$k^{\tau,\epsilon}$-anonymity: Towards Privacy-Preserving Publishing of Spatiotemporal Trajectory Data

Marco Gramaglia*, Marco Fiore†, Alberto Tarable†, Albert Banchs*

* IMDEA Networks Institute & Universidad Carlos III de Madrid
Avda. del Mar Mediterraneo, 22
28918 Leganes (Madrid), Spain
Email: name.surname@imdea.org

† CNR-IEIIT
Corso Duca degli Abruzzi, 24
10129 Torino, Italy
Email: name.surname@ieiit.cnr.it

***Abstract*—Mobile network operators can track subscribers via passive or active monitoring of device locations. The recorded trajectories offer an unprecedented outlook on the activities of large user populations, which enables developing new networking solutions and services, and scaling up studies across research disciplines. Yet, the disclosure of individual trajectories raises significant privacy concerns: thus, these data are often protected by restrictive non-disclosure agreements that limit their availability and impede potential usages. In this paper, we contribute to the development of technical solutions to the problem of privacy-preserving publishing of spatiotemporal trajectories of mobile subscribers. We propose an algorithm that generalizes the data so that they satisfy $k^{\tau,\epsilon}$-anonymity, an original privacy criterion that thwarts attacks on trajectories. Evaluations with real-world datasets demonstrate that our algorithm attains its objective while retaining a substantial level of accuracy in the data. Our work is a step forward in the direction of open, privacy-preserving datasets of spatiotemporal trajectories.**

I. Introduction

Subscriber trajectory datasets collected by network operators are logs of timestamped, georeferenced events associated to the communication activities of individuals. The analysis of these datasets allows inferring *fine-grained* information about the movements, habits and undertakings of vast user populations. This has many different applications, encompassing both business and research. For instance, trajectory data can be used to devise novel data-driven network optimization techniques [1] or support content delivery operations at the network edge [2]. They can also be monetized via added-value services such as transport analytics [3] or location-based marketing [4]. Additionally, the relevance of massive movement data from mobile subscribers is critical in research disciplines such as physics, sociology or epidemiology [5].

The importance of trajectory data has also been recognized in the design of future 5G networks, with a thrust towards the introduction of data interfaces among network operators and over-the-top (OTT) providers to give them online access to this (and other) data. OTTs can leverage such interfaces to automatically retrieve the data and process them on the fly, thus enabling new applications such as intelligent transportation [6] or assisted-life services [7].

All these use cases stem from the disclosure of trajectory datasets to third parties. However, the open release of such data is still largely withhold, which hinders potential usages and applications. A major barrier in this sense are privacy concerns: data circulation exposes it to re-identification attacks, and cognition of the movement patterns of de-anonymized individuals may reveal sensitive information about them.

This calls for anonymization techniques. The common practice operators adhere to is replacing personal identifiers (e.g., name, phone number, IMSI) with pseudo-identifiers (i.e., random or non-reversible hash values). Whether this is a sufficient measure is often called into question, especially in relation to the possibility of tracking user movements. What is sure is that pseudo-identifiers have been repeatedly proven not to protect against user trajectory uniqueness, i.e., the fact that mobile subscribers have distinctive travel patterns that make them univocally recognizable even in very large populations [8]–[10]. Uniqueness is not a privacy threat per-se, but it is a vulnerability that can lead to re-identification. Examples are brought forth by recent attempts at cross-correlating mobile operator-collected trajectories with georeferenced check-ins of Flickr and Twitter users [11], with credit card records [12] or with Yelp, Google Places and Facebook metadata [13].

More dependable anonymization solutions are needed. However, the strategies devised to date for relational databases, location-based services, or regularly sampled (e.g., GPS) mobility do not suit the irregular sampling, time sparsity, and long duration of trajectories collected by mobile operators. Moreover, current privacy criteria, including k-anonymity and differential privacy, do not provide sufficient protection or are impractical in this context. See Sec. V for a detailed discussion.

In this paper, we put forward several contributions towards *privacy-preserving data publishing (PPDP)* of mobile subscriber trajectories. Our contributions are as follows: *(i)* we outline attacks that are especially relevant to datasets of spatiotemporal trajectories; *(ii)* we introduce $k^{\tau,\epsilon}$-anonymity, a novel privacy criterion that effectively copes with the most threatening attacks above; *(iii)* we develop **`k-merge`**, an algorithm that solves a fundamental problem in the anonymization of spatiotemporal trajectories, i.e., effective generalization; *(iv)* we implement **`kte-hide`**, a practical solution based on **`k-merge`** that attains $k^{\tau,\epsilon}$-anonymity in spatiotemporal trajectory data; *(v)* we evaluate our approach on real-world datasets, showing that it achieves its objectives while retaining a substantial level of accuracy in the anonymized data.

II. Requirements and models

We first present the requirements of PPDP, in Sec. II-A, and formalize the specific attacker model we consider, in Sec. II-B. We then propose a consistent privacy model, in Sec. II-C.

A. PPDP requirements

PPDP is defined as the development of methods for the publication of information that allows meaningful knowledge discovery, and yet preserves the privacy of monitored subjects [14]. The requisites of PPDP are similar for all types of databases, including our specific case, i.e., datasets of spatiotemporal trajectories. They are as follows.

1. *The non-expert data publisher.* Mining of the data is performed by the data recipient, and not by the data publisher. The only task of the data publisher is to anonymize the data for publication.
2. *Publication of data, and not of data mining results*. The aim of PPDP is producing privacy-preserving datasets, and not anonymized datasets of classifiers, association rules, or aggregate statistics. This sets PPDP apart from privacy-preserving data mining (PPDM), where the final usage of the data is known at dataset compilation time.
3. *Truthfulness at the record level*. Each record of the published database must correspond to a real-world subject. Moreover, all information on a subject must map to actual activities or features of the subject. This avoids that fictitious data introduces unpredictable biases in the anonymized datasets.

Our privacy model will obey the principles above. We stress that they impose that the privacy model must be agnostic of data usage (points 1 and 2), and that it cannot rely on randomized, perturbed, permuted and synthetic data (point 3).

B. Attacker model

Unlike PPDP requirements, the attacker model is necessarily specific to the type of data we consider, and it is characterized by the *knowledge* and *goal* of the adversary. The former describes the information the opponent possesses, while the latter represents his privacy-threatening objective.

1) Attacker knowledge: In trajectory datasets, each data record is a sequence of spatiotemporal samples. We assume an attacker who can track a target subscriber continuously during any amount of time τ. The adversary knowledge consists then in all spatiotemporal samples in the victim's trajectory over a continuous[1] time interval of duration τ.

2) Attacker goal: Attacks against user privacy in published data can have different objectives, and a comprehensive classification is provided in [14]. Two classes of attacks are especially relevant in the context of mobile subscriber trajectory data. Both exploit the uniqueness of movement patterns that, as mentioned in Sec. I, characterizes trajectory data.

- *Record linkage attacks.* These attacks aim at univocally distinguishing an individual in the database. A successful record linkage enables cross-database correlation, which may ultimately unveil the identity of the user. Record linkage attacks on mobile traffic data have been repeatedly and successfully demonstrated [8]–[10]. As mentioned in Sec. I, they have also been used for subsequent cross-database correlations [11]–[13].
- *Probabilistic attacks.* These attacks let an adversary with partial information about an individual enlarge his knowledge on that individual by accessing the database. They are especially relevant to spatiotemporal trajectories, as shown by seminal works that first unveiled the anonymization issues of mobile traffic datasets [8], [9]. Let us imagine a scenario where an adversary knows a small set of spatiotemporal points in the trajectory of a subscriber (because, e.g., he met the target individual there). A successful probabilistic attack would reveal the complete movements of the subscriber to the attacker, who could then use them to infer sensitive information about the victim, such as home/work locations, daily routines, or visits to healthcare structures.

Our privacy model will address both classes of attacks above, led by an adversary with knowledge described in Sec. II-B1.

C. Privacy model

Our privacy model is designed following the PPDP requirements and attacker model presented before. We start by considering suitable privacy criteria against record linkage and probabilistic attacks, in Sec. II-C1 and Sec. II-C2, respectively. We then show how the first criterion is in fact a specialization of the second, in Sec. II-C3, which allows us to focus on a single unifying privacy model. Finally, we present the elementary techniques that we employ to implement the target privacy criterion, in Sec. II-C4.

1) k-anonymity: The *k-anonymity* criterion realizes the *indistinguishability principle*, by commending that each record in a database must be indistinguishable from at least $k-1$ other records in the same database [15]. In our case, this maps to ensuring that each subscriber is hidden in a crowd of k users whose trajectories cannot be told apart. The popularity of k-anonymity for PPDP has led to indiscriminated use beyond its scope, and subsequent controversy on the privacy guarantees it can provide. E.g., k-anonymity has been proven ineffective againt attacks aiming at attribute linkage (including exploits of insufficient side-information diversity), at localizing users, or at disclosing their presence and meetings [16]–[18].

However, k-anonymity remains a legitimate criterion against record linkage attacks on any kind of database [14]. Therefore, this privacy model protects trajectory data from the first type of attack in Sec. II-B, including its variations in [8]–[13].

2) $k^{\tau,\epsilon}$-anonymity: No privacy criterion proposed to date can safeguard spatiotemporal trajectory data from the second type of attacks in Sec. II-B, i.e., probabilistic attacks. This forces us to define an original criterion, as follows.

[1]Non-continuous tracking in the attacker model is an interesting but very challenging open problem. A mitigative solution realisable with our model is considering a τ that covers all disjoint tracking intervals.

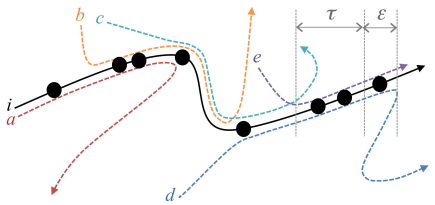


Fig. 1. Illustrative example of $k^{\tau,\epsilon}$-anonymity of user i, with k=2.

