An Analysis of Differential Privacy Research in Location and Trajectory Data

Fatima Zahra Errounda
Concordia University - Sir George Williams Campus: Concordia University
Yan Liu (yan.liu@concordia.ca)
Concordia University https://orcid.org/0000-0002-6747-8151

Survey paper

Keywords: Location, Trajectory, Differential privacy

Posted Date: October 22nd, 2020

DOI: https://doi.org/10.21203/rs.3.rs-94765/v1

RESEARCH

An Analysis of Differential Privacy Research in Location and Trajectory Data

Fatima Zahra Errounda and Yan Liu*

*Correspondence:
yan.liu@concordia.ca
Department of Electrical and Computer Engineering, Concordia University, Montreal, Canada
Full list of author information is available at the end of the article

Abstract

Location and trajectory data are routinely collected to generate valuable knowledge about users' pattern behavior. However, releasing location data may jeopardize the privacy of the involved individuals. Differential privacy is a powerful technique that prevents an adversary from inferring the presence or absence of an individual in the original data solely based on the observed data. The first challenge in applying differential privacy in location is that a it usually involves a single user. This shifts the adversary's target to the user's locations instead of presence or absence in the original data. The second challenge is that the inherent correlation between location data, due to people's movement regularity and predictability, gives the adversary an advantage in inferring information about individuals. In this paper, we review the differentially private approaches to tackle these challenges. Our goal is to help newcomers to the field to better understand the state-of-the art by providing a research map that highlights the different challenges in designing differentially private frameworks that tackle the characteristics of location data. We find that in protecting an individual's location privacy, the attention of differential privacy mechanisms shifts to preventing the adversary from inferring the original location based on the observed one. Moreover, we find that the privacy-preserving mechanisms make use of the predictability and regularity of users' movements to design and protect the users' privacy in trajectory data. Finally, we explore how well the presented frameworks succeed in protecting users' locations and trajectories against well-known privacy attacks.

Keywords: Location; Trajectory; Differential privacy

Introduction

With the help of sensing technology advancement and the popularity of location-based applications, location data is becoming ubiquitous. Global positioning systems (GPS), wireless sensor networks (WSN), and social media geo-tagged locations generate daily data that covers millions of peoples and large distances. Typically, service providers collect users location data then store, analyze, and share it with other third parties. The collected data is characterized by the 5Vs (Volume, Velocity, Variety, Value and Veracity) of big data.

Given the tremendous business opportunities, it is not a surprise that big companies such as Google, Apple, Microsoft, and Facebook are interested in users' locations. However, location sharing comes with privacy risks that might lead to stalking, fraud, or kidnapping. Companies in the US are legally required to disclose their data collection and usage practices, and can be fined in case of compliance

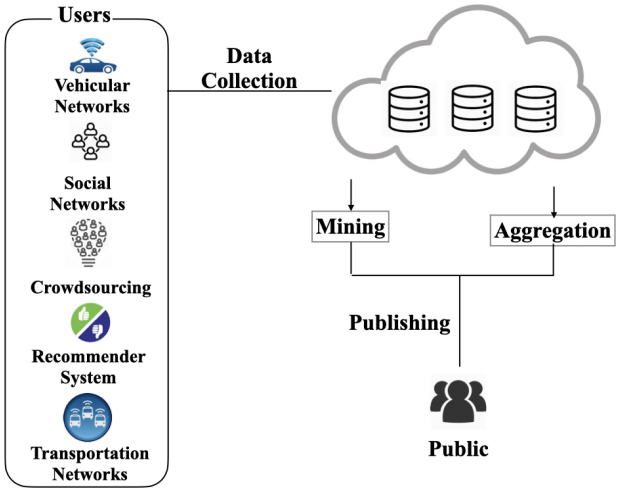


Figure 1: Location and trajectory data flow: analysis is performed on the data collected from end users, including aggregation and mining. Privacy risks are present in every step.

failure [1]. And recently, Arizona sues Google over allegations it illegally tracked Android smartphone users' locations even after users shut off location services [2].

Interest in location privacy is very high, and there are many techniques to achieve it. Traditional location privacy include anonymity [3], data obfuscation [4], computational Private Information Retrieval (PIR) [5]. However, many attacks point out these privacy techniques weaknesses. For example, anonymity is vulnerable to background and inference attacks where the adversary analyzes data in order to gain knowledge about an individual [6] (the user's location/identity/other sensitive information). Obfuscation techniques are susceptible to localization attacks where the adversary estimates the true user's location based on the obfuscated location [7]. PIR techniques may fall short against access pattern attacks where the adversary can intercept a user's query and response pairs and combine these access patterns with background knowledge to infer the user's identity [8].

Differential privacy [9] is de-facto technique for private data sharing. It quantifies the ability of an adversary from differentiating between two observed representations of the true data that differ in the presence or absence of a single individual. This concept is defined as indistinguishability. Therefore differential privacy hinders a powerful adversary from inferring private information solely from the private data representation (obfuscated data, aggregates...). Indistinguishability is usually achieved by carefully adding noise to the private data before sharing it with untrusted entities. However, a straight forward application of differential privacy to location data is not always possible. In many location-based applications, the adversary might observe the location data at the granularity of individuals, which requires a revisit of the concept of indistinguishability. This leads us to our first research question:

RQ1: How to design differentially private solutions in the context of protecting data with a single individual?

Another aspect of location data is its high predictability compared to general tabular data [10]. In fact, studies on human mobility show that people spend the majority of their time in a small set of important locations [11]. Consequently transitions between important locations are highly regular. This motivates the following research question:

RQ2: How does differential privacy protect the trajectory of a single user while taking into account his/her movement predictability?

Several privacy attacks against location information are studied and categorized depending on the targeted information (single position, identity,...), the adversary's background information (contextual or temporal information), and the privacy model (trusted vs untrusted location service provider, trusted vs untrusted users) [12]. In fact, even when the data is anonymized, meaning that the users identities are hidden, it might lead to privacy breach where users are re-identified from anonymized data [13]. Also, it is possible to exploit the uniqueness and regularity of human mobility to recover individual's trajectories from the aggregated mobility data without any prior knowledge [14]. Naturally, the following research question needs to be asked:

RQ3: How effective is differential privacy in protect against well-known location and trajectory attacks?

In this paper, we aim at answering these research questions. We conduct an investigation to understand the challenges in applying differential privacy methods on location data and provide directions for researchers to select appropriate methods for specific problems. More specifically, our **contributions** are:

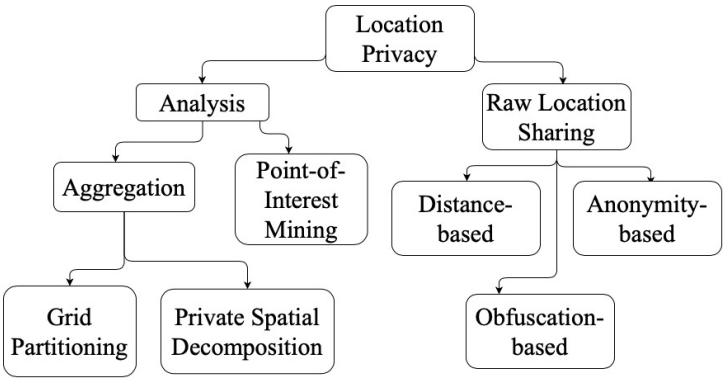
1) Provide a holistic status report on the adoption of differential privacy to location data.

2) Identify the challenges faced during the different stages of the data flow.

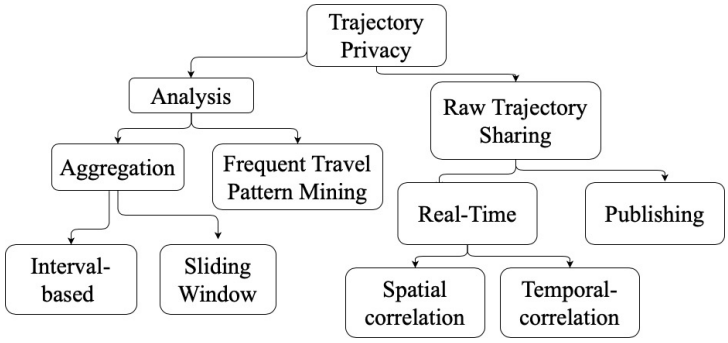
3) Compare the advantages and disadvantage of each solution.

4) Identify research gaps and offer future research directions.

In our previous work [15], we covered only the location privacy aspect. Here, we extend our work to trajectory privacy as well. The rest of the paper is organized as follow: Section 1 introduces the review method adopted in the paper, the basic concepts of location and trajectory data, and differential privacy. We review the differential privacy solutions for location data in Section 2 following the taxonomy illustrated in Fig.2a. Section 3 presents the trajectory data privacy as detailed in Fig.2b. Section 4 presents the future work deduced from the reviewed solutions. Finally Section 5 concludes the paper.



(a) Location privacy taxonomy



(b) Trajectory privacy taxonomy

Figure 2: The categorization taxonomy based on the location and trajectory data flow

1 Background

In this section, we introduce the basics of location and trajectory privacy and differential privacy.

1.1 Location and Trajectory Privacy

A location is usually represented as the tuple *¡identity; position; time¿* including the user's identity, the spatial information (position), and the temporal information (time) [16]. In some domains, such as location-based services, the user's service queries are also part of his/her location data.

Single locations may be scattered and may not correlate with each other. On the other hand, a trajectory is a group of locations with strong correlations [16]. In fact, a trajectory is the path made by the moving entity through the space where it moves. The path is never made instantly but requires a certain amount of time. Therefore, time is an inseparable aspect of a trajectory.

From a privacy perspective, we adopt Georgiadou et al. [1] location privacy definition:

> "**Location privacy** is the right to control the collection, access, recording, and usage of an individual's (location) information and determine when, how, and to what extent it is processed by others"

This privacy definition is inclusive of location and trajectory data. We further differentiate between location privacy, where a single user location is considered in isolation of the other user's locations, and trajectory privacy which covers multiple sequential locations.

1.2 Differential Privacy

Differential privacy [17] is originally formulated in the context of statistical databases. It is based on the concept of neighbourhood defined as follows:

Definition 1 (Neighbourhood) Two datasets $\mathcal{D}$ and $\mathcal{D}'$ are neighbours if they differ in at most one element. Meaning that one dataset is a subset of the other and the larger dataset contains exactly one additional row.

A randomization mechanism $\mathcal{M}$ satisfies ϵ-differential privacy if the addition or removal of a single element in the dataset $\mathcal{D}$ does not change the probability of the mechanism outcome by more than some small amount (quantified using the privacy budget ϵ). More formally:

Definition 2 (Differential Privacy) [17] A randomization mechanism $\mathcal{M}$ gives ϵ-differential privacy for all neighbouring datasets $\mathcal{D}$ and $\mathcal{D}'$ and for every set of outputs $\mathcal{O} \in Range(\mathcal{M})$, if $\mathcal{M}$ satisfies:

$$Pr[\mathcal{M}(\mathcal{D}) \in \mathcal{O}] \leq e^{\epsilon} \times Pr[\mathcal{M}(\mathcal{D}') \in \mathcal{O}] \tag{1}$$

1.2.1 Popular Mechanisms

To achieve differential privacy, two popular mechanisms are proposed in [17]: Laplace and exponential. They perturb the original query q result using random noise that is calibrated with the privacy budget ϵ and the global sensitivity Δq defined as below [18]:

Definition 3 (Global Sensitivity) The global sensitivity of a query $q : \mathcal{D} \rightarrow R$ is:

$$\Delta q = \max_{\mathcal{D},\mathcal{D}'} \parallel q(\mathcal{D}) - q(\mathcal{D}') \parallel_1 \tag{2}$$

for all neighbouring $\mathcal{D}$ and $\mathcal{D}'$.

Laplace mechanism is based on the Laplace distribution: $Lap(x \mid b) = \frac{1}{2b} e^{-\frac{|x|}{b}}$ where b is the scale factor. The mechanism uses a scale factor of $\frac{\Delta q}{\epsilon}$.

Laplace mechanism adds noises to the real output using Laplace distribution as follows:

Definition 4 (Laplace Mechanism) For a query q, a mechanism $\mathcal{M}(x) = q(x) + Lap(\frac{\Delta q}{\epsilon})$ satisfies ϵ-differential privacy

In many applications, the query q may map datasets to strings, strategies, or trees, which makes the addition of noise no longer sensible. McSherry and Talwar [19] propose the exponential mechanism to tackle this issue. First, it defines a score function $u : \mathcal{D} \times R \rightarrow \mathbb{R}$ that gives a real valued score to each possible output of the query function. Then, the exponential mechanism selects an output with a probability proportional to the sensitivity of the score function as follows: $e^{\left(\frac{\epsilon u(d,r)}{2\Delta u}\right)}$.

