Theorem 4.1. *The TP mechanism satisfies ϵ-LDP.*

Proof. For trajectory τ (we assume that the number of points $|\tau|$ in τ is odd. The proof is similar when it is even), raw sensitive data are accessed in three ways after copying τ into τ' and τ^*: independent point perturbations, direction perturbations, and other point perturbations based on the directions.

For the independent point perturbations, we use EM with privacy budget $\epsilon_{ind}/|\tau|$ for each point. The points with an even index in τ are selected as pivots when perturbing τ' and independently perturbed. Each independent perturbation of these points satisfies $\epsilon_{ind}/|\tau|-LDP$. The other points are selected as pivots when perturbing τ^* and independently perturbed. Each independent perturbation also satisfies $\epsilon_{ind}/|\tau|$-LDP. The total number of independent perturbations is $|\tau|$.

Algorithm 5: Restrict Trajectory Region (RTR)

Input: Trajectory τ, point set $\mathcal{P}$, privacy budget ϵ_R
Output: Perturbed point domain $\hat{\mathcal{P}}_\alpha$

Use Equation 9 to calculate the trajectory anchor α and map it to the nearest point p_α;
Use Equation 5 with $\epsilon_1 = 0.25\epsilon_R$ to perturb p_α into $\hat{p}_\alpha$;
Get R^τ_{max} and perturb it using the SW mechanism with $\epsilon_2 = \epsilon_R - \epsilon_1$;
Calibrate the perturbed $\hat{R}^\tau_{max}$ to obtain $\hat{R}$ using Equation 10;
Get the perturbed point set $\hat{\mathcal{P}}_\alpha$ of τ based on $\hat{p}_\alpha$ and $\hat{R}$;
return $\hat{\mathcal{P}}_\alpha$;

In the case of direction perturbations in pivot perturbations, recall that the direction is perturbed by considering the perturbed adjacent point(s) of a target point as the origin. By the post-processing property of LDP, making use of such an origin does not consume any additional privacy budget. Each perturbation for direction $d_{i,j}$ by k-RR satisfies $\epsilon_d/2(|\tau|-1)$-LDP. The total number of direction perturbations is $2(|\tau|-1)$.

Finally, based on the perturbed directions and the independently perturbed pivot points, each of the other non-pivot points in τ' and τ^* is perturbed by EM with budget $\epsilon_{rest}/|\tau|$ (i.e., point perturbations under the direction restrictions in pivot perturbations). Then, $\hat{\tau}'$ and $\hat{\tau}^*$ are obtained, and the optimal perturbed trajectory is determined. By the post-processing property, this step does not require any additional privacy budget. Thus, according to the sequential composition theorem, the entire TP mechanism satisfies ϵ-LDP, where $\epsilon = \epsilon_{ind} + \epsilon_{rest} + \epsilon_d$. □

5 ANCHOR-BASED PIVOT SAMPLING MECHANISM

The TP mechanism uses pivot perturbation to capture bi-directional information and combines the direction information with independent perturbation of each point in a trajectory. Pivot perturbations can restrict the spatial region of the points in the trajectory to reduce the negative impact of the large point domain to a certain extent. However, a large number of independent perturbations still depend on a large $|\mathcal{P}|$. These independent perturbations introduce considerable amount of noise and almost uniformly select points in the large $\mathcal{P}$ when the privacy budget ϵ is small, which makes the perturbed points far from the actual region of the trajectory. To mitigate this problem, we propose an anchor-based pivot sampling (ATP) mechanism in this section.

5.1 Restricting Trajectory Region

According to the first law of geography [37], the spatial region of a trajectory is likely to be relatively small when compared with the entire area of the map. The spatial region can be used to limit the domain of the points in a trajectory, which is used by our ATP mechanism to restrict the point domain of a trajectory. In contrast to the TP mechanism, the ATP mechanism (as shown in Algorithm 4)

first restricts the trajectory region, i.e., the perturbation domain of a trajectory. Specifically, before perturbing each copy of the trajectory, the trajectory anchor is calculated and perturbed, and then the region size is determined based on the perturbed longest distance between the perturbed anchor and the trajectory (Lines 3 and 4). After that, the trajectory can be perturbed in a sequential manner, similar to that in the TP mechanism (Lines 5 and 6).

With regard to the trajectory region, ATP identifies the points that are more centralized and as close to the original region as possible while consuming an acceptable amount of the privacy budget. In what follows, we present a simple and lightweight method to determine this region. We first need to determine a way of representing a trajectory and its region. This representation must encode the information about the points in the trajectory, especially the geographic information, and the shape of the region needs to be carefully chosen to avoid privacy leakage and enhance data utility. As such, we propose an anchor-based method (see Algorithm 5) that restricts each trajectory to a constant shape, i.e., a circular region. For each trajectory, the anchor α is first calculated and mapped to the nearest point as the anchor point p_α (Line 1) as follows:

$$lat(\alpha) = \sum_{i=1}^{|\tau|} lat(p_i)/|\tau|, lon(\alpha) = \sum_{i=1}^{|\tau|} lon(p_i)/|\tau|,$$
$$p_\alpha = \arg\min_{p} dist(\alpha, p), \tag{9}$$

where $p \in \mathcal{P}$, τ is the trajectory, $lat(p_i)$ and $lon(p_i)$ denote the latitude and longitude of p_i, respectively, and $dist(\cdot)$ is the Haversine distance. Then, the anchor point is perturbed using EM to obtain the perturbed point $\hat{p}_\alpha$ (Line 2):

$$Pr[\hat{p}_\alpha = p] = \frac{exp(\frac{\epsilon u(p_\alpha, p)}{2\Delta u})}{\sum_{p' \in \mathcal{P}} exp(\frac{\epsilon u(p_\alpha, p')}{2\Delta u})},$$

where $p \in \mathcal{P}$, and $u(p_\alpha, p) = -dist(p_\alpha, p)$. The anchor α is mapped to $p_\alpha \in \mathcal{P}$ so that each possible $u(p_\alpha, p)$ can be pre-calculated to increase the efficiency of the perturbation mechanism. For example, in Figure 2c, the anchor α is calculated and mapped to p_α, and then p_α is perturbed to $\hat{p}_\alpha$ by EM.

After the perturbed anchor point is obtained, the region of the trajectory can be determined. A simple way is to set a pre-defined and fixed value for the radius R. Then, the perturbation domain is obtained as follows:

$$\hat{\mathcal{P}}_\alpha = \{p \in \mathcal{P} | dist(\hat{p}_\alpha, p) \leq R\}.$$

Since the spatial region depends on the radius R, it should be carefully chosen.