The pertinent principle here is the so-called *uninformative principle*, i.e., ensuring that the difference between the knowledge of the adversary before and after accessing a database is small [16]. In our context, this principle warrants that an attacker who knows some subset of a subscriber's movements cannot extract from the dataset a substantially longer portion of that user's trajectory.

To attain the uninformative principle, we introduce the $k^{\tau,\epsilon}$-*anonymity* privacy criterion. $k^{\tau,\epsilon}$-anonymity can be seen as a variation of k^m-*anonymity*, which establishes that each individual in a dataset must be indistinguishable from at least $k-1$ other users in the same dataset, when limiting the attacker knowledge to any set of m attributes [19]. $k^{\tau,\epsilon}$-anonymity tailors k^m-anonymity to our scenario, as follows.

- As per Sec. II-B, the attacker knowledge can be any continued sequence of spatiotemporal samples covering a time interval of length at most τ: thus, the m parameter of k^m-anonymity maps to the (variable) set of samples contained in any time period τ. During any such time period, every trajectory in the dataset must be indistinguishable from at least other $k-1$ trajectories.
- The maximum additional knowledge that the attacker is allowed to learn is called *leakage*; it consists of the spatiotemporal samples of the target user's trajectory contained in a time interval of duration at most ϵ, disjoint from the original τ. In order to fulfill the uninformative principle, the leakage ϵ must be small.

The two requirements above imply alternating in time the $k-1$ trajectories that provide anonymization. An intuitive example is provided in Fig. 1. There, the trajectory of a target user i is $2^{\tau,\epsilon}$-anonymized using those of five other subscribers. The overlapping between the trajectories of a, b, c, d, e and that of i is partial and varied. An adversary knowing a sub-trajectory of i during any time interval of duration τ always finds at least one other user with a movement pattern that is identical to that of i during that interval, but different elsewhere. With this knowledge, the adversary cannot tell apart i from the other subscriber, and thus cannot attribute full trajectories to one user or the other. As this holds no matter where the knowledge interval is shifted to, the attacker can never retrieve the complete movement patterns of i: this achieves the uninformative principle. Still, the adversary can increase its knowledge in some cases. Let us consider the interval τ indicated in the figure: the trajectories of i, d and e are identical for some time after τ, which allows associating to i the movements during ϵ: the opponent learns one additional spatiotemporal sample of i.

3) Relationship between the privacy criteria: It is easy to see that k-anonymity is a special case of $k^{\tau,\epsilon}$-anonymity. As a matter of fact, the latter criterion reduces to the former when $\tau+\epsilon$ covers the whole temporal duration of the trajectory dataset. Then, $k^{\tau,\epsilon}$-anonymity commends that each complete trajectory is indistinguishable from $k-1$ other trajectories, which is the definition of k-anonymity. Our point here is that an anonymization solution that implements $k^{\tau,\epsilon}$-anonymity can be straightforwardly employed to attain k-anonymity as well, by properly adjusting the τ and ϵ parameters.

In the light of these considerations, we address the problem of achieving $k^{\tau,\epsilon}$-anonymity in datasets of spatiotemporal trajectories of mobile subscribers. By doing so, we develop a complete anonymization solution that is effective against probabilistic attacks, but can also be specialized to guarantee k-anonymity and counter record linkage attacks.

4) Generalization and suppression: In order to enforce $k^{\tau,\epsilon}$-anonymity for all users in the dataset, we need to tweak the spatiotemporal samples in the trajectories of individuals, so that the criterion in Sec. II-C2 is respected for all of them. To that end, we rely on two elementary techniques, i.e., *spatiotemporal generalization* and *suppression* of samples.

Spatiotemporal generalization reduces the precision of trajectory samples in space and time, so as to make the samples of two or more users indistinguishable. Suppression removes from the trajectories those samples that are too hard to anonymize. Both techniques are lossy, i.e., imply some reduction of precision in the data. Yet, unlike other approaches, these techniques conform to the PPDP requirement of truthfulness at the record level, see Sec. II-A.

III. Achieving $k^{\tau,\epsilon}$-anonymity

Our goal is ensuring that an anonymized dataset of mobile subscriber trajectories respects the uninformative principle, by implementing, through generalization and suppression, the $k^{\tau,\epsilon}$-anonymity of all subscriber trajectories in the dataset. Clearly, we aim at doing so while minimizing the loss of spatiotemporal granularity in the data.

We start by defining the basic operation of generalizing a set of spatiotemporal samples, and the associated cost in terms of loss of granularity, in Sec. III-A. We then extend both notions to (sub-)trajectories, in Sec. III-B. Building on these definitions, we discuss in Sec. III-C the optimal spatiotemporal generalization of k (sub-)trajectories. We implement the result into **`k-merge`**, an optimal low-complexity algorithm that generalizes (sub-)trajectories with minimal loss of data granularity, in Sec. III-D. Once able to merge (sub-)trajectories optimally, we propose an approach to guarantee $k^{\tau,\epsilon}$-anonymity of the trajectory of a single user, in Sec. III-E, and we then scale the solution to multiple users in Sec. III-F. Finally, we introduce **`kte-hide`**, an algorithm that ensures $k^{\tau,\epsilon}$-anonymity in spatiotemporal trajectory datasets, in Sec. III-G.

A. Generalization of samples

A (raw) *sample* of a spatiotemporal trajectory represents the position of a subscriber at a given time, and we model it with

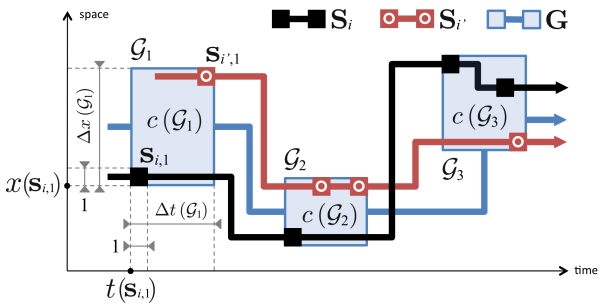


Fig. 2. Example of merging of trajectories $\mathbf{S}_i = \{\mathbf{s}_{i,j}\}$ and $\mathbf{S}_{i'} = \{\mathbf{s}_{i',j}\}$ into a generalized trajectory $\mathbf{G} = \{\mathcal{G}\}$. For clarity, space is unidimensional.

a length-3 real vector $\mathbf{s} = (t(\mathbf{s}), x(\mathbf{s}), y(\mathbf{s}))$. Since a dataset is characterized by a finite granularity in time and space, a sample is in fact a slot spanning some minimum temporal and spatial intervals. The vector entries above can be regarded as the origins of a normalized length-1 time interval and a normalized 1×1 two-dimensional area[2].

Spatiotemporal generalization merges together two or more raw samples into a *generalized sample*, i.e., a slot with a larger span. Mathematically, a generalized sample $\mathcal{G}$ can be represented as the set of the merged samples. There is a cost associated with merging samples, which is related to the span of the corresponding generalized sample, i.e., to the loss of granularity induced by the generalization. The cost of the operation of merging a set of samples into the generalized sample $\mathcal{G}$ is defined as

$$c(\mathcal{G}) = c_t(\mathcal{G})\, c_s(\mathcal{G}), \tag{1}$$

where $c_t(\mathcal{G})$ represents the cost in the time dimension, while $c_s(\mathcal{G})$ is the cost in the space dimensions.

Let $\mathcal{G}_1$ and $\mathcal{G}_2$ be two disjoint generalized samples (i.e., $\mathcal{G}_1 \cap \mathcal{G}_2 = \emptyset$). Then, we make the following two assumptions on the time and space merging costs:

$$c_t(\mathcal{G}_1 \cup \mathcal{G}_2) \geq c_t(\mathcal{G}_1) + c_t(\mathcal{G}_2) \tag{2}$$

$$c_s(\mathcal{G}_1 \cup \mathcal{G}_2) \geq \max\{c_s(\mathcal{G}_1), c_s(\mathcal{G}_2)\}. \tag{3}$$

Hereafter, we use the following definitions to implement the generic costs $c_t(\mathcal{G})$ and $c_s(\mathcal{G})$:

$$c_t(\mathcal{G}) = \Delta t(\mathcal{G}) \tag{4}$$

$$c_s(\mathcal{G}) = \Delta x(\mathcal{G}) + \Delta y(\mathcal{G}), \tag{5}$$

where

$$\Delta\star(\mathcal{G}) = \max_{\mathbf{s}\in\mathcal{G}} \star(\mathbf{s}) - \min_{\mathbf{s}\in\mathcal{G}} \star(\mathbf{s}) + 1, \tag{6}$$

with $\star \in \{t, x, y\}$, is the span in each dimension.

Therefore, in our implementation, $c(\mathcal{G})$ is the area of a rectangle with sides $\Delta t(\mathcal{G})$ and $\Delta x(\mathcal{G}) + \Delta y(\mathcal{G})$. A graphical example is provided in Fig. 2, where two raw samples $\mathbf{s}_{i,1}$ and $\mathbf{s}_{i',1}$ are merged into a generalized sample $\mathcal{G}_1$, spanning $\Delta t(\mathcal{G}_1)$ in time and $\Delta x(\mathcal{G}_1)$ in space (portrayed as unidimensional in the figure, for the sake of readability).

Remark 1: The rationale for our choice of costs is computational efficiency. Also, summing the two space spans before multiplication allows balancing the time and space contributions. Finally, note that with the definition in (5), the space merging cost assumption in (3) is trivially true. Instead, the definition in (4) lets the time merging cost assumption in (2) hold only if the time intervals spanned by $\mathcal{G}_1$ and $\mathcal{G}_2$ are non-overlapping. The time coherence property that we will introduce in Sec. III-B ensures that this is the always case.