Another algorithm is introduced to allow users to perturb their data locally before sharing it with a data curator. This is especially useful in the case where the data curator is not trusted. The randomized response algorithm allows the user responds either truthfully or the opposite answer to a sensitive question depending on a coin flip. It is commonly used in surveys of people's "yes or no" opinions about a sensitive question. Because the user gives a randomized response, the surveyors cannot determine the individual's true answer, but can still extract useful statistics [20].

1.2.2 Properties

Differential privacy is a powerful technique that is immune to post-processing. It is important to note that an adversary cannot increase her knowledge about the true data of an individual using the output of the privacy mechanism. That is, an adversary cannot combine a data-independent mapping f with a ϵ-differentially private mechanism $\mathcal{M}$ to gain more knowledge. In other words $f \circ \mathcal{M}$ is also a ϵ-differentially private mechanism.

Several differentially private mechanisms can be combined in processing the true data. McSherry et al. [21] introduce two types of mechanism compositions: sequential and parallel. When a sequence of computations provides differential privacy in isolation, the final privacy guarantee is said to be the sum of each ϵ-differential privacy. On the other hand, when the input data is partitioned into disjoint sets, independent of the original data, the composition is said to be parallel and the final privacy depends on the worst computation guarantee of the sequence. The two compositions are formalized as follows:

Theorem 1 (Sequential Composition) *Consider mechanisms $\mathcal{M}_i$ that provide ϵ_i-differential privacy. A sequence of $\mathcal{M}_i$ over a dataset $\mathcal{D}$ provides $\Sigma\epsilon_i$-differential privacy.*

Theorem 2 (Parallel Composition) *Consider mechanisms $\mathcal{M}_i$ that provide ϵ-differential privacy. A sequence of $\mathcal{M}_i$ over a set of disjoint datasets $\mathcal{D}_i$ provides ϵ-differential privacy.*

User-level privacy and event-level privacy are two notions proposed for continual differential privacy application, such as data streams [9]. Event-level privacy protects the user at any single event occurring in the data stream. While, user-level privacy protects the user within the entire data stream.

2 Location Privacy

As stated above, location privacy is concerned with a single user's location in isolation of the other user's locations. In this section, we review the real-time sharing of a user's location that occur during the collection phase of the data flow. We also describe the different approaches during the location analysis for aggregation and mining.

2.1 Raw Location Sharing

In this setting, the data curator is an untrusted entity, therefore differential privacy is applied on every user's location before it reaches the data curator. Let us consider x and x' as the user's true location and the location shared with the data curator respectively. The goal here is to prevent an adversary from accurately guessing x, based on the observed x'. This is achieved using a differentially private mechanism $\mathcal{M}$ as illustrated in Fig.3. We summarize the methods to achieve this goal into three categories: distance-based, obfuscation-based, and anonymity-based methods.

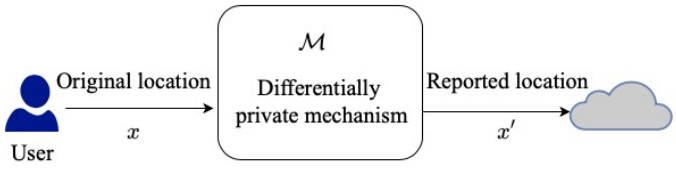


Figure 3: Differentially private mechanisms for raw location sharing report a private location x' after processing the true location x

2.1.1 Distance-based Method

Distance-based methods ([22], [23]) are based on the idea that farther apart locations should have less privacy influence on each other. It means that through perturbation any two locations within a given distance produce observations with similar distributions. Thus adversaries have no way to learn the user' true location. The distance between these locations is used to relax the guarantee of indistinguishability (Definition 2). This is formalized as follows:
For a privacy mechanism $\mathcal{M}$, the following holds true:

$$\mathcal{M}(x) \leq e^{\epsilon d_{\mathcal{X}}(x,x')}\mathcal{M}(x')) \tag{3}$$

where ϵ is the privacy budget, $\mathcal{X}$ is the location space, and $d_{\mathcal{X}}$ is a distance metric.

When the distance is fixed to a radius r, this definition is coined geo-indistinguishability [23]. In other words, for a fixed radius r, the location owner enjoys a privacy guarantee of ϵr, and an observer cannot distinguish between two originating locations within r based on the observed location. To achieve geo-indistinguishability, a privacy mechanism must output locations with a probability that decreases exponentially as the points get further from the owner's true location. The planar Laplacian distribution D_ϵ centred at x satisfies geo-indistinguishability as follows:

$$D_\epsilon(r,\theta) = \frac{\epsilon^2}{2\pi} r e^{-\epsilon r} \tag{4}$$

where r and θ are two random variable that represent the distance and the angle between the reported location x' and the true location x respectively, and ϵ is the privacy budget.

Therefore, to calculate a private location x', it suffices to draw a point (r, θ) from $D_\epsilon(r, \theta)$. Since r and θ are independent, the Equation 4 can be rewritten as:

$$D_\epsilon(r,\theta) = \underbrace{\int_0^{2\pi} D_\epsilon(r,\theta)d\theta}_{D_{\epsilon,\Theta}} \underbrace{\int_0^{\infty} D_\epsilon(r,\theta)dr}_{D_{\epsilon,R}}$$
$$D_{\epsilon,\Theta} = \frac{1}{2\pi}$$
$$D_{\epsilon,R} = \epsilon^2 r e^{-\epsilon r}$$

where $D_{\epsilon,\Theta}$ and $D_{\epsilon,R}$ correspond to the uniform distribution between [0,2π), and the Gamma distribution with shape 2 and scale $1/\epsilon$ respectively.

Geo-Indistinguishability presents an intuitive way to incorporate location characteristics into differential privacy by calibrating the degree of indistinguishability between locations based on their proximity. Therefore it attracts the interest of researchers generating a rich body of literature ([24], [25], [26], [27], [28], [29], [26], [27], and [28]). For example, a new randomization algorithm is proposed in [30] based on linear programming techniques to release locations with better utility than the planar Laplacian mechanism. Utility is defined in terms of the expected distance between the real location and the reported location.

Discussion: Distance-based methods answer the research question **RQ1** by shifting the indistinguishability to the observed locations of a single user instead of the presence or absence of the user. However in continuous location sharing, the adversary can use the correlation or closeness between locations to successfully run an inference attack such as shown in [31], where the user's points-of-interests were predicted with reasonable precision based on the reported noisy locations. Moreover, in [32], the adversary manages to infer the probability distribution used for location perturbation based on the released locations. Therefore, the method may fall short in protecting the user against popular privacy attack (**RQ3**). Finally, since the inference privacy attack in [31] uses the correlation between consecutive locations, we also conclude that the distance-based methods require additional improvement to be used in protecting a trajectory of a single user (**RQ2**).

2.1.2 Obfuscation-based Method

Data obfuscation is a privacy technique that aims at lowering the sensitive information accuracy in a systematic, controlled, and statistically rigorous way while permitting sufficient calculation accuracy [4]. This concept translates in location data as sharing an approximation of the true location [33]. Unlike the distance-based method, there doesn't have to be a distance correlation between the true and the shared locations, x and x' respectively.

Kim et al. [34] use the randomized response algorithm to achieve location obfuscation and satisfy differential privacy. Let O be a set of predefined locations. The true location x is expressed as a vector of binary values where each index corresponds

to a known location in O and the bit value shows the presence or absence (1 or 0 respectively) of the user at the indexed location. For example, this representation may be used in indoor positioning data, where installed devices in fixed locations send signals to users, and depending on the signal's strength the user identifies her current indoor positioning. Then, each user locally applies the randomized response mechanism (as described in Section 1) on every bit of his/her location vector x before sharing it with a data curator. In this case, the privacy guarantee is expressed as follows:
For all pairs of a user' locations x_1 and x_2:

$$\frac{P(\mathcal{M}(x_1) = x')}{P(\mathcal{M}(x_2) = x')} = e^{\epsilon} \tag{5}$$

where $\mathcal{M}$ is the privacy mechanism, ϵ is the privacy budget, x' is the reported location, and O is the range of $\mathcal{M}$.

Wang et al. ([35] and [36]) define an obfuscation matrix that can minimize the expectation of data uncertainty between the true and obfuscated locations, x and x' respectively. The obfuscation matrix is maintained by the data curator, uploaded by the users, and used locally to obfuscate their locations. The obfuscation matrix encodes the probabilities of obfuscating any one region to another one. More specifically, the matrix entry $[i, j]$ refers to the probability of mapping the region i to the region j. Differential privacy is achieved by guaranteeing indistinguishability between the true location region and the obfuscated one.

Discussion: Similarly to distance-based methods, obfuscation-based methods shift the indistinguishability to the observed locations of a single user instead of the presence or absence of the user **RQ1**. The randomized response method works well when the set of possible locations is fixed and the user's movements are limited, such as in indoor positioning systems. However, it does not take into account the temporal correlation between the user's locations, which can lead to privacy breaches. For example, Chen et al. [37] showed that the correlation between locations may influence the success in inference attacks' accuracy **RQ3**. To mitigate this concern, the obfuscation matrix incorporates the inference error of the optimal attack given the public prior leakage of a user's location distribution. The inference attacks also show that the obfuscation-based methods require additional improvement to be used in protecting a trajectory of a single user (**RQ2**).

2.1.3 Anonymity-based Method

The concept of location k-anonymity is introduced in [3]. It uses cloaking algorithm that partitions the space into areas (cloaking regions) that include the locations of $k-1$ other users. This way the adversary cannot distinguish between the k users. Anonymity-based methods use differential privacy to make the cloaking regions indistinguishable. To achieve this an anonymizing algorithm that collects the locations of multiple users first generates candidate cloaking regions that contain at least k users. Then, the algorithm+ applies a randomization model to select the cloaking region to share as illustrated in Fig.**??**. Differential privacy is guaranteed between the set of cloaking regions as follows:

$$Pr(\mathcal{M}(c) \in Z) \leq e^{\epsilon} Pr(\mathcal{M}(c') \in Z) \tag{6}$$

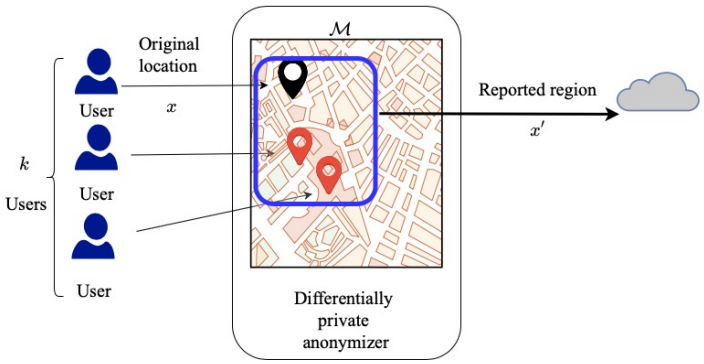
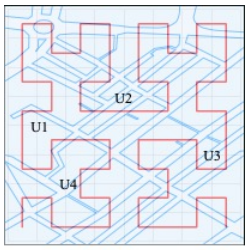


Figure 4: Differentially private anonymizer for sharing locations

where c and c' are cloaking regions that contain at least k users, Z is the set of all possible cloaking regions, $\mathcal{M}$ is the privacy mechanism, and ϵ is the privacy budget.

Hilbert curve [38] is often used to build the closest cloaking regions [39]. It is a continuous space-filling curve that maps a multi-dimensional space to one dimension. For example, in the case of two-dimensional location data, each location is assigned an index reflecting the order of visit of the Hilbert curve in the two dimensional space as illustrated in Fig.5. The indices represent the locations in a singular dimensional space. The cloaking region is built by generating an area that includes k indices from the Hilbert curve. One advantage of Hilbert curve is that it preserves locality: locations close in the two-dimensional space remain close in the linear Hilbert ordering.

Hilbert curve can be used to build a set of candidate cloaking regions as adopted in ([40], [41], and [42], by rotating and shifting the Hilbert curve around the centre point of the data space. A randomization algorithm (such as the exponential mechanism) is used to selected from the cloaking regions candidates using a score function (such as the cloaking region size).



index	1	2	3	4
user	U4	U1	U2	U3

Figure 5: An example of a transformation from two dimensional data (a map with users U1, U2, U3, and U4) to a single dimensional data (indexed 1 to 4 for clarification purposes)

Discussion: By definition k-anonymity involves more than one user, therefore the research question **RQ1** is not relevant for the proposed methods. Location

k-anonymity is vulnerable to location linking attacks where the adversary continuously computes the intersections of multiple cloaking regions and infers the user's location [6]. However, Equation 6 guarantees that even in the case that the adversary knows the set of all possible cloaking regions, he cannot conclude the real cloaking area with high accuracy, as it is illustrated in [43] (**RQ3**). The temporal correlation between consecutive locations of users is not considered when generating or selecting the cloaking areas. Therefore anonymity-based methods also require additional improvement to be used in protecting a trajectory of a single user (**RQ2**).