Intuitively, different users are likely to produce trajectories with varying area sizes. If a fixed radius R is set, each trajectory will be restricted to the same granularity, resulting in loss of a considerable amount of information or introducing many noisy points in the candidate point set. To address this challenge, we propose an adaptive strategy to set different R values for different trajectories.

We aim to determine R values that are not too large and ensure that the region covers the points of a trajectory to the best extent possible while preserving privacy. The longest distance R^τ_{max} (i.e., the purple arrow in Figure 2c) between a point in the trajectory τ and the perturbed anchor point $\hat{p}_\alpha$ can be obtained as follows:

$$R^\tau_{max} = \max_{p \in \mathcal{P}^\tau} dist(\hat{p}_\alpha, p),$$

where $\mathcal{P}^\tau$ denotes the set of the points in trajectory τ. To ensure privacy, R^τ_{max} must be perturbed (Line 3 of Algorithm 5). Since R^τ_{max} is the distance between the perturbed point $\hat{p}_\alpha \in \mathcal{P}$ and a point in the trajectory, i.e., $p \in \mathcal{P}^\tau \subseteq \mathcal{P}$, it is bounded by the maximum distance between $\hat{p}_\alpha$ and a point in $\mathcal{P}$. Therefore, we use SW mechanism, which is designed to perturb bounded continuous data. In comparison with the piecewise mechanism [28], the SW mechanism is more concentrated, which is suitable for our task. As the input and output ranges of the SW mechanism are $[0, 1]$ and $[-b, b+1]$, respectively, the distance is first normalized via $r = \frac{R^\tau_{max}}{\Delta R}$, where $\Delta R = \max_{p \in \mathcal{P}} dist(\hat{p}_\alpha, p)$. Then, r is perturbed to $\hat{r}$ by Equation 4. Finally, the output is mapped to the real distance range via $\hat{R}^\tau_{max} = \frac{(\hat{r}+b)\Delta R}{2b+1}$.

After the perturbed radius $\hat{R}^\tau_{max}$ (i.e., the red arrow in Figure 2c) is obtained, the regions of different trajectories will vary adaptively. Due to the property of the SW mechanism, the perturbed $\hat{R}^\tau_{max}$ is likely to be too small or large when given a small ϵ. Here we make a few observations. First, the distribution of the region size of human mobility trajectory can be approximated by a power-law distribution [20], implying that there should be only a few large sizes. On the other hand, the perturbed radius may become smaller due to the large randomness from a small ϵ. In the case of a trajectory region, although a larger perturbed radius is likely to cover more points and leads to a larger perturbation domain, it can increase the coverage rate of the points in the original trajectory while being still relatively small in comparison with the entire area size. That is, the benefit of a larger radius outweighs the disadvantage of the noise it introduces when the original radius is small. To this end, we guide the perturbation by a calibration that "pulls" the perturbed radius close to a "center" when ϵ is small.

5.2 Calibrating the Radius

The radius calibration includes two steps: the center of calibration is first determined, and then the moving step size is calculated and the radius is calibrated based on ϵ.

Recall that R^τ_{max} is bounded and is determined by the farthest point from $\hat{p}_\alpha$ in the trajectory. To identify a proper value η for the calibration center based on ϵ and $\hat{R}^\tau_{max}$, we aim to amplify the impact of the points satisfying a possible range of the farthest point from $\hat{p}_\alpha$. Since the input domain of SW is $[0, 1]$, we sample some test values uniformly to avoid privacy leakage and calculate the possible ranges. Let $\mathcal{V} = \{v_k | v_k = 0.1 \times k, k = 0, 1, ..., 10\}$ denote the test value set, $l_k = v_k - b$, and $u_k = v_k + b$, we have

$$\mathcal{V}^R = \{v_k \in \mathcal{V} | l_k \leq \frac{(2b+1)\hat{R}^\tau_{max}}{\Delta R} - b \leq u_k, 0 \leq k \leq 10\},$$

$$l = \min_{v_k \in \mathcal{V}^R} v_k, u = \max_{v_k \in \mathcal{V}^R} v_k,$$

$$\mathcal{P}^R = \{p \in \mathcal{P} | l \leq \frac{(2b+1)dist(\hat{p}_\alpha, p)}{\Delta R} - b \leq u\}.$$

The values in $\mathcal{V}$ are used for testing, and for each value v_k, the output range with high probability $q = \frac{e^\epsilon}{2be^\epsilon+1}$ is used to determine whether it covers $\hat{R}^\tau_{max}$. The candidate points, i.e., the points whose distance to $\hat{p}_\alpha$ are within the desired range, are obtained. Then, the weighted average distance is calculated as the calibration center

that assigns larger weights to the candidate points as follows:

$$\eta = \frac{\sum_{p\in\mathcal{P}^R} q \cdot dist(\hat{p}_\alpha, p) + \sum_{p'\in\mathcal{P}-\mathcal{P}^R} (1-q) dist(\hat{p}_\alpha, p')}{q|\mathcal{P}^R| + (1-q)|\mathcal{P} - \mathcal{P}^R|}.$$

After obtaining the calibration center η, we must determine how to "pull" the perturbed radius $\hat{R}^\tau_{max}$ close to it. As per the property of the SW mechanism, the perturbed value becomes more concentrated as the privacy budget ϵ increases. Therefore, the calibration effect should be reduced with increasing ϵ. The perturbed radius $\hat{R}^\tau_{max}$ is calibrated as follows (Line 4 of Algorithm 5):

$$\hat{R} = \hat{R}^\tau_{max} + \xi e^{-\epsilon}, \tag{10}$$

$$\xi = (\eta - \hat{R}^\tau_{max}) \cdot \frac{1}{1 - e^{-\beta/2}}, \tag{11}$$

$$\beta = \begin{cases} \frac{\eta - \hat{R}^\tau_{max}}{\eta} & \text{if } \hat{R}^\tau_{max} \le \eta, \\ \frac{\hat{R}^\tau_{max} - \eta}{\Delta R - \eta} & \text{otherwise.} \end{cases} \tag{12}$$

In Equation 10, the calibration term (i.e., the last term) is composed of two factors: the decay factor $e^{-\epsilon}$ and the moving factor ξ. The decay factor controls the effect of the calibration term by decreasing its influence as ϵ grows. The moving factor ξ depends on the gap between the perturbed radius and the calibration center. As shown in Equation 11, the moving factor will be negative when the perturbed radius $\hat{R}^\tau_{max}$ is larger than η, and positive otherwise. The gap between $\hat{R}^\tau_{max}$ and η is multiplied by a factor in the range of $[0, 1]$ to avoid obtaining an extreme value or a constant value. Therefore, β is calculated using Equation 12, which normalizes the gap between $\hat{R}^\tau_{max}$ and η to $[0, 1]$. Then $\beta/2$ is fed into an activation function $sigmoid(\cdot)$, which provides the non-linear property and prevents the moving step size from having an exact value of $|\eta - \hat{R}^\tau_{max}|$. After the calibration, we can determine the perturbed point set of the input trajectory (Line 5 of Algorithm 5).