B. Generalization of trajectories

A spatiotemporal (sub-)trajectory describes the movements of a single subscriber during the dataset timespan. Formally, a *trajectory* is an ordered vector of samples $\mathbf{S} = (\mathbf{s}_1, \ldots, \mathbf{s}_N)$, where the ordering is induced by the time coordinate, i.e., $t(\mathbf{s}_i) < t(\mathbf{s}_{i'})$ if and only if $i < i'$.

A *generalized trajectory*, obtained by merging different trajectories, is defined as an ordered vector of generalized samples $\mathbf{G} = (\mathcal{G}_1, \ldots, \mathcal{G}_Z)$. Here the ordering is more subtle, and based on the fact that the time intervals spanned by the generalized samples are non-overlapping, a property that will be called *time coherence*. More precisely, if $\mathcal{G}_i$ and $\mathcal{G}_{i'}$, $i < i'$, are two generalized samples of $\mathbf{G}$, then

$$\max_{\mathbf{s}\in\mathcal{G}_i} t(\mathbf{s}) < \min_{\mathbf{s}\in\mathcal{G}_{i'}} t(\mathbf{s}).$$

An example of a generalized trajectory $\mathbf{G}$ merging two trajectories $\mathbf{S}_i$ and $\mathbf{S}_{i'}$ is provided in Fig. 2. $\mathbf{G}$ fulfils time coherence, as its generalized samples are temporally disjoint.

Remark 2: Time coherence is a defining property of generalized trajectories in PPDP. As a matter of fact, publishing trajectory data with time-overlapping samples would generate semantic ambiguity and make analyses cumbersome.

Analogously to the cost of merging samples, we can define a cost of merging multiple trajectories into a generalized trajectory. We define such cost as the sum of costs of all generalized samples belonging to it. More precisely, if $\mathbf{G} = (\mathcal{G}_1, \ldots, \mathcal{G}_Z)$, and $c(\cdot)$ is defined as in (1), then the cost of $\mathbf{G}$ is given by:

$$C(\mathbf{G}) = \sum_{i=1}^{Z} c(\mathcal{G}_i). \tag{7}$$

Remark 3: The cost in (7) is the overall surface covered by samples of the generalized trajectory over the spatiotemporal plane. E.g., in Fig. 2, the cost of $\mathbf{G}$ is the sum of the three areas, i.e., $c(\mathcal{G}_1) + c(\mathcal{G}_2) + c(\mathcal{G}_3)$. It is thus proportional to the total loss of granularity induced by the generalization.

C. Optimal generalization of trajectories

We now formalize the problem of *optimal* generalization of spatiotemporal (sub-)trajectories. Suppose that we have k trajectories $\mathbf{S}_1, \ldots, \mathbf{S}_k$, with $\mathbf{S}_i = (\mathbf{s}_{i,1}, \ldots, \mathbf{s}_{i,N_i})$, $i = 1, \ldots, k$. The goal is a generalized trajectory $\mathbf{G}^* = (\mathcal{G}_1^*, \ldots, \mathcal{G}_Z^*)$ from $\mathbf{S}_1, \ldots, \mathbf{S}_k$, which satisfies the following conditions.

[2]For instance, in our reference datasets, the sample granularity is 1 minute in time and 100 meters in space. A raw sample spans then one slot (i.e., 1 minute) in time and one slot (i.e., a 100×100 m^2 area) in space. However, our discussion is general, and holds for any precision in the data.

i) The union of all generalized samples of $\mathbf{G}^*$ must coincide with the union of all samples of $\mathbf{S}_1, \ldots, \mathbf{S}_k$, i.e.,

$$\mathcal{G}_1^* \cup \cdots \cup \mathcal{G}_Z^* = \mathcal{S}_1 \cup \cdots \cup \mathcal{S}_k \triangleq \mathcal{S},$$

where $\mathcal{S}_i = \bigcup_{j=1}^{N_i} \{\mathbf{s}_{i,j}\}$. Thus, $\mathbf{G}^*$ is a partition of the set $\mathcal{S}$ of all samples in the input trajectories: it does not add any alien sample or discard any input sample.

ii) Each generalized sample contains at least one sample from each of the k input trajectories $\mathbf{S}_1, \ldots, \mathbf{S}_k$, i.e.,

$$\mathcal{G}_i^* \cap \mathcal{S}_{i'} \neq \emptyset, \;\; i = 1, \ldots, Z, \;\; i' = 1, \ldots, k.$$

This imposes that each input trajectory contributes to each generalized sample of $\mathbf{G}^*$. Otherwise, the merging could associate generalized samples to users that never visited the generalized location at the generalized time, violating point 3 of the PPDP requirements in Sec. II-A.

iii) The cost of the merging is minimized, i.e.,

$$\mathbf{G}^* = \arg\min_{\mathbf{G} \in \mathcal{K}} C(\mathbf{G}), \tag{8}$$

where $\mathcal{K}$ is the set of all partitions of $\mathcal{S}$ satisfying time coherence as well as condition *ii)* above, and $C(\mathbf{G})$ is in (7). In Fig. 2, the generalized trajectory $\mathbf{G}$ fulfils all these requirements, and is thus the optimal merge $\mathbf{G}^*$ of $\mathbf{S}_i$ and $\mathbf{S}_{i'}$.

Solving the problem above with a brute-force search is computationally prohibitive, since $\mathcal{K}$ has a size that grows exponentially with $|\mathcal{S}|/k$, where $|\cdot|$ denotes cardinality. However, we can characterize $\mathbf{G}^*$ so that it is possible to compute it with low complexity. To that end, we name *elementary* a partition $\mathbf{G} \in \mathcal{K}$ that cannot be refined to another partition within $\mathcal{K}$. In other words, none of the generalized samples of an elementary partition can be split into two generalized samples without violating conditions i) and ii) above, or time coherence. Then, we have the following proposition.

Proposition 1: Given the input trajectories $\mathbf{S}_1, \ldots, \mathbf{S}_k$, the optimal $\mathbf{G}^*$ defined in (8) is an elementary partition.

Proof: Suppose $\mathbf{G} \in \mathcal{K}$ is not elementary, so that it can be refined to another partition $\widetilde{\mathbf{G}} \in \mathcal{K}$. In particular, without loss of generality, suppose that $\mathbf{G} = (\mathcal{G}_1, \ldots, \mathcal{G}_Z)$ and $\widetilde{\mathbf{G}} = \left(\widetilde{\mathcal{G}}_1, \ldots, \widetilde{\mathcal{G}}_{Z+1}\right)$, where

$$\mathcal{G}_i = \begin{cases} \widetilde{\mathcal{G}}_i, & i < Z \\ \widetilde{\mathcal{G}}_Z \cup \widetilde{\mathcal{G}}_{Z+1}, & i = Z. \end{cases} \tag{9}$$

From (7) and (9), the difference between the costs of $\mathbf{G}$ and $\widetilde{\mathbf{G}}$ is given by

$$C(\mathbf{G}) - C(\widetilde{\mathbf{G}}) = c(\mathcal{G}_Z) - c(\widetilde{\mathcal{G}}_Z) - c(\widetilde{\mathcal{G}}_{Z+1}). \tag{10}$$

Since $\mathcal{G}_Z$ contains the union of raw samples in $\widetilde{\mathcal{G}}_Z$ and $\widetilde{\mathcal{G}}_{Z+1}$, we can apply properties (2) and (3) (where (2) holds because of time coherence) and obtain:

$$\begin{aligned} c(\mathcal{G}_Z) &= c_t(\mathcal{G}_Z) c_s(\mathcal{G}_Z) \\ &\geq \left(c_t(\widetilde{\mathcal{G}}_Z) + c_t(\widetilde{\mathcal{G}}_{Z+1})\right) c_s(\mathcal{G}_Z) \\ &\geq c_t(\widetilde{\mathcal{G}}_Z) c_s(\widetilde{\mathcal{G}}_Z) + c_t(\widetilde{\mathcal{G}}_{Z+1}) c_s(\widetilde{\mathcal{G}}_{Z+1}) \\ &= c(\widetilde{\mathcal{G}}_Z) + c(\widetilde{\mathcal{G}}_{Z+1}). \end{aligned} \tag{11}$$

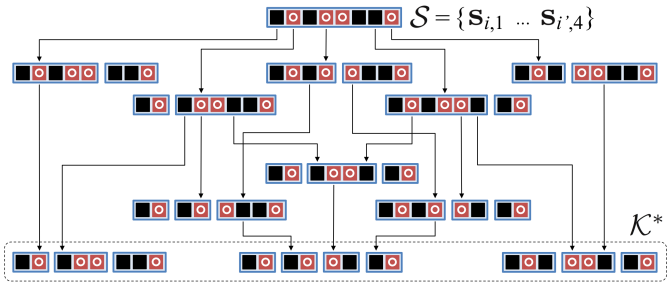


Fig. 3. Partition tree for the two trajectories $\mathbf{S}_i = \{\mathbf{s}_{i,j}\}$ and $\mathbf{S}_{i'} = \{\mathbf{s}_{i',j}\}$ in Fig. 2. Nodes in the complete tree represent the set $\mathcal{K}$ of valid partitions of the set of raw samples $\mathcal{S}$. Elementary partitions are the tree leaves and constitute $\mathcal{K}^*$. The partition in Fig. 2 is the leftmost leaf in the tree.