2.2 Location Data Analysis

For location data analysis, users share their true locations with a trusted data curator. Then the latter uses the location data (i) to calculate aggregates, such as the number of locations in each of the cells of the location space, and (ii) to extract for user's behavioural movement patterns. We summarize these uses in this section.

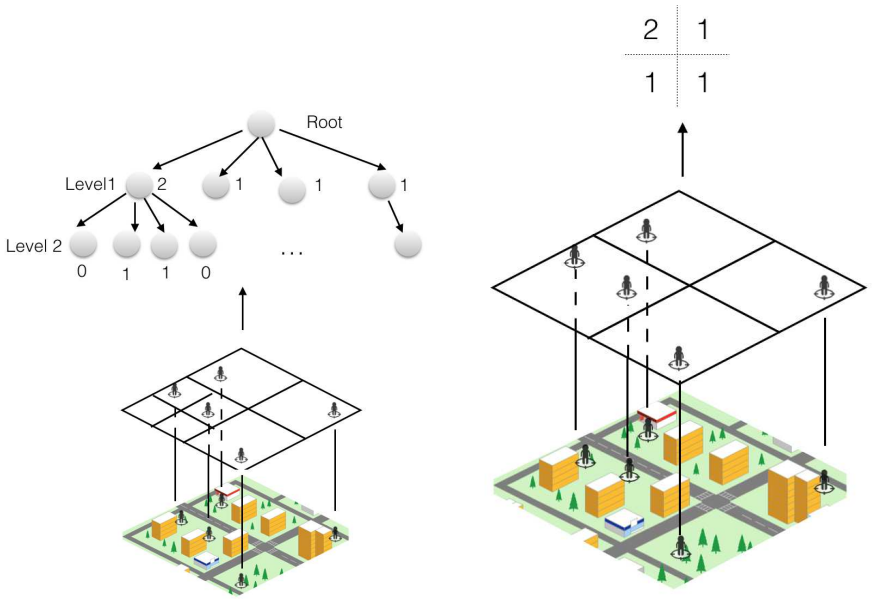
2.2.1 Location Aggregate Sharing

A common approach in location aggregation is to divide the location space in smaller areas (cells), calculate the number of users in each cell, and use this number for elaborate aggregates (such as means, medians) or to answer requests, such as range queries. The concept of neighbourhood is applied as introduced in Definition 1 and the global sensitivity (Definition 3) is calculated accordingly to calibrate the noise added to the users' counts in each cell. A challenge arise in designing a spatial division method that optimizes the aggregates' utility. For example, when a range query covers multiple cells, noise may quickly accumulate from the cells' counts reducing the query's utility. In this subsection we review two spatial division methods as illustrated in Fig.6.

Grid Partitioning One way to release the number of users located on a map is by laying a grid of equi-width cells over the location space, then counting the number of users in each cell. Laplace noise is then added to each count with sensitivity 1, since adding or removing a user changes the counts in each cell of the grid by a maximum of 1. The challenge then is to select the grid granularity that induces the lowest amount of noise and therefore assures utility. With the uniform grid partitioning presented in [44], the selected granularity is proportional to the total number of locations in the original data, the privacy budget ϵ, and a constant that reflects how uniformly the locations are distributed on the grid cells.

A challenge arise with the uniform grid approach when answering range queries that cover multiple cells fully or partially. Since the cumulative noise is proportional to the number of cells involved in answering the range query, it is important to avoid over-partitioning sparse areas or under-partitioning dense areas. An adaptive grid partitioning is presented in [44], where a second level of grid partitioning is done on denser areas. The granularity of the second partitioning depends on the noisy count of the dense areas. The total privacy budget ϵ is then distributed on the two levels of partitioning.

Crowdsourcing applications rely on knowing if any user is present at a specified area in order to assign him/her a task. This creates a constraint on the second level



(a) Private Spatial Decomposition (b) Grid partitioning

Figure 6: An example of the two spatial division approaches: (a) the space is divided using a quad-tree, the number of users present in each cell is recorded in the corresponding node (b) the space as divided using a grid of 2 by 2 and the corresponding number of users

of the grid partitioning where it should guarantee that at least one user is in a cell. To solve this issue, the work in [45], [46] propose a heuristic method to choose the granularity of the second level partitioning but only to the point where there is at least one user in a cell.

In addition to protecting the user's locations, the assigned tasks also present sensitive information in crowdsourcing. An adversary can potentially infer the user's location from tracking the accepted tasks, by linking the task's location to the user's. Therefore, privacy method presented in [47] splits the original locations of both tasks and workers in the crowdsourcing application into three-level grids to better represent sparse and dense areas.

Private Spatial Decomposition Spatial decomposition [48] divide the location space into smaller areas using a tree where each node represent a portion of the space and the tree leaves are the smallest cells in the space. The number of users is calculated at each node, including the leaves, and Laplace noise is added to generate private aggregates. Similarly to grid partitioning method, the global sensitivity has a value of 1. When answering a query, the tree is traversed to identify the nodes that correspond to the regions covered by the query and the sum of corresponding nodes' counts is calculated.

There are two main types of spatial decomposition: data independent and data dependent. Data independent decomposition does not consider the distribution of the locations in space. For example, quad-trees recursively split the location space

into four equal regions regardless of the number of locations in each region. An example of data dependent decomposition is the kd-tree where the space is recursively split via lines passing through a median calculated based on the location distribution. However, in the case of data-dependent trees, an adversary can use information, such as the median in case of kd-trees, to gain more knowledge about users' locations. Therefore, noise is also induced in the means to make the structure of the tree differentially private too.

Since the set of nodes of each level in the tree cover the whole space, the challenge is in distributing the privacy budget ϵ on the different levels of the tree. A uniform distribution of the privacy budget over the tree' levels is presented in [48]. However, the utility decreases as the tree depth increases. A geometric budget allocation strategy increases the budget geometrically at each tree level so the leaf counts have the highest accuracy. To improve the query's utility, it is also possible to use an unbalanced quadtree partition algorithm based on regional uniformity as adopted in [49]. Furthermore, the privacy budget allocation scheme is adjusted to ensure the effectiveness of the differential privacy model.

Discussion: Grid partitioning and private spatial decomposition are intuitive solutions to protect each user's privacy by hiding his/her presence or absence in the data (**RQ1**). However, the aggregates' utility differ in each solution. A comparison of different tree structures for private spatial decomposition shows that kd-tree performs best in skewed data (different levels of sparsity), while quadtree is better suited to uniform spatial distributions [50]. Both grid-partitioning and private spatial decomposition require adjustments to be used for trajectory aggregate privacy as illustrated in [51] **RQ2**. All the presented solutions rely on a trusted data curator that generates the private aggregates. This may lead to a vulnerability against the centralized privacy attack **RQ3**.

2.2.2 Mining point-of-interests

Differentially private mining is concerned with extending the current non-private mining algorithms to differentially private algorithms. Here, we focus on the most common mined information in location data, namely mining for points-of-interest.

Points-of-interests are characterized by two parameters: the encompassing geographic region of interest and the number of users' visits [52]. The challenge in private mining for points-of-interests is in the magnitude of the global sensitivity: the maximum difference in number of visits between neighbouring datasets can range from 0 to the maximum visit frequency of users. This hinders utility since a large global sensitivity results in utility loss due to the increased value of the added noise.

One way to reduce the sensitivity is by using spatial decomposition [48] to calculate local (smaller) sensitivity per geographic region as presented in [53], [54], and [55]. The local sensitivity is then used to calibrate the Laplace noise of the total number of visits per geographic region. Then a density-based spatial clustering (DBSCAN) [56] is used to identify the points-of-interests' encompassing regions. Finally Laplace noise is used to perturb the centroids of the regions of interests with a local sensitivity that is based on the maximal difference of all points in the encompassing region.

Another approach to reduce the global sensitivity is by limiting the contribution of any user's visits to a given location. It can be achieved by sampling the location data using a threshold l such that only l visits are retained per user as proposed in [57]. This threshold is then used as the upper bound of the global sensitivity.

Discussion: The presented approaches protect each user's privacy by hiding his/her presence or absence in the data (**RQ1**). However, they do not take into consideration the temporal correlation between users' visits (**RQ2**). Furthermore, limiting the number of location visits per user helps reduce the global sensitivity, however it has the disadvantage of distorting large counts. As a consequence, impacting the accuracy of the identified points-of-interests. The privacy attack conducted in [58] shows that by using a machine learning classifier trained on the adversary's prior knowledge (location aggregates), the adversary is capable of inferring an individual's membership in unseen location aggregate statistics **RQ3**.

3 Trajectory Privacy

As stated in Section 1, trajectory privacy is concerned with multiple sequential locations. In this section, we review the real-time sharing of a user's trajectory that occur during the collection phase of the data flow. We also describe the different approaches during the trajectory analysis for aggregation and mining.

3.1 Raw Trajectory Sharing

Sharing a user's trajectory is useful in many cases. For example, in traffic dispatching, users sequence of locations can support traffic's scheduling to avoid road congestion. In this section we review real-time trajectory sharing and trajectory publication.

3.1.1 Real-time Sharing

Real-time trajectory sharing methods make use of two aspects of location data: spatial and temporal correlation between locations. We categorize the methods following these two aspects.

Spatial-correlation Methods Similarly to location privacy methods presented in Section 2.1.1, spatial correlation methods use distance between the true and the released location to calibrate the degree of indistinguishability (Equation 3). However, the privacy guarantee decreases over the trajectory due to the repeated application of differential privacy in time (due to the sequential composition property Definition 1). As a consequence, the solutions reviewed here tackle this problem by reducing the need to consume the privacy budget ϵ at each location sharing.

R-tree is used to cache previously release private location in [59]. When sharing a location, the privacy mechanism first traverses the R-tree to find a previously released private location that satisfies the indistinguishability requirement. If such a location exists then it is released instead of calculating a new private location. This method optimizes the use of the privacy budget over the user's trajectory.

Another way to improve the consumption of the privacy budget is by making use of the predictability of users movements [60], [61]. At each timestamp, the privacy algorithm predicts a location based on the previously released ones. If the

predicted location is close enough to the user's true location, then it is released. Hence decreasing the privacy budget used for the trajectory. Otherwise, a portion of the privacy budget ϵ is used to generate a new private location.

Discussion: Similarly to raw location sharing methods, privacy is guaranteed by shifting the indistinguishability to the observed locations of a single user instead of the presence or absence of the user in the original data **RQ1**. Furthermore, the reviewed solutions take into account the user's movement predictability and regularity either by caching previously released locations or by predicting future locations **RQ2**. Even obfuscated location sharing discloses information which poses a risk to privacy as demonstrated in the mobility profile localization attack [7]. Furthermore, the study conducted in [62] explores the vulnerability of distance-based methods against this attack. It shows that the frequency of location sharing directly impacts the efficiency of the privacy guarantee. Moreover, the privacy attack presented in [63] shows that the co-location of users can disclose one's social connections, intimate partners, or more. Since the presented method considers a single user's trajectory, it might put the users at risk of the co-location privacy attack **RQ3**.

Temporal Correlation Methods Temporal correlation methods use the predictability of the user's movement to guarantee privacy. Let us consider the user's true location x_t at timestamp t. temporal correlation methods share a private location $x_t{}'$ calculated based on the temporal correlation between x_t and the true location x_{t-1} at timestamp $t-1$.

The solution presented in [64] and [65] build a set of all probabilistic locations S_t that a user may occupy at timestamp t with a given threshold α as follows:

$$S_t = min\{x_i | \sum_{x_i} p_t{}^-[i] \geq 1 - \alpha\} \tag{7}$$

where α is the probability threshold, x_i is a location in S_t, $p_t{}^-[i]$ is the prior probability of the user being at location x_i at timestamp t.

The true location x_t is assumed to be part of S_t set and correction measures are taken if not. The privacy mechanism reports one of the locations in S_t with ϵ-differential privacy guarantee. This is formalized as follows:

$$Pr[\mathcal{M}(x_1) = x_t{}'] \leq e^{\epsilon} \times Pr[\mathcal{M}(x_2) = x_t{}'] \tag{8}$$

where $\mathcal{M}$ is the privacy mechanism, x_1 and x_2 are any locations in S_t, ϵ is the privacy budget, and $x_t{}'$) is the reported location.

The privacy algorithm relies on the global sensitivity of the locations in the candidate set S_t. For example, if the global sensitivity is calculated in terms of the maximal distance between the locations in neighbouring candidate sets (Definition 3), the resulting value may be high and induce low utility. Therefore, the proposed solution transforms the locations in the S_t set to an isomorphic shape to reduce the sensitivity. Then a location is sampled from the resulting transformation using the newly calculated global sensitivity and shared as the private location $x_t{}'$.