5.3 Analysis of Privacy and Utility

In this subsection, we prove that the proposed ATP mechanism satisfies ϵ-LDP.

Theorem 5.1. *The ATP mechanism satisfies ϵ-LDP.*

Proof. The ATP mechanism restricts the spatial regions of the two trajectory copies in a sequential manner. The perturbed anchors $\hat{p}'_\alpha$ and $\hat{p}^*_\alpha$ are obtained using EM, each satisfying $\epsilon_1/2$-LDP. Then $\hat{R}'$ and $\hat{R}^*$ are calculated to restrict the perturbation domains $\hat{\mathcal{P}}'_\alpha$ and $\hat{\mathcal{P}}^*_\alpha$ of the points in $\hat{\tau}'$ and $\hat{\tau}^*$, respectively. The anchors are perturbed, and then the largest distance between the perturbed $\hat{p}'_\alpha$ (or $\hat{p}^*_\alpha$) and the points in τ' and τ^* is calculated as $\hat{R}'$ (or $\hat{R}^*$) by the SW mechanism, which satisfies $\epsilon_2/2$-LDP. The calibration operation does not access any raw data because the test values are independent of the trajectory and the distances used to identify $\mathcal{P}^R$ are based on the perturbed $\hat{p}'_\alpha$ (or $\hat{p}^*_\alpha$). By the post-processing property, the calibration operation does not require additional privacy budget. Next, $\mathcal{P}$ is replaced once with $\hat{\mathcal{P}}'_\alpha$ and then with $\hat{\mathcal{P}}^*_\alpha$. The TP mechanism that satisfies $\epsilon_3/2$-LDP is then invoked. Again, by the post-processing theorem, the replacements do not consume additional privacy budget. Finally, the sequential composition theorem ensures that the ATP mechanism satisfies ϵ-LDP, where $\epsilon = \epsilon_1 + \epsilon_2 + \epsilon_3$. □

To understand the impact of different R values, we perform a utility analysis by estimating the upper bound of the probability that the anchor points p_α in the trajectory are not covered by the circle whose radius is R and origin is the perturbed $\hat{p}_\alpha$.

Theorem 5.2. *Let $C_\tau(p, R) = \{p' \in \mathcal{P} | dist(p, p') \le R\}$ and u_x be the minimum score that satisfies $u(p_\alpha, p) = -dist(p_\alpha, p) > -R$. If $R = 2t\Delta u/\epsilon - u_x$, we have*

$$Pr[p_\alpha \notin C_\tau(\hat{p}_\alpha, R)] \le \frac{|\mathcal{P} - C_\tau(p_\alpha, R)|}{|C_\tau(p_\alpha, R)|} exp(-t).$$

Proof. We have the probability

$$Pr[p_\alpha \notin C_\tau(\hat{p}_\alpha, R)] = Pr[\hat{p}_\alpha \notin C_\tau(p_\alpha, R)] = Pr[u(p_\alpha, \hat{p}_\alpha) \le -R]$$

$$= \frac{\sum_{p_c\in\mathcal{P}-C_\tau(p_\alpha,R)} exp(\frac{\epsilon u(p_\alpha,p_c)}{2\Delta u})}{\sum_{p\in\mathcal{P}} exp(\frac{\epsilon u(p_\alpha,p)}{2\Delta u})} \le \frac{|\mathcal{P} - C_\tau(p_\alpha, R)| exp(\frac{\epsilon(-R)}{2\Delta u})}{|C_\tau(p_\alpha, R)| exp(\frac{\epsilon u_x}{2\Delta u})}$$

$$= \frac{|\mathcal{P} - C_\tau(p_\alpha, R)|}{|C_\tau(p_\alpha, R)|} exp(-\frac{\epsilon(R + u_x)}{2\Delta u}).$$

Let $R = \frac{2t\Delta u}{\epsilon} - u_x$, then we have

$$Pr[p_\alpha \notin C_\tau(\hat{p}_\alpha, R)] \le \frac{|\mathcal{P} - C_\tau(p_\alpha, R)|}{|C_\tau(p_\alpha, R)|} exp(-t),$$

which completes the proof. □

According to Theorem 5.2, for a certain R, the probability of an anchor point p_α being not covered by the circle with the origin at $\hat{p}_\alpha$ and radius R decreases exponentially as the radius R increases. For the sake of data utility, a small probability can reduce the error of anchor point selection. Furthermore, a small R can avoid resulting in an overly large perturbation domain and introducing excessive noise. Hence the selection of R becomes a trade-off with respect to the privacy budget ϵ. Besides, this selection of R is also related to the point distribution of a dataset, i.e., the minimum score u_x that satisfies $u(p_\alpha, p) = -dist(p_\alpha, p) > -R$. A smaller u_x results in a tighter bound in the theorem. For this reason, we propose an adaptive approach, as described in Sections 5.1 and 5.2, to find a reasonable R to improve data utility.

6 EXPERIMENTS

6.1 Experimental Setting

6.1.1 Datasets. In the experiments, we use three real-world and one synthetic datasets, namely, NYC, CHI, CLE, and CPS. NYC consists of check-in trajectories in New York City extracted from the Foursquare dataset [43], while CHI and CLE, extracted from the Gowalla dataset [12], consist of check-in trajectories in Chicago and Cleveland, respectively.[1] We consider the 1,000 most popular points as $\mathcal{P}$ to generate CHI and CLE, and the 2,000 most popular POIs in NYC. For a fair comparison, we adopt the same preprocessing steps used in the previous study [14]. We randomly delete the points that appear within a 10-minute duration in each trajectory until only one point remains. If the time interval between any two adjacent points in a trajectory exceeds three hours, we split it into two trajectories. After these preprocessing steps, we obtain 7,951, 3,162, and 2,794 trajectories in NYC, CHI, and CLE, respectively. For CPS, we follow

[1]We select the points in the approximate range of [87.4W—88W, 41.6N—42N] for Chicago and [122.46W—122.9W, 45.4N—45.6N] for Cleveland.

the previous study [14] to generate trajectories on the campus of the University of British Columbia[2]. We take 262 campus buildings as $\mathcal{P}$ and generate 4,000 trajectories.