Algorithm 1: `k-merge` algorithm pseudocode.

input : Trajectories $\mathbf{S}_1, \ldots, \mathbf{S}_k$, where $\mathbf{S}_i = (\mathbf{s}_{i,1}, \ldots, \mathbf{s}_{i,N_i})$
output: Generalized sample set $\mathbf{G}^*$, Cost $C(\mathbf{G}^*)$

1 **foreach** $i \in [1, k]$ **do**
2 $\quad \mathcal{S}_i = \bigcup_{j=1}^{N_i} \{\mathbf{s}_{i,j}\}$;
3 $\mathcal{S} \leftarrow$ `timesort` $(\mathcal{S}_1 \cup \cdots \cup \mathcal{S}_k)$;
4 Cost $\leftarrow (0, \infty, \ldots, \infty)$;
5 Partition $\leftarrow$ (NULL, ..., NULL);
6 **foreach** $\mathbf{s}_\theta \in \mathcal{S}$ **do**
7 $\quad \theta' = \theta - 1$;
8 $\quad$ **while** `incomplete` $(\mathbf{s}_{\theta'}, \ldots, \mathbf{s}_\theta)$ **do**
9 $\quad\quad \theta' = \theta' - 1$;
10 $\quad$ **while** `elementary` $(\mathbf{s}_{\theta'}, \ldots, \mathbf{s}_\theta)$ **do**
11 $\quad\quad \mathcal{G} \leftarrow$ `generalize` $(\mathbf{s}_{\theta'}, \ldots, \mathbf{s}_\theta)$;
12 $\quad\quad$ **if** Cost$[\theta] >$ Cost$[\theta' - 1]$ + $c(\mathcal{G})$ **then**
13 $\quad\quad\quad$ Cost$[\theta] \leftarrow$ Cost$[\theta' - 1]$ + $c(\mathcal{G})$;
14 $\quad\quad\quad$ Partition $\leftarrow (\theta' - 1, \mathcal{G})$;
15 $\quad\quad \theta' = \theta' - 1$;
16 $\mathbf{G}^* \leftarrow$ `visit` (Partition);
17 $C(\mathbf{G}^*) \leftarrow$ Cost$[|\mathcal{S}|]$;

Comparing (11) with (10), we get that $C(\mathbf{G}) \geq C(\widetilde{\mathbf{G}})$. Thus, to search for the optimal $\mathbf{G}^*$, we can drop $\mathbf{G}$ and keep only $\widetilde{\mathbf{G}}$. If $\widetilde{\mathbf{G}}$ is not elementary, then we can find one of its refinements, and repeat the above steps to drop also $\widetilde{\mathbf{G}}$. This way, we can drop all partitions that are not elementary and be left only with elementary partitions as $\mathbf{G}^*$ candidates. ■

If we build a tree of partitions belonging to $\mathcal{K}$, such that the $\mathcal{S}$ is the root and each node is a partition whose children are its refinements, the leaves are the elementary partitions, which form a subset $\mathcal{K}^*$. The above proposition states that we can limit the search of $\mathbf{G}^*$ to $\mathcal{K}^*$, drastically reducing the search space of $\mathbf{G}^*$ to the set $\mathcal{K}^* \subset \mathcal{K}$ of elementary partitions of $\mathcal{S}$. An example is provided in Fig. 3, for the trajectories in Fig. 2.

D. Optimal merging algorithm

We propose **`k-merge`**, an algorithm to efficiently search the set of raw samples $\mathcal{S}$, extract the subset of elementary partitions, $\mathcal{K}^*$, and identify the optimal partition $\mathbf{G}^*$.

The algorithm, detailed in Alg. 1, starts by populating a set of raw samples $\mathcal{S}$, whose items $\mathbf{s}_{i,j}$ are ordered according to their time value $t(\mathbf{s}_{i,j})$ (lines 1–3). Then, it processes all samples according to their temporal ordering (line 6). Specifically, the algorithm tests, for each sample $\mathbf{s}_\theta$ in position θ, all sets $\{\mathbf{s}_{\theta'}, \ldots, \mathbf{s}_\theta\}$, with $\theta' < \theta$, as follows.

The first loop skips incomplete sets that do not contain at least one sample from each input trajectory (line 8). The second loop runs until the first non-elementary set is encountered (line 10). Therein, the algorithm generalizes the current (complete and elementary) set $\{\mathbf{s}_{\theta'}, \ldots, \mathbf{s}_\theta\}$ to $\mathcal{G}$, and checks if $\mathcal{G}$ reduces the total merging cost up to $\mathbf{s}_\theta$. If so, the cost is updated by summing $c(\mathcal{G})$ to the accumulated cost up to $\mathbf{s}_{\theta'-1}$, and the resulting (partial) partition of $\mathcal{S}$ that includes $\mathcal{G}$ is stored (lines 11–14). Once out of the loops, the cost associated to the last sample is the optimal cost, and it is sufficient to backward navigate the partition structure to retrieve the associated $\mathbf{G}^*$ (lines 16–17).

Note that, in order to update the cost of including the current sample $\mathbf{s}_\theta$ (line 13), the algorithm only checks previous samples in time. It thus needs that the optimal decision up to $\mathbf{s}_\theta$ does not depend on any of the samples in the original trajectories that come later in time than $\mathbf{s}_\theta$. The following proposition guarantees that this is the case.

Proposition 2: Let $\mathbf{G}^* = (\mathcal{G}_1^*, \ldots, \mathcal{G}_Z^*)$ be the optimal generalized trajectory and let us make the hypothesis that $\mathbf{s}_\theta$ and $\mathbf{s}_{\theta+1}$ do not belong to the same generalized sample of $\mathbf{G}^*$. Let $\mathbf{G}_\mathrm{p}^* = (\mathcal{G}_1^*, \ldots, \mathcal{G}_{Z_1}^*)$ and $\mathbf{G}_\mathrm{f}^* = (\mathcal{G}_{Z_1+1}^*, \ldots, \mathcal{G}_Z^*)$, so that $\mathbf{s}_\theta \in \mathcal{G}_{Z_1}^*$ and $\mathbf{s}_{\theta+1} \in \mathcal{G}_{Z_1+1}^*$. Then, $\mathbf{G}_\mathrm{p}^*$ can be derived independently of $\mathbf{G}_\mathrm{f}^*$.

Proof: Let $\mathbf{G}$, $\mathbf{G}_\mathrm{p}$ and $\mathbf{G}_\mathrm{f}$ be any generalized sequences containing raw samples $(\mathbf{s}_1, \ldots, \mathbf{s}_N)$, $(\mathbf{s}_1, \ldots, \mathbf{s}_\theta)$ and $(\mathbf{s}_{\theta+1}, \ldots, \mathbf{s}_N)$, respectively. According to the cost definition, we generally have

$$\begin{aligned} \min_{\mathbf{G}} C(\mathbf{G}) &\leq \min_{\mathbf{G}_\mathrm{p}, \mathbf{G}_\mathrm{f}} C((\mathbf{G}_\mathrm{p}, \mathbf{G}_\mathrm{f})) \\ &= \min_{\mathbf{G}_\mathrm{p}} C(\mathbf{G}_\mathrm{p}) + \min_{\mathbf{G}_\mathrm{f}} C(\mathbf{G}_\mathrm{f}), \end{aligned}$$

where $(\mathbf{G}_\mathrm{p}, \mathbf{G}_\mathrm{f})$ is the concatenation of $\mathbf{G}_\mathrm{p}$ and $\mathbf{G}_\mathrm{f}$. However, by virtue of the hypothesis and by construction,

$$\begin{aligned} \min_{\mathbf{G}} C(\mathbf{G}) &= C(\mathbf{G}^*) \\ &= C(\mathbf{G}_\mathrm{p}^*) + C(\mathbf{G}_\mathrm{f}^*) \\ &= \min_{\mathbf{G}_\mathrm{p}} C(\mathbf{G}_\mathrm{p}) + \min_{\mathbf{G}_\mathrm{f}} C(\mathbf{G}_\mathrm{f}) \end{aligned}$$

so that, to minimize $C(\mathbf{G})$ we only need to minimize $C(\mathbf{G}_\mathrm{p})$ and $C(\mathbf{G}_\mathrm{f})$ independently. ■

The above proposition guarantees that the algorithm is exploring all possibilities, and as a result, the cost $C(\mathbf{G}^*)$ returned by **k-merge** is optimal, i.e., it is the minimum loss of granularity necessary to merge the original trajectories.

Note that **k-merge** has a very low complexity in practical cases. Let $l(\theta)$ be the number of sets $\{\mathbf{s}_{\theta'}, \ldots, \mathbf{s}_\theta\}$ that are both complete and elementary for a given θ. Then, the number of computations and comparisons of sample generalization costs that are performed in **k-merge** is $\sum_\theta l(\theta) = |\mathcal{S}|\overline{l}$, where $\overline{l}$ is the average value of $l(\theta)$. If $\overline{l} = \mathcal{O}(1)$, which happens in most trajectory data where the samples of the input trajectories are intercalated in the time axis, then **k-merge** runs in a time $\mathcal{O}(|\mathcal{S}|)$, i.e., linear in the number of samples.

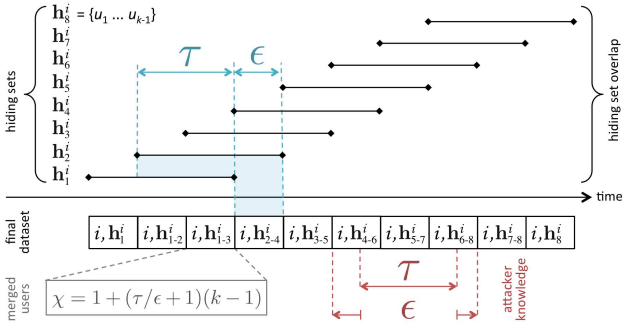


Fig. 4. Overlapping hiding set structure realizing $k^{\tau,\epsilon}$-anonymity for user i.

E. Single user $k^{\tau,\epsilon}$-anonymity

We implement $k^{\tau,\epsilon}$-anonymity for a generic subscriber i as shown in Fig. 4. We discretize time into intervals of length ϵ, named *epochs*. At the beginning of the m-th epoch, we select a set of $k-1$ users different from i, named a *hiding set* of i and denoted as $\mathbf{h}_m^i$. The hiding set $\mathbf{h}_m^i$ provides k-anonymity to subscriber i for a subsequent time window $\tau+\epsilon$. By repeating the hiding set selection for all epochs, $\tau/\epsilon+1$ subsequent hiding sets of user i overlap at any point in time. Such a structure of overlapping hiding sets assures the following.