Discussion: The proposed method achieves privacy while considering the locations of a single user by making his/her location indistinguishable from the point of view of an adversary **RQ1**. Moreover, the proposed solutions take into account the user's movement regularity by predicting the next location to protect the user's trajectory **RQ2**. Finally, the privacy attack presented in [63] shows that the co-location of users can disclose one's social connections, intimate partners, or more. Since the presented method considers a single user's trajectory, it might put the users at risk of the co-location privacy attack **RQ3**.

3.1.2 Raw Trajectory Publication

Raw trajectory publishing is useful in many situations. For instance, having access to high-quality trajectory data is the prerequisite for effective data mining. However, obfuscating each locations individually might result in unrealistic-looking trajectories such as the zigzagged shape illustrated in Fig.7. The methods presented in this section aim at guaranteeing users' privacy while maintaining the trajectories' utility.

When publishing a trajectory, the maximum distance between two consecutive points in the trajectory can be used to calibrate the added noise as the global sensitivity. This is adopted in [66] by bounding the global sensitivity value when protecting ships' trajectories. However, the resulting trajectories have zigzag shapes and many crossings and are unrepresentative of the original ones.

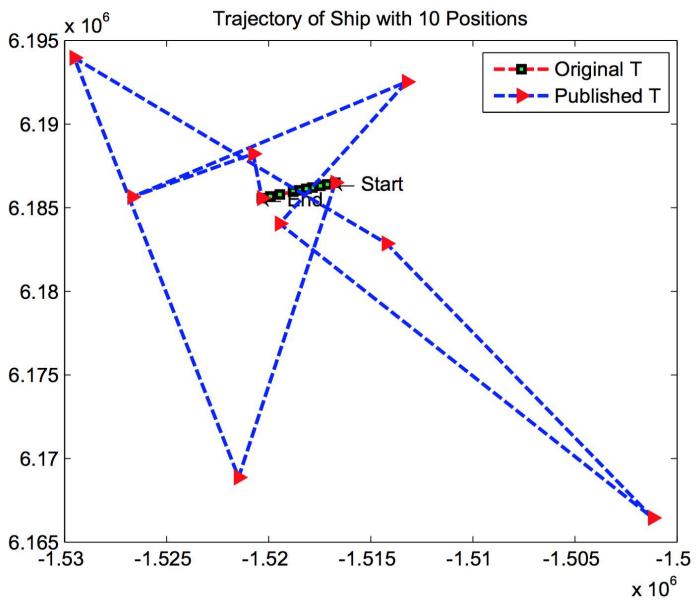


Figure 7: An example of zigzagged trajectory generated with Laplace noise [66]

To solve this problem, the presented solution samples locations to publish using the exponential mechanism by taking into account the direction and maximum speed of the ships. At each location, the angle and distance with the previous one

is calculated, then noise relative to these values is sampled and used to obfuscate the current location.

Another way to avoid generating unrealistically shaped private trajectories is to privately sample locations from the trajectory and interpolate between them to get a smooth trajectory. This is adopted in [67] by partitioning the trajectory into segments, then uniformly sampling locations from these segments, and finally interpolating the sampled locations with Bezier.

The method presented in [68] [69], and [70] aim at preserving the semantic significance of the perturbed location. It builds an obfuscation region around containing candidate locations that have similar semantic significance. Then, the exponential mechanism is used to sample a location from the obfuscation region.

Discussion: The presented methods guarantee a user's privacy by making every location in his/her trajectory indistinguishable from the point of view of an adversary (**RQ1**, **RQ2**). However, the attack presented in [71] shows that given a set of known trajectories and their pairwise distances to a private trajectory, it is possible to infer the private trajectory with high confidence. By obfuscating individual locations in the trajectory, noise is induced into the pairwise distances between trajectories, which may reduce the success of this privacy attack [71]. Another privacy attack is possible by considering the location-correlated information between users and background knowledge attack to obtain the user's private trajectory as shown in [72]. A solution to neutralize such attacks is presented [73] **RQ3**.

3.2 Trajectory Data Analysis

For trajectory data analysis, users share their true trajectories with a trusted data curator. Then the latter uses the trajectories (i) to calculate aggregates, such as the number of locations in all the cells of the location space at each timestamp, and (ii) to extract the user's behavioural movement patterns. We summarize the presented methods based on these objectives.

3.2.1 Trajectory Aggregate Sharing

When continuously publishing location statistics for monitoring purposes such as traffic analysis and social trends observation, the privacy of the user's trajectory is compromised. Continuous privacy guarantee approaches are categorized into user-level and event-level [9]. Event-level privacy protects several events in the data sequence while user-level privacy protects the entire sequence. For example, in privacy solutions for trajectory aggregates, event-level privacy hides the user's participation to a single timestamp in the trajectory's aggregate, the timestamp where the monitored event occurs. While user-level privacy hides the user's participation in the aggregates of each timestamp. The challenge in achieving user-level privacy is the degradation of the privacy guarantee du to the accumulation of privacy budget ϵ at each timestamp. The presented solutions aim at achieving a balance between user-level and the event-level privacy by managing the privacy budget distribution. We categorize the solutions into: (i) interval-based method: where user-level privacy is guaranteed in the whole trajectory by selecting timestamps to add noise and approximating aggregates for the in-between timestamps, and (ii) sliding window method: where user-level privacy is guaranteed within a window of timestamps that slides over the whole trajectory aggregates.

Interval-based Methods The privacy guarantee quickly degrades when differential privacy is applied continuously. Especially that the privacy guarantee by a factor of up to $e^{T\epsilon}$ for T consecutive location collection due to the sequential composition property of differential privacy (Theorem 1). For example, if we want to protect the users' privacy with an indistinguishability level of $\epsilon = 1$ in a trajectory aggregate composed of 1000 timestamp, then we would have to apply differential privacy at each timestamp with the privacy budget $\frac{\epsilon}{1000} = 0,001$. The problem is that the smaller the privacy budget, the higher the added noise. Therefore, this approach does not provide acceptable utility. The sample and interpolate solutions solve this problem by choosing timestamps where to use a portion of the privacy budget to calculate a private aggregate, and interpolate the aggregates of the in-between timestamps as illustrated in Fig.8. This requires answering the following questions: what strategy to use to sample the timestamps where private aggregates are calculated? how to allocate the privacy budget over the sampled timestamps? and how to interpolate the in-between aggregates?

The framework presented in [74], [75], [74] uses intervals to sample the private aggregate calculation, and prediction to release the in-between aggregates. At each timestamp, the framework predicts the next aggregate value based on the previously observed private aggregates. At sampling timestamps, the framework uses a corrector to reduce the error between the predicted and the private aggregate, the resulting value is then released. While the predicted aggregates are released for the in-between timestamps. The privacy budget is allocated evenly between all sampling timestamps.

Fixed-rate intervals samples the trajectory aggregates based on a pre-defined interval. The challenge then is to determine the optimal interval size. When the interval size is low the perturbation error introduced at each time stamp is increased. On the other hand, when the interval size is high, the released aggregates will not reflect up-to-date data values. To solve this problem, adaptive sampling based on proportional-integral-derivation controller (PID) is presented in [74] and [76]. PID controller is a form of feedback control mainly used to measure the variation of sampling performance over time [77]. The feedback is the error between the posterior and the prior estimated aggregates at each timestamp.

Sliding Window Methods Interval-based methods require knowing the overall length of the trajectory aggregates for privacy budget management, which is not always possible. The sliding window methods achieve a balance between user-privacy and event-level privacy by guaranteeing indistinguishability within a sliding window of n timestamps as illustrated in Fig.9. This requires answering the following questions: what is the optimal window size? how to allocate the privacy budget over the timestamps within the sliding window?

$w - event$ privacy is a sliding window method presented in [78], [79]. A privacy mechanism is applied at each timestamp of the sliding window to: (1) determine if a portion of the privacy budget is going to be used to calculate a private aggregate, and (2) manage the privacy budget allocation at the current timestamp to guarantee that the overall privacy budget in the sliding window does not exceed the privacy budget (ϵ). At each timestamp the similarity between the aggregate at the

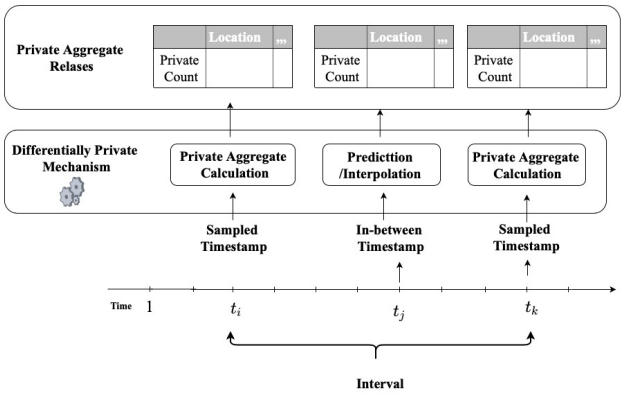


Figure 8: The interval-based method: timestamps are sampled based on an interval where a private aggregate is calculated with a portion of the privacy budget. The timestamps in-between approximate the private aggregate.

current timestamp and the last privately released aggregate is evaluated based on a threshold to decide if a new private release is required. If the difference is below a threshold, then it is more beneficial to approximate the current aggregate with the last private release. Otherwise, noise drawn from Laplace distribution is added to the aggregate using a portion of the privacy budget ϵ, then released.

Two privacy budget allocation approaches are proposed to generate the Laplace noise: the budget distribution and the budget absorption. The budget distribution approach allocates the privacy budget in an exponentially decreasing fashion, favouring the earlier timestamps with exponentially more budget than the later ones. Initially, the budget absorption approach distributes the budget uniformly to all w timestamps. Then, the accumulated privacy budget from timestamps where a noisy release is not required becomes available for future private release.

When the users' trajectories are sparse in time, the locations that need to be privately protected may stretch on a window time larger than w. Therefore, $\ell - trajectory$ defines a window size that contains ℓ different locations that the user visited instead of timestamps [80].

The sliding window approach is also used in [81], but with variable size windows. The solution considers traffic data property, such as road structure and time-based traffic variation to adaptively select the window size. A transition matrix between road sections and intersections is used to model the network connectivity. If a road has three consecutive segments, then it is easy to predict a user trajectory at the corresponding three consecutive timestamps. Therefore, the size of the sliding window is calculated based on the entropy value extracted from the transition matrix. The privacy budget is exponentially decreasing with each timestamp with a decaying factor determined by the maximum possible window size.

Discussion: The interval-based methods guarantee the user's privacy in trajectory aggregates while taking into account the user's movement regularity by factoring in the predicted aggregates to release a private aggregates **RQ2**. The sliding window methods achieve the same goal by adjusting the sliding window size based

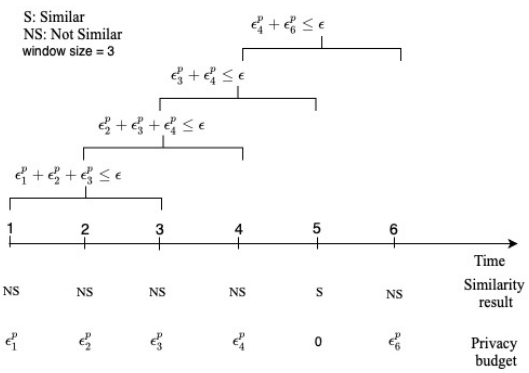


Figure 9: An example of the sliding window approach with a window size of 3 timestamps. The privacy budget is not consumed at timestamps where the aggregates are similar. The privacy budget within each sliding window is below ϵ.

on the changes in the trajectory aggregates **RQ2**. However, the privacy attack conducted in [14] shows that it is possible to recover individual user trajectories from aggregated mobility data without any prior information. This is achieved by exploiting the uniqueness and regularity of human mobility. Moreover, the problem with the trusted curator model adopted in these methods is its vulnerability to privacy attacks in which an adversary accesses the data gathered by the data curator before the data obfuscation takes place [34] . Several works ([82], [83], and [84]) show that a well designed perturbation schema can reduce the regularity and uniqueness of any individual users' trajectory **RQ3**. Therefore the presented methods are robust against attacks that exploit the user's movement regularity. Additionally, several works ([85], [86]) combined these methods with the local differential privacy setting to neutralize the vulnerability against the centralized privacy attack aforementioned.

3.2.2 Mining Frequent Travel Patterns

Finding frequent travel patterns is concerned with counting the number of times a sequence appears in a dataset of trajectories. Guaranteeing an individual's privacy in this setting means that his presence or absence does not impact the resulting travel patterns by much (proportional to ϵ). However offering such guarantee is not an easy task. Let use consider a domain with l locations, mining for a sequence of length m would require considering all l^m possible sequences. Quantifying an individual's participation impact on all these sequence can be very difficult. For instance, adding or removing a location in the dataset may generate a new sequence, therefore hindering the calculation of the global sensitivity.