6.1.2 Baselines and Parameter Setting. The only study that satisfies pure ϵ-LDP is NGRAM mechanism [14], which perturbs POI trajectories by incorporating external knowledge. As aforementioned, it is often difficult to acquire such external knowledge in practice, which is the key motivation of our paper. Hence, we consider NGRAM mechanism without any additional knowledge as a baseline and set the grid granularity to 3 or 4 for different datasets. Another baseline is a direct application of the exponential mechanism (referred to as EXP). It perturbs each point in a trajectory by using the same utility function used in the mechanisms proposed in this study, i.e., $-dist(\cdot)$. The last baseline is CGM [3], a state-of-the-art mechanism for streaming data collection under (ϵ, δ)-LDP. We normalize the latitude and longitude of each point in the same way as in the previous study [3], by setting $\delta = 10^{-2}$ or 10^{-1}, and $C = 0.1$. For all the mechanisms, we use the Haversine distance as the distance metric.

As for the privacy budget allocation scheme in the ATP mechanism, $\epsilon' = \epsilon^* = \frac{\epsilon}{2}$ are used to perturb τ' and τ^* respectively. For ϵ', $\frac{\epsilon'}{4}$ is used to determine the region. As the region size plays a more significant role in determining the trajectory region, a quarter of $\frac{\epsilon'}{4}$ is used to perturb the trajectory anchor while the other three quarters are used to perturb the radius. The remaining budget (i.e., $\frac{3\epsilon'}{4}$) is used to perturb the directions and points in τ'. As the directions have a larger impact on the perturbations of trajectories, three quarters of $\frac{3\epsilon'}{4}$ are uniformly divided to perturb the directions while another quarter is uniformly divided to perturb the points. The allocation of ϵ^* for perturbing τ^* is the same as τ'. As for the TP mechanism, we use the same budget allocation strategy for the perturbations of directions and points in the ATP mechanism. All the mechanisms are executed 5 times and the average is plotted.

6.2 Results

6.2.1 Measures. We evaluate the utility of the mechanisms using two measures adopted in the previous study [14]. The first measure is the mean normalized error (NE), which is the normalized distance between each point of a perturbed trajectory and the corresponding point of the original trajectory:

$$NE = \frac{1}{|\mathcal{T}|}\sum_{i=1}^{|\mathcal{T}|}\frac{1}{|\tau_i|}\sum_{j=1}^{|\tau_i|} dist(\hat{p}_j, p_j),$$

where $|\mathcal{T}|$ is the number of trajectories, and $dist(\cdot)$ denotes the Haversine distance. We normalize the results by the maximum distance in a dataset to make the results easier to understand. The other measure is the preservation range query (PRQ), which evaluates whether each point of the perturbed trajectory is within the δ (km) range of the corresponding true point:

$$PRQ = \frac{1}{|\mathcal{T}|}\sum_{i=1}^{|\mathcal{T}|}\frac{1}{|\tau_i|}\sum_{j=1}^{|\tau_i|} \pi(p_j, \hat{p}_j, \delta) \times 100\%,$$

$$\pi(p_j, \hat{p}_j, \delta) = \begin{cases} 1 & \text{if } dist(\hat{p}_j, p_j) \le \delta, \\ 0 & \text{otherwise,} \end{cases}$$

[2]https://github.com/UBCGeodata/ubcv-buildings

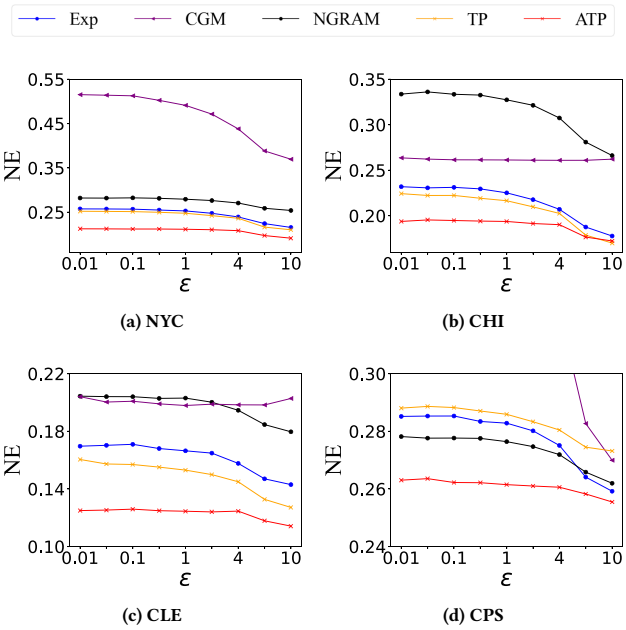


Figure 4: Mean normalized errors (NEs) for different methods when varying privacy budget ϵ.

where $dist(\cdot)$ is also measured by the Haversine distance. A larger PRQ value indicates better performance.

6.2.2 Evaluation of Utility. We first compare our proposed mechanisms with the baseline mechanisms in terms of NE. The results are presented in Figure 4. The proposed mechanisms outperform the baselines in most situations. From Figure 4, we observe that the TP mechanism (i.e., the orange line) outperforms the baselines on all real-world datasets. Although the EM used to sample points in the TP mechanism has difficulty in choosing points near the original point when ϵ is small, the use of the direction constraint can effectively enhance the utility. In NGRAM, since external knowledge is not used, the hierarchical decomposition in this mechanism cannot achieve significant performance improvement, and the large bigram universe adversely affects the utility. In CGM, the unbounded noise largely impacts the resulting utility. Although the TP mechanism performs worse than the baselines on the synthetic dataset, the ATP mechanism (represented by the red line) performs significantly better than other mechanisms, showing that the trajectory region constraint is more helpful when the points in a dataset are distributed more uniformly, such as the CPS dataset. The better performance is partially due to the fact that the SW mechanism used to perturb the region size R is more accurate when ϵ is large. The proposed adaptive calibration method is also advantageous to eliminate extreme values of region sizes due to the perturbation of the SW mechanism. As a result, even when ϵ is small, ATP can restrict trajectory regions better and achieve better performance.