First, subscriber i is k-anonymized for any possible knowledge of the attacker. No matter where a time interval of length τ is shifted to along the time dimension, it will be always completely covered by the time window of one hiding set, i.e., a period during which i's trajectory is indistinguishable from those of $k-1$ other users. As an example, in Fig. 4, the attacker knowledge τ (bottom-right of the plot) is fully enclosed in the time window of $\mathbf{h}_6^i$, and his sub-trajectory is indistinguishable from those of users in $\mathbf{h}_6^i$.

Second, the additional knowledge leaked to the attacker is exactly ϵ. From the first point above, the adversary cannot tell apart i from the users in the hiding set $\mathbf{h}_m^i$ whose time window covers his knowledge τ. However, the adversary can follow the (generalized) trajectories of i and users in $\mathbf{h}_m^i$ for the full time window $\tau+\epsilon$. Therefore, the adversary can infer new information about the (generalized) trajectory of i during the time window period that exceeds his original knowledge τ, i.e., ϵ. E.g., in Fig. 4, the time window of $\mathbf{h}_6^i$ spans before and after the attacker knowledge τ, for a total of ϵ.

The two guarantees above let $k^{\tau,\epsilon}$-anonymity, as defined in Sec. II-C2, be fulfilled for the generic user i. The epoch duration ϵ maps to the knowledge leakage. The following important remarks are in order.

1. Hiding set selection. The structure of overlapping hiding sets is to be implemented so that the loss of accuracy in the $k^{\tau,\epsilon}$-anonymized trajectory is minimized. Thus, the users in the generic hiding set $\mathbf{h}_m^i$ shall be those who, during the time window $\tau+\epsilon$ starting at the m-th epoch, have sub-trajectories with minimum **k-merge** cost with respect to i's.

2. Reuse constraint. The uninformative principle requires alternating the $k-1$ trajectories used in different hiding sets, as per Sec. II-C2. A simple way to enforce this is limiting the inclusion of any subscriber in at most one hiding set of i.

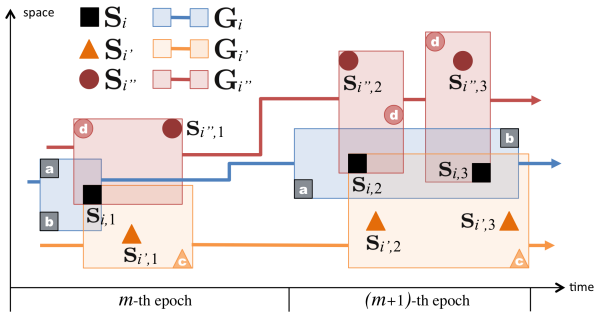


Fig. 5. Example of k-pick constraint, with k=3, for user i during the m-th hiding set selection. Here $\epsilon = \tau$, hence the time windows of hiding sets span two epochs. For clarity, space is unidimensional. Figure best viewed in colors.

3. Generalization set. As evidenced by the example in Fig. 4, the configuration of hiding sets changes at every epoch, and $\tau/\epsilon + 1$ hiding sets overlap during each epoch. This means that a spatiotemporal generalization must be used to merge a set of $\chi = 1 + (\tau/\epsilon + 1)(k-1)$ trajectories at each epoch.

4. Epoch duration tradeoff. The epoch duration ϵ is a configurable system parameter, whose setting gives rise to a tradeoff between knowledge leakage and accuracy of the anonymized data. A lower ϵ reduces knowledge leakage. However, it also increases χ, which typically entails a more marked generalization and a higher loss of data granularity.

F. Multiple user $k^{\tau,\epsilon}$-anonymity

Scaling $k^{\tau,\epsilon}$-anonymity from a single user to all subscribers in a dataset implies that the choice of hiding sets cannot be made independently for every user. Therefore, trajectory similarity and reuse constraint fulfillment are not sufficient norms anymore. In addition to the above, the selection of hiding sets needs to be concerted among all users so as to ensure that the generalized trajectories are correctly intertwined and all subscribers are k-anonymized during each time window $\tau+\epsilon$.

An intuitive solution is enforcing *full consistency*: including a subscriber i into the hiding set of user i' at epoch m makes i' automatically become part of i's hiding set at the same epoch. Formally, $i \in \mathbf{h}_m^{i'} \Rightarrow i' \in \mathbf{h}_m^i$, $\forall i \neq i', \forall m$.

In fact, full consistency is an unnecessarily restrictive condition. It is sufficient that hiding set concertation satisfies a *k-pick constraint*: during the m-th epoch, each user i in the dataset has to be picked in the hiding sets of at least other $k-1$ subscribers. Formally, $|\{i',\ i \in \mathbf{h}_m^{i'}\}| \geq k-1$, $\forall i, \forall m$. This provides an increased flexibility over all existing approaches which rely on fully consistent generalization strategies.

The rationale behind the k-pick constraint is best illustrated by means of a toy example, in Fig. 5. The figure portrays the spatiotemporal samples of users i, i' and i'' during epochs m and $m+1$. The sub-trajectory of subscriber i in this time interval is $\mathbf{S}_i = (\mathbf{s}_{i,1}, \mathbf{s}_{i,2}, \mathbf{s}_{i,3})$, represented as black squares; equivalently for i' (orange triangles) and i'' (red circles). Samples denoted by letters belong to other users a, b, c and d, and they are instrumental to our example.

Let us assume that $\epsilon = \tau$ (i.e., hiding sets span an interval $2\tau = 2\epsilon$, or epochs m and $m+1$), and $k = 3$. At the beginning of the m-th epoch, for subscriber i (resp., i' and i''), one needs to select $k-1 = 2$ other users that constitute the hiding set $\mathbf{h}_m^i$ (resp., $\mathbf{h}_m^{i'}$ and $\mathbf{h}_m^{i''}$). Let us consider $\mathbf{h}_m^i$=$\{a, b\}$, $\mathbf{h}_m^{i'}$=$\{i, c\}$, $\mathbf{h}_m^{i''}$=$\{i, d\}$, which results in the generalized sub-trajectories $\mathcal{G}_i$, $\mathcal{G}_{i'}$, $\mathcal{G}_{i''}$ in Fig. 5. The configuration satisfies the k-pick constraint for subscriber i, who is picked in $k-1 = 2$ hiding sets, i.e., $\mathbf{h}_m^{i'}$ and $\mathbf{h}_m^{i''}$. Suppose now that the attacker knows the spatiotemporal samples of i's trajectory during any time interval τ within the m-th and $(m+1)$-th epoch: as these samples are within $\mathcal{G}_i$, $\mathcal{G}_{i'}$ and $\mathcal{G}_{i''}$, then i is 3-anonymized.

The key consideration is that i is k-anonymized at epoch m by i' and i'', yet it does not contribute to the anonymization of neither i' nor i'', as $i', i'' \notin \mathbf{h}_m^i$. Thus, it is possible to decouple the choice of hiding sets across subscribers, without jeopardizing the privacy guarantees granted by k-anonymity. Such a decoupling entails a dramatic increase of flexibility in the choice of hiding sets, as per the following proposition.

Proposition 3: Given a dataset of U trajectories and a fixed value of k, the number of hiding set configurations allowed by full consistency is a fraction of that allowed by k-pick that vanishes more than exponentially for $U \to \infty$.

Proof: Let us consider a set of U users, where U is a multiple of k, since otherwise full consistency cannot even be enforced. Let us build a $k \times U$ matrix, in which the i-th column contains $(i, \mathbf{h}_m^i)$, where $\mathbf{h}_m^i$ is the hiding set for user i at a given epoch m. (For simplicity, in this proof, we do not take into account the reuse constraints.)

The solution set under the k-pick constraint coincides with the set of normalized Latin rectangles[3] of size $k \times U$. Let $K_{k,U}$ be the number of $k \times U$ normalized Latin rectangles, which equals the number of possible solutions for our problem with the k-pick constraint. An old result by Erdős and Kaplansky [20] states that, for $U \to \infty$ and $k = O\left((\log U)^{3/2-\epsilon}\right)$,

$$K_{k,U} \sim (U!)^{k-1} \exp\left(-k(k-1)/2\right) \tag{12}$$

If, instead, we enforce full consistency, then the number of solutions equals the number of different partitions of a size-U set into U/k subsets, all with size k. Denoting by $C_{k,U}$ this number, we can compute it as

$$C_{k,U} = \frac{\binom{U}{k}\binom{U-k}{k} \cdot \dots \cdot \binom{k}{k}}{(U/k)!} = \frac{U!}{(k!)^{U/k}\,(U/k)!} \tag{13}$$

Thus, for fixed k and $U \to \infty$

$$\frac{C_{k,U}}{K_{k,U}} \sim \frac{\exp\left(k(k-1)/2\right)}{(U!)^{k-2}(k!)^{U/k}\,(U/k)!}$$

which tends to zero more than exponentially for $U \to \infty$. ■

For large datasets of hundreds of thousands trajectories, k-pick enables a much richer choice of merging configurations. This reasonably unbinds better combinations of the original trajectories, and results in more accurate anonymized data.

[3] A $k \times n$ *Latin rectangle*, $k \leq n$, is a matrix in which all entries are taken from the set $\{1, \dots, n\}$, in such a way that each row and column contains each value at most once. The Latin rectangle is said to be normalized if the first row is the ordered set $(1, \dots, n)$.