The trajectories can be represented with a tree to model all possible sequences and their corresponding frequency as presented in [87] and [88]. The root node does not correspond to any location in the dataset; it marks the beginning of every sequence. The edges represent the transitions between two locations in a trajectory. Each node holds a location and the number of trajectories it appears in. Then

the most frequent patterns are extracted from the tree. To guarantee differential privacy, Laplace noise is added to the number of location appearance at each node in the tree.

One limitation of the trees model is the necessity to bound the tree height, resulting in dropping some travel patterns. To remedy this limitations, probabilistic model is used in [89] to represent contiguous subsequences of size n (n-grams). An exploration tree is used to model the patterns frequencies in the trajectories. Laplace noise is added corresponding to the path length to achieve differential privacy. Variable length n-grams are then extracted from the exploration tree representing the probability of occurrence of each subsequence of length n.

Discussion: The tree-based methods guarantee the user's privacy in mining trajectories for frequent patterns by carefully inducing noise to the uniqueness of any individual users' trajectory **RQ2**. Similarly to the methods presented for trajectory aggregate sharing, the tree-based method is robust against attacks that exploit the user's movement regularity. However, since the method relies on a trusted centralized curator, it can be vulnerable to centralized privacy attacks **RQ3**.

4 Future Directions

A summary of the solutions presented for location privacy and trajectory privacy are summarized in Tables 1 and 2 respectively. We identify the challenges in each category and introduce the areas that require further attention: the utility and privacy tradeoff, the privacy needs in application domains, and the resiliency to privacy attacks.

4.1 Utility and Privacy Tradeoff

The definition of utility and privacy depends on the objective of applying differential privacy: protecting the user's location, such as in the case of location collection, or the presence/absence of the user in the dataset, such as the case of sharing location aggregates.

Usually in raw location privacy, the distance between the private and true locations is used to measure utility. Privacy solutions that use to calculate the private location treat space in a uniform way, regardless of the privacy needs difference between regions. For example, choosing a large radius of private location calculation in a dense area may skew the significance of the released private location. Consequently imposing the addition of the same amount of noise everywhere on the location space results in a utility loss as introduced in Challenge **C1**. The solution presented in [90] defines an elastic indistinguishability metric that links the privacy needs to the region population density. It captures the different degrees of density of each area on the map, and a randomization algorithm that adapts the level of noise while achieving the same degree of privacy everywhere. However, further investigation of retaining the semantic significance of the user's location when releasing the private location is also necessary.

Laplace mechanism is de-facto mechanism for protecting the user's privacy when sharing location and trajectory aggregates. When applied to location data, Laplace is usually combined with techniques such as spatial decomposition. However, it may yield large noise in some cases, such as range queries and mining for points-of-interests. Further improvements on the utility of private analysis of location and trajectory data are necessary.

Table 1: Summary of the location privacy work

	Method		Description	Challenges
Raw location sharing	Distance-based methods [22], [23], [24], [25], [26], [27], [28], [29], [26], [27], [28], and [30]		The indistinguishability between any two users' locations is proportional to the distance between these locations.	(C1) Utility may be compromised because of the high sensitivity of the distance.
	Obfuscation-based methods [34], [35], and [36]		Random generation of an obfuscated location	(C2) Application of randomly generated locations may not be possible in every application domain.
	Anonymity-based methods [40], [41], and [42]		The indistinguishability is between locations of multiple users	(C3) The need for an anonymizer that collects the locations of multiple users may create privacy attack opportunities.
Location data analysis	Aggregation	Grid partitioning [44], [45], [46], and [47]	Divide the location space using a uniform grid, then induce noise to the counts of each cell	(C1) Utility of range queries may be compromised due to the accumulation of noise from multiple cells/regions and the difference in density between regions .
		Private Spatial Decomposition [48] and [49]	Recursively divide the location using a tree structure, then induce noise to the count at each node of the tree	
	Mining of Points-of-interests [53], [54], and [55]		Limits the number of visits per user considered when calculating the sensitivity of the popularity of a location	(C1) Limiting the number of visits may results in dropping out locations of interest and skewing the mining' utility

4.2 Privacy Needs in Application Domains

The broad spectrum of application domains renders a variety of location data with different privacy needs. Depending on the application domain, users may need to

Table 2: Summary of the trajectory privacy work

	Method		Description	Challenges
Raw trajectory sharing	Real-time	distance-based [59], [60], and [61]	Generates private locations based on the distance. Takes into account the user's movement predictability either by caching previously released locations or by predicting future locations	(C1) Utility may be compromised because of the high sensitivity of the distance
		temporal-correlation-based [64] and [65]	Takes into account the temporal correlation between consecutive locations	(C3) Co-location of users may disclose one's social connections, intimate partners, or more.
	Raw trajectory publishing [66], [67], [68], [69], and [70]		Guarantees privacy by making every location in the user's trajectory indistinguishable to the adversary	(C2) Some domain applications may requires highly accurate trajectories
Trajectory data analysis	interval-based methods [74], [75], [74], and [76]		Achieves aggregates indistinguishability while factoring in the predicted aggregates at each timestamp	(C3) The need for a centralized curator may create privacy attack opportunities
	sliding window methods [78], [79], [80], [81], and [81]		Guarantees indistinguishability within a window of timestamps	
	Frequent travel patterns mining [87] [88], and [89]		Builds a representation of the travel patterns with private frequencies	

reveal or conceal pieces of their location data. For LBS applications, [91] shows that for different types of LBS, the privacy leak can be on the location, or on the query or both. In crowdsourcing systems, the user's identity and location must be protected, however there are may be no queries to protect (Challenge **C2**). Furthermore, different application domains may require different accuracy levels. Supporting applications that require precise information, such as calculating eTolling fees, while guaranteeing differential privacy is yet to be explored.

4.3 Privacy Attacks Resiliency

The reviewed methods that protect raw locations and trajectories guarantee the user's privacy by shifting the indistinguishability to the observed locations of a single user instead of the presence or absence of the user. However, this creates opportunities for novel privacy attacks (Challenge **C3**), such as the inference attack presented in [31]. The adversary predicts the user's points-of-interests with reasonable precision (63% of the points of interests were identified) based on the reported perturbed locations. Furthermore, many of the proposed solutions rely on a trusted centralized curator. We believe that similar evaluation of privacy attacks need to be investigated in the differential privacy variations that release private locations.

5 Conclusion

This paper presents an overview of differential privacy advances in location and trajectory data. We mapped the differential privacy solutions to the data flow as follows: (i) raw location and trajectory collection and (ii) privacy-preserving frameworks that design solutions for aggregation and mining. Several aspects of location and trajectory data are taken into account when designing location and trajectory collection, including the spatial and temporal correlation between locations. The analysis frameworks combine the traditional differential privacy mechanisms (Laplace, exponential, randomized response) with partitioning and clustering techniques to tackle the high sensitivity challenge in location data. Although differential privacy is originally designed for statistical datasets, the subsequent literature proves its power and versatility in guaranteeing privacy in novel settings, such as location obfuscation. However, there are research directions that need to be explored, such as the impact of the high sensitivity on the utility of the shared data, the satisfaction of different privacy needs in application domains, and the robustness of the designed solutions w.r.t location privacy attacks.

Appendix

Acknowledgements
The authors would like to thank the anonymous reviewers for their valuable comments and suggestions on the paper.

Funding

Abbreviations

Availability of data and materials

Ethics approval and consent to participate

Competing interests
The authors declare that they have no competing interests.

Consent for publication

Authors' contributions
FZE contributed to the paper collection, manuscript organization, and drafted the first version of the manuscript. YL double checked the manuscript. All authors read and approved the final manuscript.

Authors' information
Author details
Department of Electrical and Computer Engineering, Concordia University, Montreal, Canada.

References

1. Georgiadou, Y., de By, R.A., Kounadi, O.: Location privacy in the wake of the gdpr. ISPRS international journal of geo-information **8**(3), 157 (2019)
2. Romm, T.: Arizona sues Google over allegations it illegally tracked Android smartphone users' locations. [Online; posted 27-May-2020] (2020). https://www.washingtonpost.com/technology/2020/05/27/google-android-privacy-lawsuit/
3. Gruteser, M., Grunwald, D.: Anonymous usage of location-based services through spatial and temporal cloaking. In: Proceedings of the 1st International Conference on Mobile Systems, Applications and Services, pp. 31–42 (2003). ACM
4. Bakken, D.E., Rarameswaran, R., Blough, D.M., Franz, A.A., Palmer, T.J.: Data obfuscation: Anonymity and desensitization of usable data sets. IEEE Security & Privacy **2**(6), 34–41 (2004)
5. Kushilevitz, E., Ostrovsky, R.: Replication is not needed: Single database, computationally-private information retrieval. In: Foundations of Computer Science, 1997. Proceedings., 38th Annual Symposium On, pp. 364–373 (1997). IEEE
6. Krumm, J.: Inference attacks on location tracks. In: International Conference on Pervasive Computing, pp. 127–143 (2007). Springer
7. Shokri, R., Theodorakopoulos, G., Troncoso, C., Hubaux, J.-P., Le Boudec, J.-Y.: Protecting location privacy: optimal strategy against localization attacks. In: Proceedings of the 2012 ACM Conference on Computer and Communications Security, pp. 617–627 (2012). ACM
8. Islam, M.S., Kuzu, M., Kantarcioglu, M.: Access pattern disclosure on searchable encryption: Ramification, attack and mitigation. In: Ndss, vol. 20, p. 12 (2012)
9. Dwork, C., Naor, M., Pitassi, T., Rothblum, G.N.: Differential privacy under continual observation. In: Proceedings of the Forty-second ACM Symposium on Theory of Computing, pp. 715–724 (2010). ACM
10. Song, C., Qu, Z., Blumm, N., Barabási, A.-L.: Limits of predictability in human mobility. Science **327**(5968), 1018–1021 (2010)
11. Faghihi, F., Nurmi, P.: An empirical study on the regularity of route mobility. In: Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct, pp. 1418–1425 (2016)
12. Wernke, M., Skvortsov, P., Dürr, F., Rothermel, K.: A classification of location privacy attacks and approaches. Personal and ubiquitous computing **18**(1), 163–175 (2014)
13. Zang, H., Bolot, J.: Anonymization of location data does not work: A large-scale measurement study. In: Proceedings of the 17th Annual International Conference on Mobile Computing and Networking, pp. 145–156 (2011). ACM
14. Xu, F., Tu, Z., Li, Y., Zhang, P., Fu, X., Jin, D.: Trajectory recovery from ash: User privacy is not preserved in aggregated mobility data. In: Proceedings of the 26th International Conference on World Wide Web, pp. 1241–1250 (2017)
15. Errounda, F.Z., Liu, Y.: An analysis of differential privacy research in location data. In: 2019 IEEE 5th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing,(HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS), pp. 53–60 (2019). IEEE
16. Liu, B., Zhou, W., Zhu, T., Gao, L., Xiang, Y.: Location privacy and its applications: A systematic study. IEEE access **6**, 17606–17624 (2018)
17. Dwork, C., Roth, A., *et al.*: The algorithmic foundations of differential privacy. Foundations and Trends® in Theoretical Computer Science **9**(3–4), 211–407 (2014)
18. Dwork, C.: Differential privacy: A survey of results. In: International Conference on Theory and Applications of Models of Computation, pp. 1–19 (2008). Springer
19. McSherry, F., Talwar, K.: Mechanism design via differential privacy. In: Foundations of Computer Science, 2007. FOCS'07. 48th Annual IEEE Symposium On, pp. 94–103 (2007). IEEE
20. Warner, S.L.: Randomized response: A survey technique for eliminating evasive answer bias. Journal of the American Statistical Association **60**(309), 63–69 (1965)
21. McSherry, F.D.: Privacy integrated queries: an extensible platform for privacy-preserving data analysis. In: Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data, pp. 19–30 (2009). ACM
22. ElSalamouny, E., Gambs, S.: Differential privacy models for location-based services. Transactions on Data Privacy **9**(1), 15–48 (2016)
23. Andrés, M.E., Bordenabe, N.E., Chatzikokolakis, K., Palamidessi, C.: Geo-indistinguishability: Differential privacy for location-based systems. In: Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security, pp. 901–914 (2013). ACM
24. Zhou, L., Yu, L., Du, S., Zhu, H., Chen, C.: Achieving differentially private location privacy in edge-assistant connected vehicles. IEEE Internet of Things Journal **6**(3), 4472–4481 (2018)
25. Shi, D., Ding, J., Errapotu, S.M., Yue, H., Xu, W., Zhou, X., Pan, M.: Deep q-network-based route scheduling for tnc vehicles with passengers' location differential privacy. IEEE Internet of Things Journal **6**(5), 7681–7692 (2019)
26. Wang, Z., Hu, J., Lv, R., Wei, J., Wang, Q., Yang, D., Qi, H.: Personalized privacy-preserving task allocation for mobile crowdsensing. IEEE Transactions on Mobile Computing **18**(6), 1330–1341 (2018)
27. Yan, K., Luo, G., Zheng, X., Tian, L., Sai, A.M.V.V.: A comprehensive location-privacy-awareness task selection mechanism in mobile crowd-sensing. IEEE Access **7**, 77541–77554 (2019)
28. Wang, L., Yang, D., Han, X., Zhang, D., Ma, X.: Mobile crowdsourcing task allocation with differential-and-distortion geo-obfuscation. IEEE Transactions on Dependable and Secure Computing (2019)