With regard to the PRQ, the ATP mechanism outperforms the baseline mechanisms on all three real-world datasets as shown in Figure 5. We set δ to 1, 2, or 4, and observe the changes in *PRQ* when ϵ varies from 1 to 10 on different datasets (we set δ to 0.25,

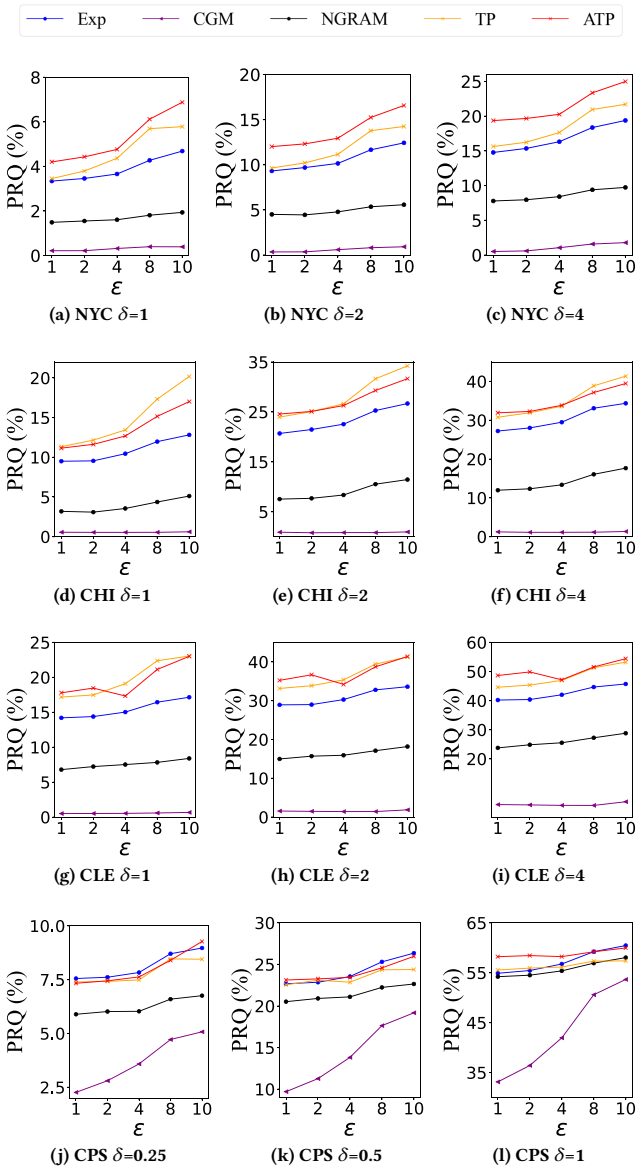


Figure 5: Preservation range queries (PRQs) vs. privacy budget ϵ under different ranges δ.

0.5, or 1 for the CPS dataset due to the small entire area). The value of *PRQ* increases as the privacy budget increases. For the CPS dataset, although the ATP mechanism performs worse than EXP when $\delta = 0.25$, it performs better when δ becomes larger, especially when ϵ is small.

6.2.3 Effects of Parameters. Due to the space limitation, we only report the results on the NYC and CLE datasets. Similar results and conclusions can be observed and drawn on the CHI and CPS datasets. First, we study the performance of the ATP mechanism with varying fixed R values. We experiment with different R values on different datasets: $R \in \{5, 10, 15, 20\}$ for NYC and $R \in \{2, 4, 6, 8\}$ for CLE. As shown in Figures 6a and 6d, as ϵ increases, a larger

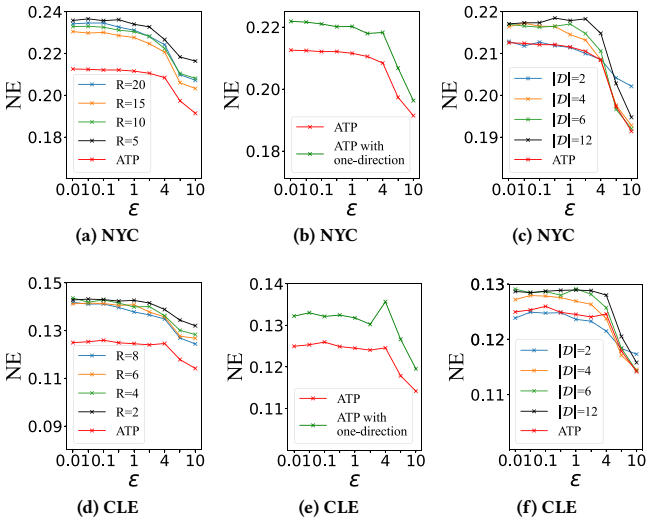


Figure 6: Mean normalized errors (NEs) of different radius R, uni-directional information and direction granularities under varying privacy budget ϵ.

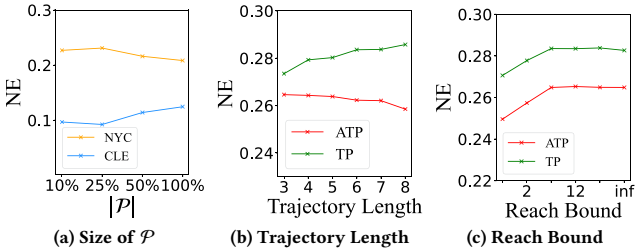


Figure 7: Mean normalized errors (NEs) under different sizes of $\mathcal{P}$, trajectory length and reach bound ($\epsilon = 4$).

R value can cover more points in the trajectories whose original regions are large, leading to better performance. We observe that the value of R that results in the best performance is not always the largest or smallest on different datasets, irrespective of whether ϵ is small or large. Therefore, we advise to use our adaptive method to determine trajectory regions, which can restrict trajectory regions more precisely and thus obtain the best performance.

Next, we investigate whether the bi-directional information improves the performance of our mechanism. From Figures 6b and 6e, it can be seen that the ATP mechanism using bi-directional information outperforms the same mechanism using uni-directional information, which confirms that additional direction information enables more accurate restriction of trajectory regions.