Algorithm 2: **kte-hide** algorithm pseudocode.

input : Anonymization level k, attacker knowledge τ, leakage ϵ
input : Trajectory dataset $\mathbb{D}$
output: Anonymized trajectory dataset $\mathbb{D}$

1 **foreach** $e_\theta \in$ epochs ($\mathbb{D}$) **do**
2 $\mathbb{D}_f \leftarrow$ filter ($e_\theta, \mathbb{D}$);
3 **foreach** $\mathbf{S}_i, \mathbf{S}_{i'} \in \mathbb{D}_f, \mathbf{S}_i \neq \mathbf{S}_{i'}$ **do**
4 Costs [$\mathbf{S}_i, \mathbf{S}_{i'}$] $\leftarrow$ **k-merge** ($\mathbf{S}_i, \mathbf{S}_{i'}$);
5 Clusters [θ] $\leftarrow$ spectralClustering (Costs);
6 **if** $\theta \geq \tau/\epsilon + 1$ **then**
7 **foreach** c $\in$ Clusters [θ] **do**
8 Subs $\leftarrow$ split (c,Clusters [$\theta - \tau/\epsilon : \theta - 1$]);
9 **foreach** $\mathsf{c}_s \in$ Subs [θ] **do**
10 $\mathsf{g}_s \leftarrow$ graph (c_s);
11 $\mathsf{g}_{sc} \leftarrow$ greedyCycle (g_s,k);
12 **if** $\exists \mathsf{g}_{sc}$ **then**
13 **foreach** $\mathbf{S}_i \in \mathsf{c}_s$ **do**
14 $\mathbf{h}^i_{\theta-\tau/\epsilon} \leftarrow \mathsf{g}_{sc}[\mathbf{S}_i]$;
15 **else**
16 suppression (c_s);
17 **foreach** $e_\theta \in$ epochs ($\mathbb{D}$) **do**
18 **foreach** $\mathbf{S}_i \in \mathbb{D}$ **do**
19 $\mathbf{h} \leftarrow$ filter ($e_\theta, \mathbf{S}_i, \mathbf{h}^i_{\theta-\tau/\epsilon}, \ldots, \mathbf{h}^i_\theta$);
20 $\mathbb{D} \leftarrow$ replace (**k-merge** ($\mathbf{h}$));

G. Practical $k^{\tau,\epsilon}$-anonymity algorithm

Capitalizing on all previous results, we design **kte-hide**, an algorithm that achieves $k^{\tau,\epsilon}$-anonymity in datasets of spatiotemporal trajectories. Since even the optimal solution to the simpler k-anonymity problem is known to be NP-hard [14], we resort here to an heuristic solution.

The algorithm, in Alg. 2, proceeds on a per-epoch basis (line 1), finding, for each epoch θ, a set of χ users (with χ defined as in Sec. III-E) that hide each subscriber at low merging cost. An extensive search for the set of χ users would have an excessive cost $\mathcal{O}(U^\chi)$, where U is the number of users in dataset, and $\chi \geq 3$. Thus, we adopt a computationally efficient approach, by clustering user sub-trajectories based on their pairwise merging cost. Costs are computed via **k-merge** (lines 2–4), and a standard spectral clustering algorithm groups similar trajectories into same clusters (line 5). This allows operating on each cluster independently in the following.

Starting from epoch $\tau/\epsilon+1$ (line 6), the algorithm processes each identified cluster at epoch θ separately (line 7). It splits the current cluster c into subsets, which contain user trajectories that share the same sequence of clusters during the last τ/ϵ epochs (line 8).

Let c_s be any of such subsets: c_s is mapped to a directed graph whose nodes are the users within c_s, and there is an edge going from user j to user i if j can be in the hiding set $\mathbf{h}^i_{\theta-\tau/\epsilon}$ of i without violating the reuse constraint (line 10). If a k-anonymity level is required, $k-1$ directional cycles are then built within the graph, involving all nodes in the graph, in such a way that each node has a different parent in each cycle (line 11). The hiding set $\mathbf{h}^i_{t-\tau/\epsilon}$ is then obtained as the set of user i's parents in the $k-1$ cycles (lines 13–14).

Such a construction of hiding sets complies with the k-pick constraint, since every user i is in the hiding set of $k-1$ other users. It may however happen that no valid $k-1$ cycles can be created within c_s: this means that subscribers in c_s share a sub-trajectory that is rare in the dataset, and their number is insufficient to implement $k^{\tau,\epsilon}$-anonymity. In this case, we apply suppression and remove all spatiotemporal samples of such users' sub-trajectories (line 16). Once all hiding sets are determined, the merging is performed, on each epoch and for each user, using **k-merge** (lines 17–20).

Overall, the heuristic algorithm above guarantees that overlapping hiding sets that satisfy the reuse constraint (Sec. III-E) are selected for all users. It also ensures that such a choice of hiding sets fulfils the k-pick requirement (Sec. III-F). Together, these conditions realize $k^{\tau,\epsilon}$-anonymity of the trajectory data.

The complexity of **kte-hide** is as follows. Let U be the number of users, Θ be the number of epochs and $\overline{\mathcal{N}}$ be the average number of samples per user per epoch, so that $\mathcal{N}_{tot} = \Theta U \overline{\mathcal{N}}$ is the total number of samples in the dataset. Then: *(i)* lines 2–4 perform **k-merge** on two input trajectories ΘU^2 times, each of them with a complexity $\mathcal{O}(\overline{\mathcal{N}})$, for a total complexity of $\mathcal{O}(\mathcal{N}_{tot}U)$; *(ii)* spectral clustering (line 5) can be implemented with complexity $\mathcal{O}(\Theta U^2)$ using KASP [21]; *(iii)* the complexity of lines 17–20, performing **k-merge** on χ input trajectories ΘU times, is $\mathcal{O}(\mathcal{N}_{tot}\chi)$. All other subroutines of **kte-hide** have a much smaller complexity.

IV. Performance evaluation

We evaluate our anonymization solutions with five real-world datasets of mobile subscriber trajectories, introduced in Sec. IV-A. A comparative evaluation of **k-merge** is in Sec. IV-B, while the results of $k^{\tau,\epsilon}$-anonymization via **kte-hide** are presented in Sec. IV-C.

A. Reference datasets

Our datasets consist of user trajectories extracted from call detail records (CDR) released by Orange within their D4D Challenges [22], and by the University of Minnesota [23]. Three datasets, denoted as abi, dak and shn, describe the spatiotemporal trajectories of tens of thousands mobile subscribers in urban regions, while the other two, civ and sen hereinafter, are nationwide. In all datasets, user positions map to the latitude and longitude of the current base station (BS) they are associated to. The main features of the datasets are listed in Tab. I, revealing the heterogeneity of the scenarios.

In order to ensure that all datasets yield a minimum level of detail in the trajectory of each tracked subscriber, we had to preprocess the abi and civ datasets. Specifically, we only retained those users whose trajectories have at least one spatiotemporal sample on every day in a specific two-week period. No filtering was needed for the dak and sen datasets, which already contain users who are active for more than 75% of a 2-week timespan, and shn, whose users have even higher sampling rates.

In all datasets, user positions map to the latitude and longitude of the current base station (BS) they are associated to. We discretized the resulting positions on a 100-m regular grid,

TABLE I
FEATURES OF REFERENCE MOBILE TRAFFIC DATASETS.

Dataset	Surface [Km^2]	BS	BS/Km^2	Users	Density [user/Km^2]	Samples [per user/h]	Timespan [days]
abi	2,731	400	0.14	29,191	10.68	0.90	14
dak	1,024	457	0.44	71,146	69,47	0.74	14
shn	3,329	2961	0.89	50,000	15.01	1.00	1
civ	322,463	1238	0.0038	82,728	0.26	0.75	14
sen	196,712	1666	0.0085	286,926	1.45	0.45	14

TABLE II
COMPARATIVE PERFORMANCE EVALUATION OF **K-MERGE**

Dataset	k	**k-merge**		Static generalization [success %]			W4M				GLOVE	
		Time [min]	Space [Km]	2h - 4Km	4h - 10Km	8h - 20Km	Deleted [%]	Created [%]	Time [min]	Space [Km]	Time [min]	Space [Km]
abi	2	51	0.624	27.2	56.7	80.3	9.6	22.0	57	1.166	114	2.626
	5	228	3.423	0.7	11.0	40.5	31.9	31.2	185	3.809	292	3.740
	8	349	5.720	0.1	5.1	22.6	23.9	36.7	198	6.163	—	—
dak	2	47	0.701	43.2	68.7	93.3	5.9	11.4	39	1.466	116	2.498
	5	220	5.286	2.2	14.0	67.0	20.3	21.2	172	5.807	294	3.192
	8	377	7.794	0.1	8.6	50.7	22.0	18.6	189	8.477	—	—

which represents the finest spatial granularity we consider[4].

Samples are timestamped with an precision of one minute. This is the granularity granted in the abi and civ datasets. The dak and sen datasets feature a temporal granularity of 10 minutes: in order to have comparable datasets, we added a random uniform noise over a ten-minute timespan to each sample, so as to artificially refine the time granularity of the data to one minute as well. In the case of the shn dataset, the precision is one second, and we used a one-minute binning to uniform the data to the standard format.

B. Comparative evaluation of *k-merge*

Since no previous solution for $k^{\tau,\epsilon}$-anonymity exists, we are forced to compare our algorithms to previous techniques in terms of simpler k-anonymity. Interestingly, this allows validating our proposed approach for merging spatiotemporal trajectories via the **k-merge** algorithm.

We thus run **k-merge** on 100 random k-tuples of mobile users from the reference datasets, for different values of k, and we record the spatiotemporal granularity retained by the resulting generalized trajectories. We compare our results against those obtained by the only three approaches proposed in the literature for the k-anonymization of trajectories along both spatial and temporal dimensions.

The first is static generalization [8], [9], which consists in a homogeneous reduction of data granularity, decided arbitrarily and imposed on all user trajectories. Static generalization is a trial-and-error process, and it does not guarantee k-anonymity of all users. The second benchmark solution is Wait for Me (W4M) [36]. Intended for regularly sampled (e.g., GPS) trajectories, W4M performs the minimum spatiotemporal translation needed to push all the trajectories within the same cylindrical volume. It allows the creation of new synthetic samples, and it is thus not fully compliant with PPDP principles in Sec. II-A. The latter operation is leveraged to improve the matching among trajectories in a cluster, and assumes that mobile objects (i.e., subscribers in our case) effectuate linear constant-speed movements between spatiotemporal samples. We use W4M with linear spatiotemporal distance (W4M-L), i.e., the version intended for large databases such as those we consider [5], and configure it with the settings suggested in [36]. The third approach is GLOVE [10], which relies on a heuristic measure of anonymizability to assess the similarity of spatiotemporal trajectories. This measure is fed to a greedy algorithm to achieve k-anonymity with limited loss of granularity and without introducing fictitious data. However, unlike **k-merge**, GLOVE does not provide an optimal solution, and is computationally expensive.