29. Qiu, Y., Ma, M.: A privacy-preserving proximity testing for location-based services. In: 2018 IEEE Global Communications Conference (GLOBECOM), pp. 1–6 (2018). IEEE
30. Bordenabe, N.E., Chatzikokolakis, K., Palamidessi, C.: Optimal geo-indistinguishable mechanisms for location privacy. In: Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, pp. 251–262 (2014). ACM
31. Primault, V., Mokhtar, S.B., Lauradoux, C., Brunie, L.: Differentially private location privacy in practice. Proceedings of the Third Workshop on Mobile Security Technologies **abs/1410.7744** (2014)
32. Theodorakopoulos, G.: The same-origin attack against location privacy. In: Proceedings of the 14th ACM Workshop on Privacy in the Electronic Society, pp. 49–53 (2015). ACM
33. Brush, A., Krumm, J., Scott, J.: Exploring end user preferences for location obfuscation, location-based services, and the value of location. In: Proceedings of the 12th ACM International Conference on Ubiquitous Computing, pp. 95–104 (2010). ACM
34. Kim, J.W., Kim, D.-H., Jang, B.: Application of local differential privacy to collection of indoor positioning data. IEEE Access (2018)
35. Wang, L., Zhang, D., Yang, D., Lim, B.Y., Han, X., Ma, X.: Sparse mobile crowdsensing with differential and distortion location privacy. IEEE Transactions on Information Forensics and Security **15**, 2735–2749 (2020)
36. Wang, L., Zhang, D., Yang, D., Lim, B.Y., Ma, X.: Differential location privacy for sparse mobile crowdsensing. In: Data Mining (ICDM), 2016 IEEE 16th International Conference On, pp. 1257–1262 (2016). IEEE
37. Chen, T., Boreli, R., Kaafar, M.-A., Friedman, A.: On the effectiveness of obfuscation techniques in online social networks. In: International Symposium on Privacy Enhancing Technologies Symposium, pp. 42–62 (2014). Springer
38. Hilbert, D.: Über die stetige abbildung einer linie auf ein flächenstück. In: Dritter Band: Analysis· Grundlagen der Mathematik· Physik Verschiedenes, pp. 1–2. Springer, ??? (1935)
39. Yarovoy, R., Bonchi, F., Lakshmanan, L.V., Wang, W.H.: Anonymizing moving objects: How to hide a mob in a crowd? In: Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology, pp. 72–83 (2009). ACM
40. Ngo, H., Kim, J.: Location privacy via differential private perturbation of cloaking area. In: Computer Security Foundations Symposium (CSF), 2015 IEEE 28th, pp. 63–74 (2015). IEEE
41. Yu, L., Liu, L., Pu, C.: Dynamic differential location privacy with personalized error bounds. In: The Network and Distributed System Security Symposium (2017)
42. Yang, X., Gao, L., Zheng, J., Wei, W.: Location privacy preservation mechanism for location-based service with incomplete location data. IEEE Access **8**, 95843–95854 (2020)
43. Dewri, R.: Local differential perturbations: Location privacy under approximate knowledge attackers. IEEE Transactions on Mobile Computing **12**(12), 2360–2372 (2013). doi:10.1109/TMC.2012.208
44. Li, N., Yang, W., Qardaji, W.: Differentially private grids for geospatial data. In: Proceedings of the 2013 IEEE International Conference on Data Engineering (ICDE 2013). ICDE '13, pp. 757–768. IEEE Computer Society, Washington, DC, USA (2013). doi:10.1109/ICDE.2013.6544872. http://dx.doi.org/10.1109/ICDE.2013.6544872
45. To, H., Ghinita, G., Shahabi, C.: A framework for protecting worker location privacy in spatial crowdsourcing. Proceedings of the VLDB Endowment **7**(10), 919–930 (2014)
46. To, H., Ghinita, G., Fan, L., Shahabi, C.: Differentially private location protection for worker datasets in spatial crowdsourcing. IEEE Transactions on Mobile Computing **16**(4), 934–949 (2017)
47. Wei, J., Lin, Y., Yao, X., Zhang, J.: Differential privacy-based location protection in spatial crowdsourcing. IEEE Transactions on Services Computing (2019)
48. Cormode, G., Procopiuc, C., Srivastava, D., Shen, E., Yu, T.: Differentially private spatial decompositions. In: Proceedings of the 2012 IEEE 28th International Conference on Data Engineering. ICDE '12, pp. 20–31. IEEE Computer Society, Washington, DC, USA (2012). doi:10.1109/ICDE.2012.16. http://dx.doi.org/10.1109/ICDE.2012.16
49. Yan, Y., Gao, X., Mahmood, A., Feng, T., Xie, P.: Differential private spatial decomposition and location publishing based on unbalanced quadtree partition algorithm. IEEE Access **8**, 104775–104787 (2020)
50. Zhang, J.D., Ghinita, G., Chow, C.Y.: Differentially private location recommendations in geosocial networks. In: Mobile Data Management (MDM), 2014 IEEE 15th International Conference On, vol. 1, pp. 59–68 (2014). IEEE
51. Huo, Z., Wang, T., He, P.: Differentially private moving object database publication in location tracking service. In: Proceedings of the 14th International Conference on Advances in Mobile Computing and Multi Media, pp. 276–279 (2016). ACM
52. Ho, S.-S.: Preserving privacy for moving objects data mining. In: Intelligence and Security Informatics (ISI), 2012 IEEE International Conference On, pp. 135–137 (2012). IEEE
53. Ho, S.-S., Ruan, S.: Differential privacy for location pattern mining. In: Proceedings of the 4th ACM SIGSPATIAL International Workshop on Security and Privacy in GIS and LBS. SPRINGL '11, pp. 17–24. ACM, New York, NY, USA (2011). doi:10.1145/2071880.2071884. http://doi.acm.org/10.1145/2071880.2071884
54. Ho, S.-S., Ruan, S.: Preserving privacy for interesting location pattern mining from trajectory data. Trans. Data Privacy **6**(1), 87–106 (2013)
55. Wang, S., Sinnott, R.O.: Supporting geospatial privacy-preserving data mining of social media. Social Network Analysis and Mining **6**(1), 109 (2016)
56. Ester, M., Kriegel, H.-P., Sander, J., Xu, X., *et al.*: A density-based algorithm for discovering clusters in large spatial databases with noise. In: Kdd, vol. 96, pp. 226–231 (1996)
57. Acs, G., Castelluccia, C.: A case study: privacy preserving release of spatio-temporal density in paris. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1679–1688 (2014). ACM
58. Pyrgelis, A., Troncoso, C., De Cristofaro, E.: Knock knock, who's there? membership inference on aggregate location data. arXiv preprint arXiv:1708.06145 (2017)

59. Ma, X., Ma, J., Li, H., Jiang, Q., Gao, S.: Agent: an adaptive geo-indistinguishable mechanism for continuous location-based service. Peer-to-Peer Networking and Applications, 1–13 (2017)
60. Chatzikokolakis, K., Palamidessi, C., Stronati, M.: A predictive differentially-private mechanism for mobility traces. In: International Symposium on Privacy Enhancing Technologies Symposium, pp. 21–41 (2014). Springer
61. Chatzikokolakis, K., Palamidessi, C., Stronati, M.: Location privacy via geo-indistinguishability. ACM SIGLOG News **2**(3), 46–69 (2015)
62. Mendes, R., Cunha, M., Vilela, J.P.: Impact of frequency of location reports on the privacy level of geo-indistinguishability. Proceedings on Privacy Enhancing Technologies **2020**(2), 379–396 (2020)
63. Ahuja, R., Ghinita, G., Krishna, N., Shahabi, C.: Protecting against inference attacks on co-location data. In: 2019 28th International Conference on Computer Communication and Networks (ICCCN), pp. 1–11 (2019). IEEE
64. Xiao, Y., Xiong, L.: Protecting locations with differential privacy under temporal correlations. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, pp. 1298–1309 (2015). ACM
65. Xiao, Y., Xiong, L., Zhang, S., Cao, Y.: Loclok: location cloaking with differential privacy via hidden markov model. Proceedings of the VLDB Endowment **10**(12), 1901–1904 (2017)
66. Jiang, K., Shao, D., Bressan, S., Kister, T., Tan, K.-L.: Publishing trajectories with differential privacy guarantees. In: Proceedings of the 25th International Conference on Scientific and Statistical Database Management. SSDBM, pp. 12–11212. ACM, New York, NY, USA (2013). doi:10.1145/2484838.2484846. http://doi.acm.org/10.1145/2484838.2484846
67. Shao, D., Jiang, K., Kister, T., Bressan, S., Tan, K.-L.: In: Decker, H., Lhotská, L., Link, S., Basl, J., Tjoa, A.M. (eds.) Publishing Trajectory with Differential Privacy: A Priori vs. A Posteriori Sampling Mechanisms, pp. 357–365. Springer, Berlin, Heidelberg (2013). doi:10.1007/978-3-642-40285-2. http://dx.doi.org/10.1007/978-3-642-40285-2
68. Assam, R., Hassani, M., Seidl, T.: Differential private trajectory protection of moving objects. In: Proceedings of the Third ACM SIGSPATIAL International Workshop on GeoStreaming. IWGS '12, pp. 68–77. ACM, New York, NY, USA (2012). doi:10.1145/2442968.2442977. http://doi.acm.org/10.1145/2442968.2442977
69. Assam, R., Hassani, M., Seidl, T.: Differential private trajectory obfuscation. In: International Conference on Mobile and Ubiquitous Systems: Computing, Networking, and Services, pp. 139–151 (2012). Springer
70. Assam, R., Seidl, T.: A model for context-aware location identity preservation using differential privacy. In: 2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, pp. 346–353 (2013). doi:10.1109/TrustCom.2013.45
71. Kaplan, E., Gürsoy, M.E., Nergiz, M.E., Saygin, Y.: Location disclosure risks of releasing trajectory distances. Data & Knowledge Engineering (2017)
72. Ou, L., Qin, Z., Liu, Y., Yin, H., Hu, Y., Chen, H.: Multi-user location correlation protection with differential privacy. In: Parallel and Distributed Systems (ICPADS), 2016 IEEE 22nd International Conference On, pp. 422–429 (2016). IEEE
73. Peng, Z., An, J., Gui, X., Wang, Z., Zhang, W., Gui, R., Xu, J.: Location correlated differential privacy protection based on mobile feature analysis. Ieee Access **7**, 54483–54496 (2019)
74. Fan, L., Xiong, L.: An adaptive approach to real-time aggregate monitoring with differential privacy. IEEE Transactions on Knowledge and Data Engineering **26**(9), 2094–2106 (2014)
75. Fan, L., Xiong, L., Sunderam, V.: Fast: differentially private real-time aggregate monitor with filtering and adaptive sampling. In: Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data, pp. 1065–1068 (2013). ACM
76. Yan, Y., Zhang, L., Sheng, Q.Z., Wang, B., Gao, X., Cong, Y.: Dynamic release of big location data based on adaptive sampling and differential privacy. IEEE Access **7**, 164962–164974 (2019)
77. King, M., *et al.*: Process Control: a Practical Approach. Wiley Online Library, ??? (2011)
78. Kellaris, G., Papadopoulos, S., Xiao, X., Papadias, D.: Differentially private event sequences over infinite streams. Proceedings of the VLDB Endowment **7**(12), 1155–1166 (2014)
79. Nie, Y., Huang, L., Li, Z., Wang, S., Zhao, Z., Yang, W., Lu, X.: Geospatial streams publish with differential privacy. In: International Conference on Collaborative Computing: Networking, Applications and Worksharing, pp. 152–164 (2016). Springer
80. Cao, Y., Yoshikawa, M.: Differentially private real-time data release over infinite trajectory streams. In: 2015 16th IEEE International Conference on Mobile Data Management, vol. 2, pp. 68–73 (2015). doi:10.1109/MDM.2015.15
81. Jo, G., Jung, K., Park, S.: An adaptive window size selection method for differentially private data publishing over infinite trajectory stream. Journal of Advanced Transportation **2018** (2018)
82. Chen, Z., Kan, X., Zhang, S., Chen, L., Xu, Y., Zhong, H.: Differentially Private Aggregated Mobility Data Publication Using Moving Characteristics (2019). 1908.03715
83. Gursoy, M.E., Liu, L., Truex, S., Yu, L., Wei, W.: Utility-aware synthesis of differentially private and attack-resilient location traces. In: Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, pp. 196–211 (2018)
84. Ghane, S., Kulik, L., Ramamohanarao, K.: Publishing spatial histograms under differential privacy. In: Proceedings of the 30th International Conference on Scientific and Statistical Database Management, pp. 1–12 (2018)
85. Zahra, F., Liu, Y.: Continuous location statistics sharing algorithm with local differential privacy. In: 2018 IEEE International Conference on Big Data (Big Data), pp. 5147–5152 (2018). IEEE
86. Ezabadi, S.G., Jolfaei, A., Kulik, L., Kotagiri, R.: Differentially private streaming to untrusted edge servers in intelligent transportation system. In: 2019 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), pp. 781–786 (2019). IEEE
87. Chen, R., Fung, B.C.M., Desai, B.C., Sossou, N.M.: Differentially private transit data publication: A case study