We also study the effects of different direction granularities $|\mathcal{D}| \in \{2, 4, 6, 12\}$ on the performance. As shown in Figures 6c and 6f, when ϵ is small, the ATP mechanism with coarse-grained directions performs better than that with fine-grained directions. The choice of the cardinality of discrete directions mainly depends on the given privacy budget. Note that direction selection aims to restrict

Table 1: Comparison of success probability. Each value is calculated as per Equation 8. The boldfaced values are the largest values under different ϵ values.

	$\|\mathcal{D}\|$=2	$\|\mathcal{D}\|$=4	$\|\mathcal{D}\|$=6	$\|\mathcal{D}\|$=12
ϵ=0.01	**0.25035156**	0.20871446	0.16699901	0.09869639
ϵ=0.05	**0.25175778**	0.21024432	0.16833461	0.09954752
ϵ=0.1	**0.25351539**	0.21216864	0.17001818	0.10062269
ϵ=0.5	**0.26754921**	0.22803653	0.18405465	0.10968742
ϵ=1	**0.28492633**	0.24901168	0.20304037	0.12224172
ϵ=2	**0.3185154**	0.29434453	0.24584151	0.15185139
ϵ=4	0.37745749	**0.39365306**	0.34902402	0.23167506
ϵ=8	0.45232527	0.57649644	**0.58164843**	0.47196792
ϵ=10	0.47167379	0.63974545	**0.6787087**	0.60876684

the perturbation domain, i.e., $|\mathcal{D}| = g$ means to narrow down the domain to $\frac{1}{g}$, which reduces the location perturbation noise and thus improves the data utility. However, direction perturbation itself also introduces some noise, where finer direction granularity comes with heavier perturbation under LDP. Therefore, for a small ϵ, we tend to select a small $|\mathcal{D}|$ value to avoid large direction perturbation noise. On the contrary, as ϵ increases, direction perturbation becomes less eminent, and, therefore, we can select a larger $|\mathcal{D}|$ value to enhance the utility more significantly.

We then evaluate how the size of $\mathcal{P}$ impacts the utility of the ATP mechanism. We conduct experiments on the datasets with thousands of locations, i.e., NYC and CLE datasets with 1,000 and 2,000 locations, respectively, following the previous study [14]. We take the most popular 10%, 25%, 50%, and 100% locations from them to evaluate the performance of our mechanism with privacy budget $\epsilon = 4$. We can observe in Figure 7a that the impact of a large $|\mathcal{P}|$ is not too significant. This is because the anchor-based trajectory region constraint mitigates the negative influence of a large $|\mathcal{P}|$.

Furthermore, we conduct experiments on the synthetic CPS dataset to assess the performance of our mechanisms under different tuning parameters. As with the previous study [14], we generate 4,000 trajectories with different fixed lengths ranging from 3 to 8 and different reach bounds between each two points in a trajectory with different travel speeds from 1 to 16 km/hr, and with no reach constraint (i.e., ∞). As shown in Figures 7b and 7c, overall, ATP outperforms TP, and their performance is quite stable with different trajectory lengths or different reach bounds. Besides, we have an interesting observation that the NE of ATP even slightly decreases with the increasing trajectory length, which is probably because more locations contribute to the effect of direction restriction.

Finally, we validate our proposed direction granularity strategy. As aforementioned, we calculate the average probability of success to choose the correct discrete direction for different ranges and for each granularity $|\mathcal{D}|$ under different ϵ values. We set different radian intervals as $\Theta = \{\pi/2, \pi/4, \pi/6, \pi/12\}$, corresponding to different candidate granularities $|\mathcal{D}| \in \{2, 4, 6, 12\}$. In Table 1, ϵ denotes the total privacy budget, instead of the budget that is used to compute the probability for direction perturbation (i.e., three quarters of $\frac{3\epsilon'}{4}$ or $\frac{3\epsilon^*}{4}$, as described in Section 6.1.2). We can observe that the granularity chosen based on the average direction-preserving success probability almost always corresponds to one of the best granularities under different ϵ values in Figures 6c and 6f.

Table 2: Comparison of average count differences (ACDs) for different methods (ϵ=4).

	NYC	CHI	CLE	CPS
CGM	17.7344	11.4397	12.1424	73.446
NGRAM	13.5946	11.7533	10.2275	32.3889
TP	10.1774	7.1965	5.8968	25.8770
ATP	10.9542	7.4568	6.1147	28.4412

6.2.4 Evaluation on Practical Applications. We further consider hotspot preservation as a practical application to demonstrate the advantages of our mechanisms over the existing works. Since we do not assume time information in our setting, we evaluate the average count difference (ACD), a popular metric for hotspot preservation [14]. Given a set of the most popular locations, ACD calculates the average absolute difference of their counts before and after perturbation. The experiments are conducted on four datasets with privacy budget $\epsilon = 4$. In particular, for the NYC dataset, we evaluate the ACD of the top 50% of popular locations, while for the CHI, CLE, and CPS datasets with less locations than NYC, we evaluate the ACD of their top 75% of popular locations. As shown in Table 2, we can observe that both TP and ATP mechanisms achieve significantly lower ACD than NGRAM and CGM on all datasets, which again confirms the superiority of our proposed methods. Without additional external knowledge, the unbounded noise will largely impact the performance of NGRAM and CGM. TP and ATP achieve similar performance. TP can obtain slightly lower ACD than ATP. We believe this is because ATP restricts the perturbation region of a trajectory, which may lead to a slightly more skewed hotspot count distribution. Nevertheless, ATP is still the best choice for a data collector to perform a wide range of analysis tasks.

7 CONCLUSION

In this paper, we propose a novel private trajectory data collection mechanism called ATP that satisfies pure ϵ-LDP. We model trajectory perturbation as a two-stage process, which first estimates discrete directions and then performs point perturbations. The proposed mechanism is further enhanced by adaptively restricting the trajectory region. To the best of our knowledge, our solution is the first to combine direction information with the perturbation of a trajectory under LDP. We also provide theoretical utility analysis of the region size and a guideline to choose a suitable direction granularity based on the given privacy budget. From the experimental results, we can observe an interesting correlation between the underlying distribution of the location points (e.g., their densities in different regions) and the direction granularity selection process. We believe exploring such a correlation is a valuable direction for future work.

ACKNOWLEDGMENTS

This work was supported by the National Key R&D Program of China (Grant No. 2020YFB1710200), the National Natural Science Foundation of China (Grant No. 62072136, 62102334, 62072390, and 92270123), and the Research Grants Council, Hong Kong SAR, China (Grant No. 15218919, 15203120, 15226221, 15225921, 15209922, and C2004-21GF).