The results of our comparative evaluation are summarized in Tab. II, for the abi and dak datasets, when varying number k of trajectories merged together. Similar results were obtained for the other datasets, and are omitted due to space limitations. We immediately note how static aggregation is an ineffective approach: the percentage of successfully merged k-tuples is well below 100%, even when dramatically reducing the data granularity to 8 hours in time and 20 km in space. Instead, **k-merge**, W4M and GLOVE can merge all of the k-tuples, while retaining a good level of accuracy in the data. We can directly compare the granularity in time (min) and space (km) retained by **k-merge**, W4M and GLOVE in merging groups of k trajectories: the spatiotemporal accuracy is comparable in all cases. However, it is important to note that W4M attains this result by deleting and creating a significant amount of samples: in the end, only 40-70% of the original samples are maintained in the generalized data. Conversely, all of the generalized samples created by **k-merge** reflect the actual real-world data. Also, **k-merge** obtains a level of precision that is always higher than that of GLOVE, and scales better: indeed, the complexity of GLOVE did not allow computing a solution when $k = 8$.

Overall, the results uphold **k-merge** as the current state-of-the-art solution to generalize sparse spatiotemporal trajectories while obeying PPDP principles and minimizing accuracy loss.

C. Performance evaluation of *kte-hide*

We run **kte-hide** on our reference datasets of mobile subscriber trajectories, so that they are $k^{\tau,\epsilon}$-anonymized. As the anonymized data are robust to probabilistic attacks by design, we focus our evaluation on the cost of the anonymization, i.e., the loss of granularity. All results refer to the case of $2^{\tau,\epsilon}$-anonymization, with $\epsilon = \tau$.

1) Citywide datasets: Fig. 6 portrays the mean, median and first/third quartiles of the sample granularity in the $k^{\tau,\epsilon}$-anonymized citywide datasets abi, dak and shn. The plots show how results vary when the adversary knowledge τ ranges

[4]At 100-m spatial granularity, each grid cell contains at most one antenna from the original dataset: the process does not cause any loss in data accuracy.

[5]Implementation at http://kdd.isti.cnr.it/W4M/.

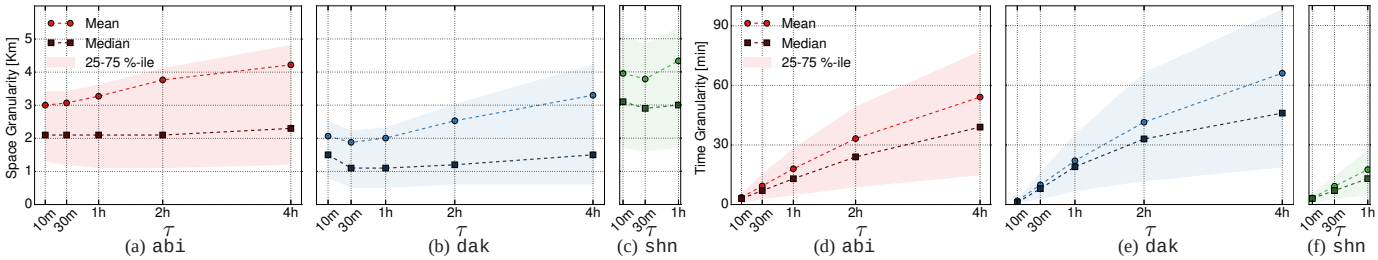


Fig. 6. Spatial (a,b,c) and temporal (d,e,f) granularity versus the adversary knowledge τ in the citywide reference datasets.

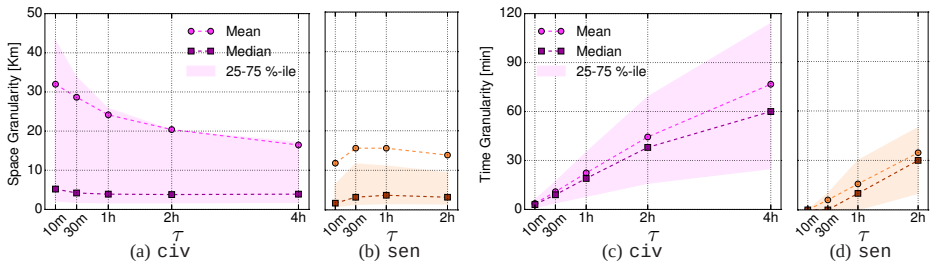


Fig. 7. Spatial (a,b) and temporal (c,d) granularity versus τ in the nationwide reference datasets.

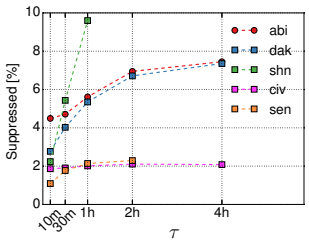


Fig. 8. Suppressed samples versus τ.

from 10 minutes to 4 hours[6]. They refer to the anonymized data granularity in space[7], in Fig.6a-c and time, in Fig.6d-f.

We remark how the $k^{\tau,\epsilon}$-anonymized datasets retain significant levels of accuracy, with a median granularity in the order of 1-3 km in space and below 45 minutes in time. These levels of precision are largely sufficient for most analyses on mobile subscriber activities, as discussed in, e.g., [24]. The temporal granularity is negatively affected by an increasing adversary knowledge τ, which is expected. Interestingly, however, the spatial granularity is only marginally impacted by τ: protecting the data from a more knowledgeable attacker does not have a significant cost in terms of spatial accuracy.

2) Nationwide datasets: Fig. 7 shows equivalent results for the nationwide datasets `civ` and `sen`. The evolution of temporal granularity versus τ, in Fig.7c-d is consistent with citywide scenarios. Differences emerge in terms of spatial granularity: in the `civ` case (Fig.7a) a reversed trend emerges, as accuracy grows along with the attacker knowledge. This counterintuitive result is explained by the thin user presence in the `civ` dataset: as per Tab. I, `civ` has a density of subscribers per Km^2 that is one or two orders of magnitude lower than those in our other reference datasets. Such a geographical sparsity makes it difficult to find individuals with similar spatial trajectories: increasing τ has then the effect of enlarging the set of candidate trajectories for merging at each epoch, with a positive influence on the accuracy in the generalized data.

These considerations are confirmed by the results with the `sen` dataset (Fig.7b). As per Tab. I, this dataset features a subscriber density that is about one order of magnitude higher than that of `civ`, but around one order of magnitude lower than those of the `abi`, `dak` and `shn`. Coherently, the spatial granularity trend falls in between those observed for such datasets, and it is not positively or negatively impacted by the attacker knowledge.

More generally, the results in Fig. 7 demonstrate that **`kte-hide`** can scale to large-scale real-world datasets. The absolute performance is good, as the $k^{\tau,\epsilon}$-anonymized data retains substantial precision: the median levels of granularity in space and time are comparable to those achieved in citywide datasets. Finally, we remark that, in all cases, the amount of samples suppressed by **`kte-hide`** is in the 1%–7% range.

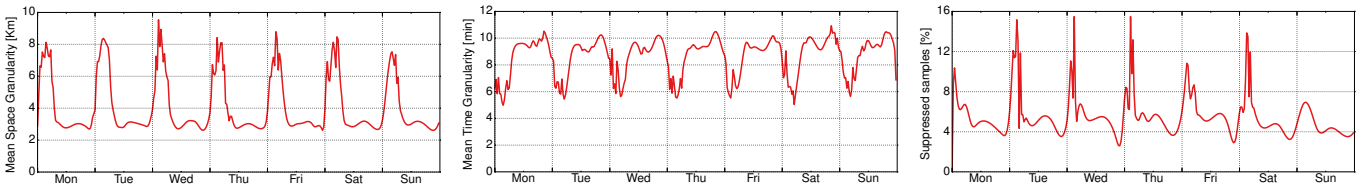
3) Sample suppression: The amount of samples suppressed by **`kte-hide`** in the $k^{\tau,\epsilon}$-anonymization process is portrayed in Fig. 8. We note that resorting to suppression becomes more frequent as the adversary knowledge increases. However, even when the opponent is capable of tracking a user during four continued hours, the percentage of suppressed samples remains low, typically well below 10%. Moreover, the trend in the long-timespan datasets is clearly sublinear, suggesting that suppression does not become prevalent with higher τ. Results are fairly consistent across citywide datasets[8]. Nationwide datasets are also aligned, and yield even lower suppression rates, at around 2%. This difference is explained by the fact that a larger number of users allows for a more efficient spectral clustering in **`kte-hide`**.

4) Disaggregation over time: As an intriguing concluding remark, Fig. 9 reveals a clear circadian rhythm in the granularity of $k^{\tau,\epsilon}$-anonymized data, as well as in the percentage of suppressed samples. The plots refer to one sample week in the `abi` and `dak` datasets, when τ = 30 min, but consistent results were observed in all of our reference datasets. Specifically,

[6]The limited temporal span of the `shn` data prevents us from testing attacks with knowledge τ higher than one hour. Indeed, a τ too close to the full dataset duration implies that the opponent has an a-priori knowledge of the victim's trajectory that is comparable to that contained in the data, making attempts at countering a probabilistic attack futile.

[7]The spatial granularity in Fig. 6 is expressed as the sum of spans along the Cartesian axes. For instance, 1 km maps to, e.g., a square of side 500 m.

[8]The spurious point at τ = 1 hour in `shn` is due to the fact that the time interval $\tau + \epsilon$ is already very large, at around the same order of magnitude of the full dataset duration.