on the montreal transportation system. In: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '12, pp. 213–221. ACM, New York, NY, USA (2012). doi:10.1145/2339530.2339564. http://doi.acm.org/10.1145/2339530.2339564
88. Chen, R., Fung, B.C.M., Desai, B.C.: Differentially private trajectory data publication. CoRR **abs/1112.2020** (2011)
89. Chen, R., Acs, G., Castelluccia, C.: Differentially private sequential data publication via variable-length n-grams. In: Proceedings of the 2012 ACM Conference on Computer and Communications Security. CCS '12, pp. 638–649. ACM, New York, NY, USA (2012). doi:10.1145/2382196.2382263. http://doi.acm.org/10.1145/2382196.2382263
90. Chatzikokolakis, K., Palamidessi, C., Stronati, M.: Constructing elastic distinguishability metrics for location privacy. Proceedings on Privacy Enhancing Technologies **2015**(2), 156–170 (2015)
91. Wang, S., Hu, Q., Sun, Y., Huang, J.: Privacy preservation in location-based services. IEEE Communications Magazine **56**(3), 134–140 (2018)

Figures

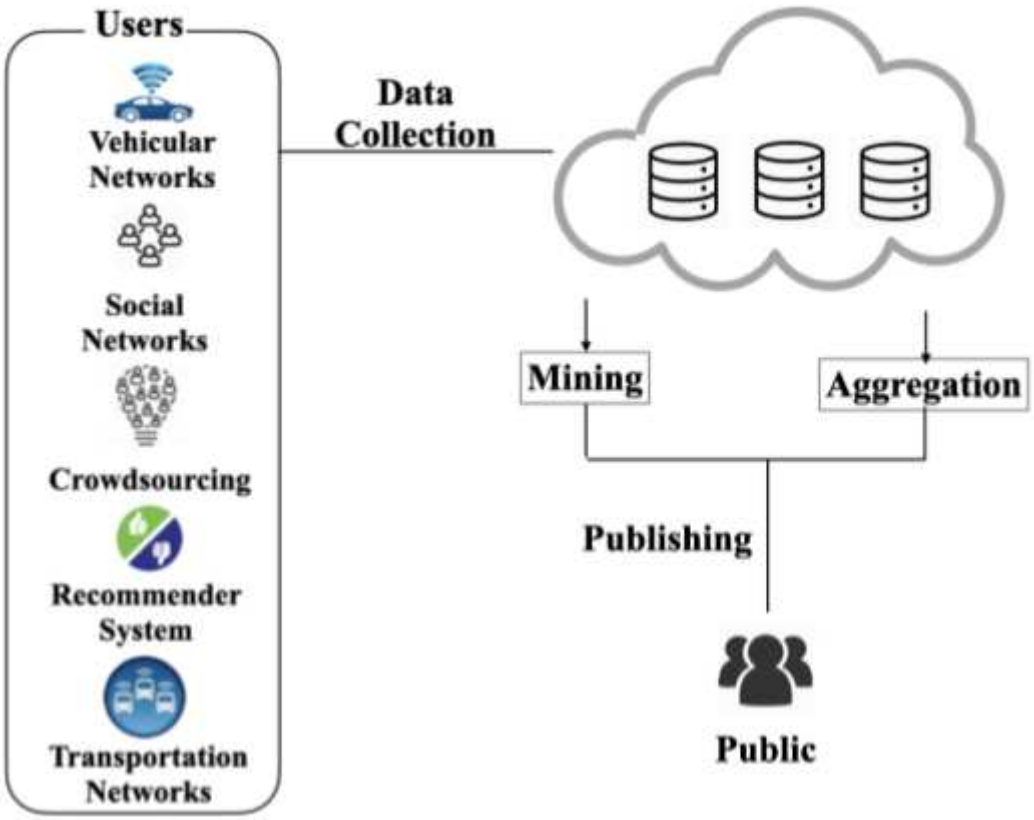


Figure 1

Location and trajectory data flow: analysis is performed on the data collected from end users, including aggregation and mining. Privacy risks are present in every step.

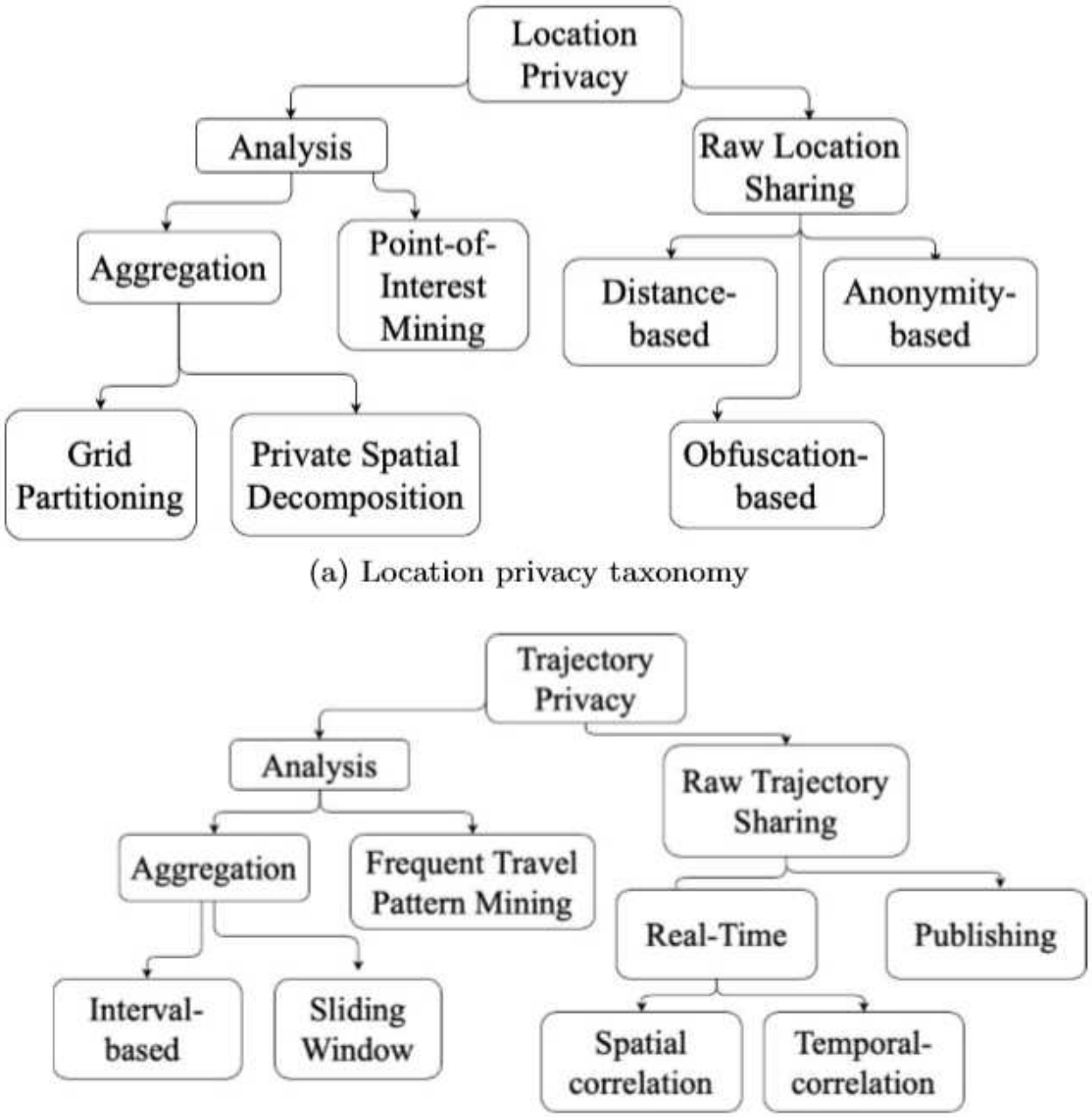


Figure 2

The categorization taxonomy based on the location and trajectory data flow

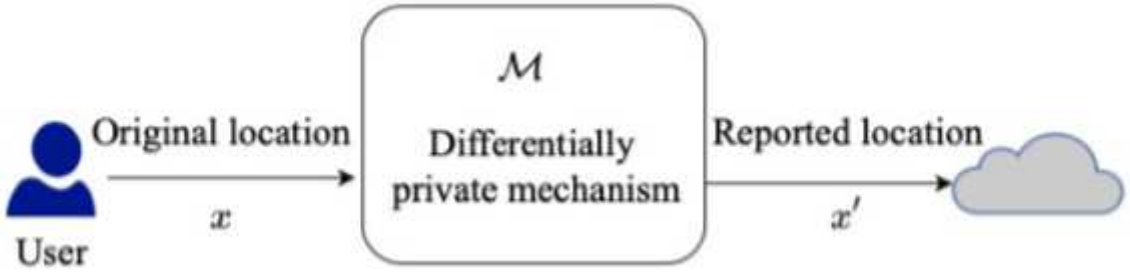


Figure 3

Differentially private mechanisms for raw location sharing report a private location x0 after processing the true location x

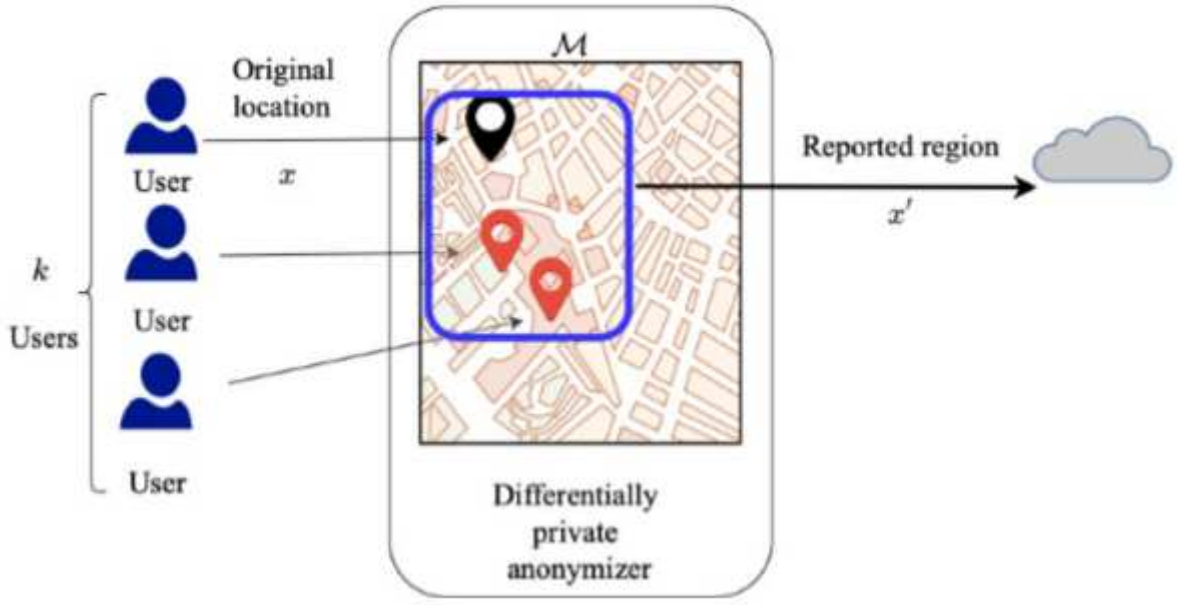
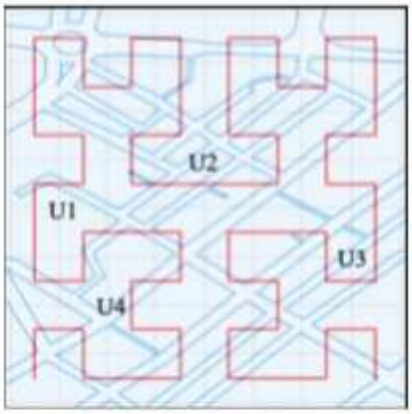