REFERENCES

[1] Miguel E Andrés, Nicolás E Bordenabe, Konstantinos Chatzikokolakis, and Catuscia Palamidessi. 2013. Geo-Indistinguishability: Differential Privacy for Location-Based Systems. In *Proceedings of the 20th ACM SIGSAC Conference on Computer & Communications Security (CCS)*. 901–914.

[2] D Apple. 2017. Learning with Privacy at Scale. In *Apple Machine Learning Journal*.

[3] Ergute Bao, Yin David Yang, X. Xiao, and Bolin Ding. 2021. CGM: An Enhanced Mechanism for Streaming Data Collectionwith Local Differential Privacy. In *Proceedings of the VLDB Endowment (PVLDB)*, Vol. 14. 2258–2270.

[4] Dan Boneh, Elette Boyle, Henry Corrigan-Gibbs, Niv Gilboa, and Yuval Ishai. 2021. Lightweight Techniques for Private Heavy Hitters. In *Proceedings of 42nd IEEE Symposium on Security and Privacy (S&P)*. 762–776.

[5] Nicolás E Bordenabe, Konstantinos Chatzikokolakis, and Catuscia Palamidessi. 2014. Optimal Geo-Indistinguishable Mechanisms for Location Privacy. In *Proceedings of the 21st ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 251–262.

[6] Konstantinos Chatzikokolakis, Miguel E Andrés, Nicolás E Bordenabe, and Catuscia Palamidessi. 2013. Broadening the Scope of Differential Privacy Using Metrics. In *Proceedings of the 13th International Symposium on Privacy Enhancing Technologies Symposium (PETS)*. 82–102.

[7] Rui Chen, Benjamin CM Fung, Bipin C Desai, and Nériah M Sossou. 2012. Differentially Private Transit Data Publication: A Case Study on the Montreal Transportation System. In *Proceedings of the 18th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (SIGKDD)*. 213–221.

[8] Rui Chen, Haoran Li, A Kai Qin, Shiva Prasad Kasiviswanathan, and Hongxia Jin. 2016. Private Spatial Data Aggregation in the Local Setting. In *Proceedings of the IEEE 32nd International Conference on Data Engineering (ICDE)*. 289–300.

[9] Rui Chen, Qian Xiao, Yu Zhang, and Jianliang Xu. 2015. Differentially Private High-Dimensional Data Publication via Sampling-Based Inference. In *Proceedings of the 21st ACM SIGKDD Conference on Knowledge Discovery and Data Mining (SIGKDD)*. 129–138.

[10] Xiang Cheng, Sen Su, Shengzhi Xu, Li Xiong, Ke Xiao, and Mingxing Zhao. 2018. A Two-Phase Algorithm for Differentially Private Frequent Subgraph Mining. In *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, Vol. 30. 1411–1425.

[11] Xiang Cheng, Peng Tang, Sen Su, Rui Chen, Zequn Wu, and Binyuan Zhu. 2020. Multi-Party High-Dimensional Data Publishing Under Differential Privacy. In *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, Vol. 32. 1557–1571.

[12] Eunjoon Cho, Seth A Myers, and Jure Leskovec. 2011. Friendship and Mobility: User Movement in Location-Based Social Networks. In *Proceedings of the 17th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (SIGKDD)*. 1082–1090.

[13] Graham Cormode, Somesh Jha, Tejas Kulkarni, Ninghui Li, Divesh Srivastava, and Tianhao Wang. 2018. Privacy at Scale: Local Differential Privacy in Practice. In *Proceedings of the 44th ACM Conference on Management of Data (SIGMOD)*. 1655–1658.

[14] Teddy Cunningham, Graham Cormode, Hakan Ferhatosmanoglu, and Divesh Srivastava. 2021. Real-World Trajectory Sharing with Local Differential Privacy. In *Proceedings of the VLDB Endowment (PVLDB)*, Vol. 14. 2283–2295.

[15] Bolin Ding, Janardhan Kulkarni, and Sergey Yekhanin. 2017. Collecting Telemetry Data Privately. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*. 3574–3583.

[16] Jiawei Duan, Qingqing Ye, and Haibo Hu. 2022. Utility Analysis and Enhancement of LDP Mechanisms in High-Dimensional Space. In *Proceedings of the IEEE 38th International Conference on Data Engineering (ICDE)*. 407–419.

[17] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating Noise to Sensitivity in Private Data Analysis. In *Proceedings of the 3rd Conference on Theory of Cryptography (TCC)*. 265–284.

[18] Cynthia Dwork, Aaron Roth, et al. 2014. The Algorithmic Foundations of Differential Privacy. In *Foundations and Trends® in Theoretical Computer Science (TCS)*, Vol. 9. 211–407.

[19] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. 2014. Rappor: Randomized Aggregatable Privacy-Preserving Ordinal Response. In *Proceedings of the 21st ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 1054–1067.

[20] Marta C Gonzalez, Cesar A Hidalgo, and Albert-Laszlo Barabasi. 2008. Understanding Individual Human Mobility Patterns. In *Nature*, Vol. 453. 779–782.

[21] Mehmet E Gursoy, Ling Liu, Stacey Truex, and Lei Yu. 2018. Differentially Private and Utility Preserving Publication of Trajectory Data. In *IEEE Transactions on Mobile Computing (TMC)*, Vol. 18. 2315–2329.

[22] Mehmet E Gursoy, Vivekanand Rajasekar, and Ling Liu. 2020. Utility-Optimized Synthesis of Differentially Private Location Traces. In *Proceedings of the 2nd IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA)*. 30–39.

[23] Xi He, Graham Cormode, Ashwin Machanavajjhala, Cecilia M Procopiuc, and Divesh Srivastava. 2015. DPT: Differentially Private Trajectory Synthesis Using Hierarchical Reference Systems. In *Proceedings of the VLDB Endowment (PVLDB)*, Vol. 8. 1154–1165.

[24] Kaifeng Jiang, Dongxu Shao, Stéphane Bressan, Thomas Kister, and Kian-Lee Tan. 2013. Publishing Trajectories with Differential Privacy Guarantees. In *Proceedings of the 25th International Conference on Scientific and Statistical Database Management (SSDBM)*. 1–12.

[25] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. 2014. Extremal Mechanisms for Local Differential Privacy. In *Proceedings of the 27th International Conference on Neural Information Processing Systems (NeurIPS)*. 2879–2887.

[26] Shiva Prasad Kasiviswanathan, Homin K Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. 2011. What Can We Learn Privately?. In *SIAM Journal on Computing (SICOMP)*, Vol. 40. 793–826.

[27] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. 2006. t-Closeness: Privacy Beyond k-Anonymity and l-Diversity. In *Proceedings of the IEEE 23rd International Conference on Data Engineering (ICDE)*. 106–115.

[28] Zitao Li, Tianhao Wang, Milan Lopuhaä-Zwakenberg, Ninghui Li, and Boris Škoric. 2020. Estimating Numerical Distributions under Local Differential Privacy. In *Proceedings of the 46th ACM Conference on Management of Data (SIGMOD)*. 621–635.

[29] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. 2007. l-Diversity: Privacy Beyond k-Anonymity. In *ACM Transactions on Knowledge Discovery from Data (TKDD)*, Vol. 1. 3–es.

[30] Frank McSherry and Kunal Talwar. 2007. Mechanism Design via Differential Privacy. In *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. 94–103.

[31] Zhan Qin, Yin Yang, Ting Yu, Issa Khalil, Xiaokui Xiao, and Kui Ren. 2016. Heavy Hitter Estimation over Set-Valued Data with Local Differential Privacy. In *Proceedings of the 23rd ACM SIGSAC Conference on Computer and Communications Security (CCS)*. 192–203.

[32] Pierangela Samarati. 2001. Protecting Pespondents Identities in Microdata Release. In *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, Vol. 13. 1010–1027.

[33] Sen Su, Peng Tang, Xiang Cheng, Rui Chen, and Zequn Wu. 2016. Differentially Private Multi-Party High-Dimensional Data Publishing. In *Proceedings of the IEEE 32nd International Conference on Data Engineering (ICDE)*. 205–216.

[34] Xinyue Sun, Qingqing Ye, Haibo Hu, Yuandong Wang, Kai Huang, Tianyu Wo, and Jie Xu. 2023. Synthesizing Realistic Trajectory Data With Differential Privacy. In *IEEE Transactions on Intelligent Transportation Systems (TITS)*, Vol. 24. 5502–5515.

[35] Latanya Sweeney. 2002. k-Anonymity: A Model for Protecting Privacy. In *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems (IJUFKS)*, Vol. 10. 557–570.

[36] Shun Takagi, Yang Cao, Yasuhito Asano, and Masatoshi Yoshikawa. 2019. Geo-Graph-Indistinguishability: Protecting Location Privacy for LBS over Road Networks. In *Proceedings of the 33rd IFIP Annual Conference on Data and Applications Security and Privacy (DBSec)*. 143–163.

[37] Waldo R Tobler. 1970. A Computer Movie Simulating Urban Growth in the Detroit Region. In *Economic Geography*, Vol. 46. 234–240.

[38] Ning Wang, Xiaokui Xiao, Yin Yang, Jun Zhao, Siu Cheung Hui, Hyejin Shin, Junbum Shin, and Ge Yu. 2019. Collecting and Analyzing Multidimensional Data with Local Differential Privacy. In *Proceedings of the IEEE 35th International Conference on Data Engineering (ICDE)*. 638–649.

[39] Tianhao Wang, Jeremiah Blocki, Ninghui Li, and Somesh Jha. 2017. Locally Differentially Private Protocols for Frequency Estimation. In *Proceedings of the 26th USENIX Security Symposium (USENIX Security)*. 729–745.

[40] Stanley L Warner. 1965. Randomized Response: A Survey Technique for Eliminating Evasive Answer Bias. In *Journal of the American Statistical Association (JASA)*, Vol. 60. 63–69.

[41] Shengzhi Xu, Sen Su, Li Xiong, Xiang Cheng, and Ke Xiao. 2016. Differentially Private Frequent Subgraph Mining. In *Proceedings of the IEEE 32nd International Conference on Data Engineering (ICDE)*. 229–240.

[42] Qiao Xue, Qingqing Ye, Haibo Hu, Youwen Zhu, and Jian Wang. 2023. DDRM: A Continual Frequency Estimation Mechanism with Local Differential Privacy. In *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, Vol. 35. 6784–6797.

[43] Dingqi Yang, Daqing Zhang, Vincent W Zheng, and Zhiyong Yu. 2014. Modeling User Activity Preference by Leveraging User Spatial Temporal Characteristics in LBSNs. In *IEEE Transactions on Systems, Man, and Cybernetics: Systems (TSMC)*, Vol. 45. 129–142.

[44] Jianyu Yang, Xiang Cheng, Sen Su, Huizhong Sun, and Changju Chen. 2022. Collecting Individual Trajectories under Local Differential Privacy. In *Proceedings of the 23rd IEEE International Conference on Mobile Data Management (MDM)*. 99–108.

[45] Lin Yao, Zhenyu Chen, Haibo Hu, Guowei Wu, and Bin Wu. 2022. Privacy Preservation for Trajectory Publication Based on Differential Privacy. In *ACM Transactions on Intelligent Systems and Technology (TIST)*, Vol. 13. 1–21.

[46] Qingqing Ye, Haibo Hu, Man Ho Au, Xiaofeng Meng, and Xiaokui Xiao. 2022. LF-GDPR: A Framework for Estimating Graph Metrics with Local Differential Privacy. In *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, Vol. 34. 4905–4920.

[47] Qingqing Ye, Haibo Hu, Xiaofeng Meng, and Huadi Zheng. 2019. PrivKV: Key-Value Data Collection with Local Differential Privacy. In *Proceedings of the 40th IEEE Symposium on Security and Privacy (SP)*. 317–331.

[48] Lei Yu, Ling Liu, and Calton Pu. 2017. Dynamic Differential Location Privacy with Personalized Error Bounds. In *Proceedings of the 24th Annual Network and Distributed System Security Symposium (NDSS)*.

[49] Jing Yuan, Yu Zheng, Liuhang Zhang, Xing Xie, and Guangzhong Sun. 2011. Where to Find My Next Passenger. In *Proceedings of the 13th International Conference on Ubiquitous Computing (UbiComp)*. 109–118.

[50] Jun Zhang, Graham Cormode, Cecilia M Procopiuc, Divesh Srivastava, and Xiaokui Xiao. 2017. Privbayes: Private Data Release via Bayesian Networks. In *ACM Transactions on Database Systems (TODS)*, Vol. 42. 1–41.