Figure 4

Differentially private anonymizer for sharing locations



index	1	2	3	4
user	U4	U1	U2	U3

Figure 5

An example of a transformation from two dimensional data (a map with users U1, U2, U3, and U4) to a single dimensional data (indexed 1 to 4 for clarification purposes)

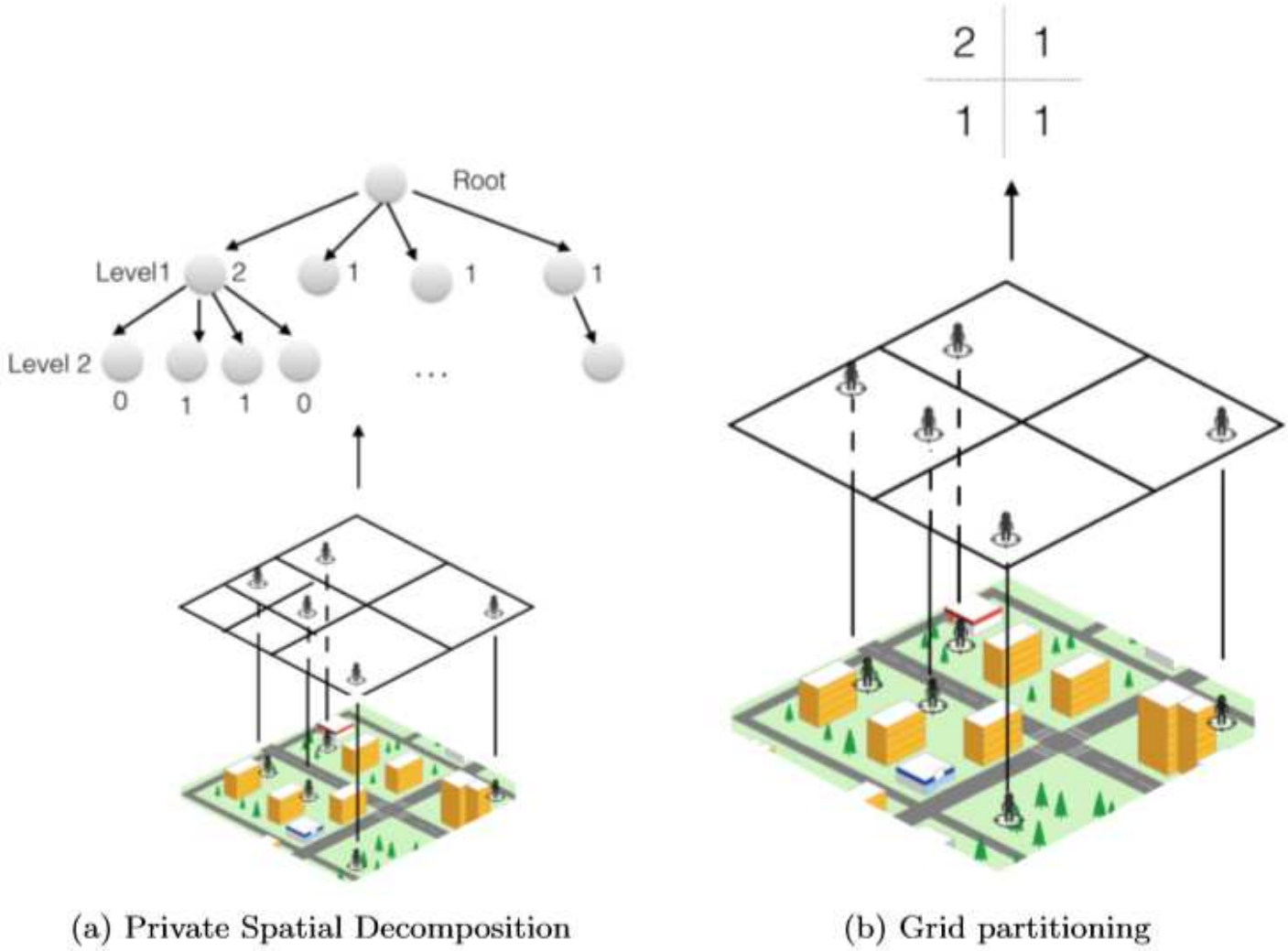


Figure 6

An example of the two spatial division approaches: (a) the space is divided using a quad-tree, the number of users present in each cell is recorded in the corresponding node (b) the space as divided using a grid of 2 by 2 and the corresponding number of users

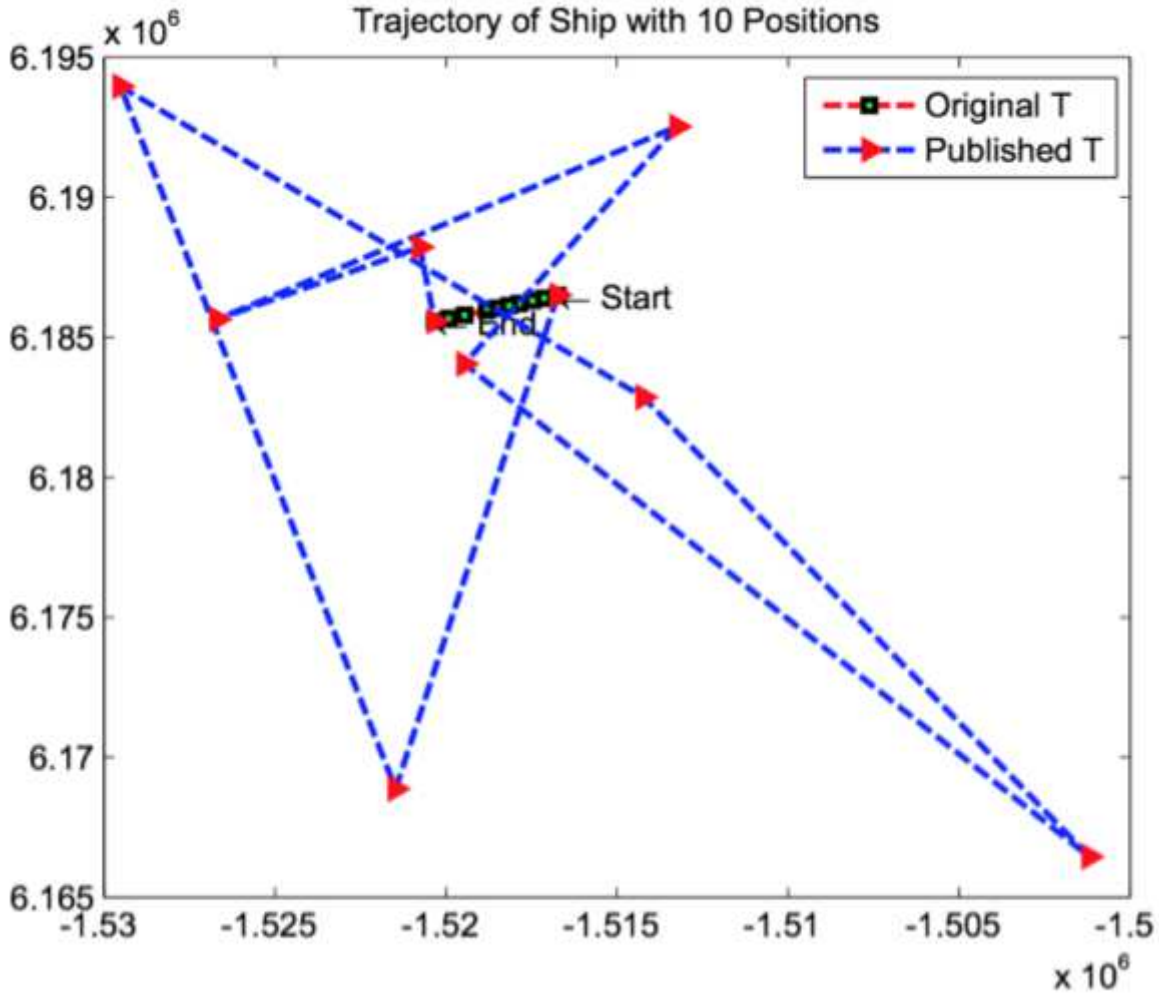


Figure 7

An example of zigzagged trajectory generated with Laplace noise [66]

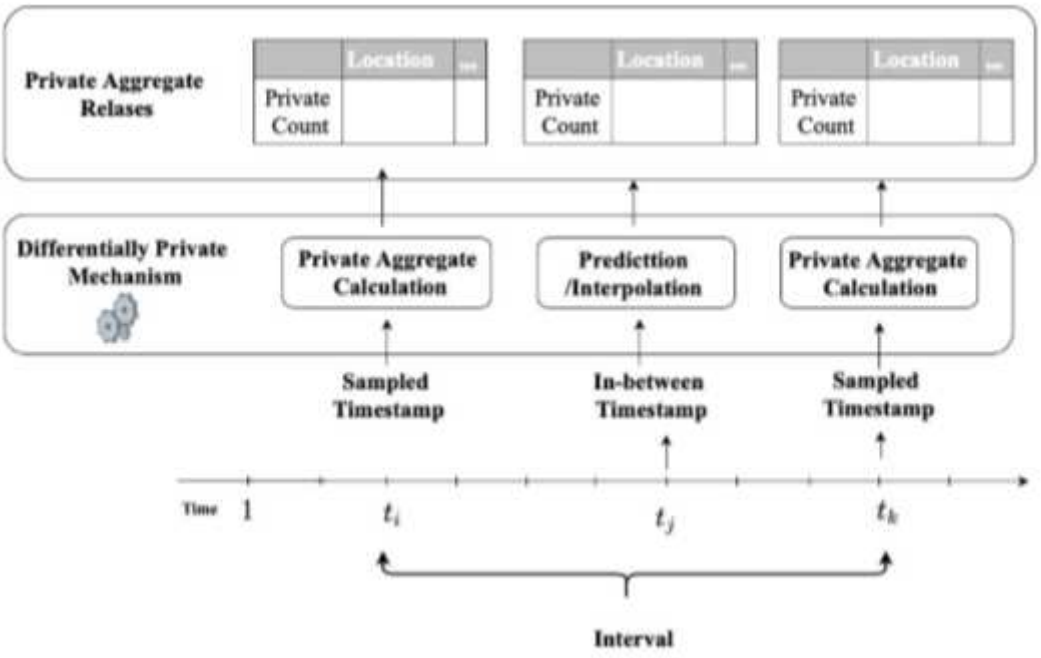


Figure 8

The interval-based method: timestamps are sampled based on an interval where a private aggregate is calculated with a portion of the privacy budget. The timestamps in-between approximate the private aggregate.

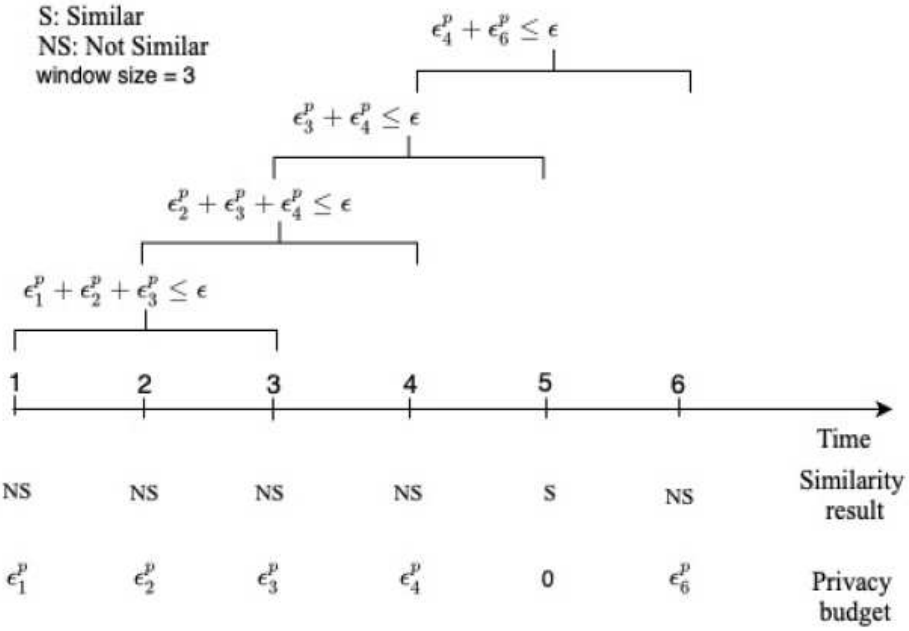


Figure 9

An example of the sliding window approach with a window size of 3 timestamps. The privacy budget is not consumed at timestamps where the aggregates are similar. The privacy budget within each sliding window is below.