(a) `abi`, space (b) `abi`, time (c) `abi`, suppression

Fig. 9. Time series of spatiotemporal accuracy (a,b) and suppression usage (c) for one sample week in the `abi` dataset.

the mean spatial granularity, in Fig. 9a, is much finer during daytime, when subscribers are more active and the volume of trajectories is larger: here, it is easier to hide a user into the crowd. Overnight displacements are instead harder to anonymize, since subscribers are limited in number and they tend to have diverse patterns. This is also corroborated by the significantly higher suppression of samples between midnight and early morning, in Fig. 9c. Time granularity, in Fig. 9b, is less subject to day-night oscillations: the slightly higher accuracy recorded at night is an artifact of the important relative suppression of samples at those times.

5) Summary: Overall, our results show that **`kte-hide`** attains $k^{\tau,\epsilon}$-anonymity of real-world datasets of mobile traffic, while maintaining a remarkable level of accuracy in the data. Interestingly, its performance is better when most needed, at daytime, when the majority of human activities take place.

V. Related work

Protection of individual mobility data has attracted significant attention in the past decade. However, attack models and privacy criteria are very specific to the different data collection contexts. Hence, solutions developed for a specific type of movement data are typically not reusable in other environments.

For instance, a vast amount of works have targeted user privacy in location-based services (LBS). There, the goal is ensuring that single georeferenced queries are not uniquely identifiable [25]. This is equivalent to anonymizing each spatiotemporal sample independently, and a whole other problem from protecting full trajectories. Even when considering sequences of queries, the LBS milieu allows pseudo-identifier replacement, and most solutions rely on this approach, see, e.g., [26], [27]. If applied to spatiotemporal trajectories, these techniques would seriously and irreversibly break up trajectories in time, disrupting data utility.

Another popular context is that of spatial trajectories that do not have a temporal dimension. The problem of anonymizing datasets of spatial trajectories has been thoroughly explored in data mining, and many practical solutions based on generalization have been proposed, see, e.g., [28]–[31]. Such solutions are not compatible with or easily extended to the more complex spatiotemporal data we consider.

Some works explicitly target privacy preservation of spatiotemporal trajectories. However, the precise context they refer to makes again all the difference. First, most such solutions consider scenarios where user movements are sampled at regular time intervals that are identical for all individuals [32], [33], or where the number of samples per device is very small [34]. These assumptions hold, e.g., for GPS logs or RFID record, but not for trajectories recorded by mobile operators: the latter are irregularly sampled, temporally sparse, and cover long time periods, which results in at least hundreds of samples per user. Second, many of the approaches above disrupt data utility, by, e.g., trimming trajectories [35], or violate the principles of PPDP, by, e.g., perturbating or permutating the trajectories [32], [33], or creating fictitious samples [36]. Third, all previous studies aim at attaining k-anonymity of spatiotemporal trajectories, i.e., they protect the data against record linkage; this includes recent work specifically tailored to mobile subscriber trajectory datasets [10]. As explained in Sec. II, k-anonymity is only a partial countermeasure to attacks on spatiotemporal trajectories.

Provable privacy guarantees are instead offered by *differential privacy*, which commends that the presence of a user's data in the published dataset should not change substantially the output of the analysis, and thus formally bounds the privacy risk of that user [37]. There have been attempts at using differential privacy with mobility data. Specifically, it has been successfully used the in the LBS context, when publishing aggregate information about the location of a large number of users, see, e.g., [38]. However, the requirements of these solutions already become too strong in the case of individual LBS access data [39]. To address this problem, a variant of differential privacy, named *geo-indistinguishability* has been introduced: it requires that any two locations become more indistinguishable as they are geographically closer [40]. Practical mechanisms achieve geo-indistinguishability, see, e.g., [39], [40]. However, all refer to the anonymization of single LBS queries: as of today, differential privacy and its derived definitions still appear impractical in the context of spatiotemporal trajectories.

VI. Conclusions

In this paper, we presented a first PPDP solution to probabilistic and record linkage attacks against mobile subscriber trajectory data. To that end, we introduced a novel privacy model, $k^{\tau,\epsilon}$-anonymity, which generalizes the popular criterion of k-anonymity. Our proposed algorithm, **`kte-hide`**, implements $k^{\tau,\epsilon}$-anonymity in real-world datasets, while retaining substantial spatiotemporal accuracy in the anoymized data.

REFERENCES

[1] K. Zheng, Z. Yang, K. Zhang, P. Chatzimisios, K. Yang, W. Xiang, "*Big data-driven optimization for mobile networks toward 5G*," IEEE Network, 30(1), 2016.

[2] M. Leconte, G. Paschos, L. Gkatzikis, M. Draief, S. Vassilaras, S. Chouvardas, "*Placing Dynamic Content in Caches with Small Population*," IEEE INFOCOM, 2016.

[3] Telefonica Smart Steps, http://dynamicinsights.telefonica.com/smart-steps/.

[4] Orange Flux Vision, http://www.orange-business.com/fr/produits/flux-vision.

[5] D. Naboulsi, M. Fiore, R. Stanica, S. Ribot, "*Large-scale Mobile Traffic Analysis: a Survey*," IEEE Communications Surveys and Tutorials, 18(1), 2016.

[6] M. T. Asif, N. Mitrovic, J. Dauwels, P. Jaillet, "*Matrix and Tensor Based Methods for Missing Data Estimation in Large Traffic Networks*," IEEE Transactions on ITS, 17(7), 2016.

[7] G. Czibula, A. M. Guran, I. G. Czibula, G. S. Cojocar, "*IPA - An intelligent personal assistant agent for task performance support*," IEEE ICCP, 2009.

[8] H. Zang, J. Bolot, "*Anonymization of location data does not work: A large-scale measurement study*," ACM MobiCom, 2011.

[9] Y. de Montjoye, C.A. Hidalgo, M. Verleysen, V. Blondel, "*Unique in the Crowd: The privacy bounds of human mobility*," Nature Scientific Reports, 3(1376), 2013.

[10] M. Gramaglia, M. Fiore, "*Hiding Mobile Traffic Fingerprints with GLOVE*," ACM CoNEXT, 2015.

[11] A. Cecaj, M. Mamei, N. Bicocchi, "*Re-identification of Anonymized CDR datasets Using Social Network Data*," IEEE PerCom Workshops, 2014.

[12] C. Riederer, Y. Kim, A. Chaintreau, N. Korula, S. Lattanzi, "*Linking Users Across Domains with Location Data: Theory and Validation*," ACM WWW, 2016.

[13] J. Mayer, P. Mutchler, J.C. Mitchell, "*Evaluating the privacy properties of telephone metadata*," PNAS, 113(20), 2016.

[14] B.C.M. Fung, K. Wang, R. Chen, P.S. Yu, "*Privacy-preserving data publishing: A survey of recent developments*," ACM Computing Surveys, 42(4), 2010.

[15] L. Sweeney, "*k-anonymity: A model for protecting privacy*," International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 10(5), 2002.

[16] A. Machanavajjhala, D. Kifer, J. Gehrke, M. Venkitasubramaniam, "*l-diversity: Privacy beyond k-anonymity*," ACM Transactions on Knowledge Discovery from Data, 1(1):3, 2007.

[17] R. Shokri, G. Theodorakopoulos, J.-Y. Le Boudec, J.-P. Hubaux, "*Quantifying Location Privacy*," IEEE SP, 2011.

[18] M. Srivatsa, M. Hicks, "*Deanonymizing Mobility Traces: Using Social Networks as a Side-Channel*," AMC CCS, 2012.

[19] M. Terrovitis, N. Mamoulis, P. Kalnis, "*Privacy-preserving Anonymization of Set-valued Data*," VLDB, 2008.

[20] P. Erdős, I. Kaplansky, "*The asymptotic number of Latin rectangles*," Amer. J. Math., 68:230-236, 1946.

[21] D. Yan, L. Huang, M.I. Jordan, "*Fast approximate spectral clustering*," ACM SIGKDD, 2009.

[22] Orange D4D Challenge. http://www.d4d.orange.com/en/.

[23] D. Zhang, J. Huang, Y. Li, F. Zhang, C. Xu, T. He, "*Exploring Human Mobility with Multi-Source Data at Extremely Large Metropolitan Scales*," ACM MobiCom, 2014.

[24] M. Coscia, S. Rinzivillo, F. Giannotti, D. Pedreschi, "*Optimal Spatial Resolution for the Analysis of Human Mobility*," IEEE/ACM ASONAM, 2012.

[25] M. Gruteser, D. Grunwald, "*Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking*," ACM MobiSys, 2003.

[26] J. Meyerowitz, R.R. Choudhury, "*Hiding stars with fireworks: location privacy through camouflage*," ACM MobiCom, 2009.

[27] B. Hoh, M. Gruteser, H. Xiong, A. Alrabady, *Preserving privacy in GPS traces via uncertainty-aware path cloaking*. ACM CSS, 2007.

[28] A. Monreale, G. Andrienko, N. Andrienko, F. Giannotti, D. Pedreschi, S. Rinzivillo, S. Wrobel "*Movement Data Anonymity through Generalization*," Transactions on Data Privacy 3(2), 2010.

[29] M.E. Nergiz, M. Atzori, Y. Saygin, B. Güç "*Towards Trajectory Anonymization: a Generalization-Based Approach*," Transactions on Data Privacy 2(1), 2009.

[30] R. Chen, B.C.M. Fung, B.C. Desai, N.M. Sossou, "*Differentially private transit data publication: a case study on the Montreal transportation system*," ACM KDD, 2012.

[31] G. Poulis, S. Skiadopoulos, G. Loukides, A. Gkoulalas-Divanis, "*Apriori-based algorithms for k^m-anonymizing trajectory data*," Transactions on Data Privacy 7(2), 2014.

[32] J. Domingo-Ferrer, R. Trujillo-Rasúa, "*Microaggregation- and permutation-based anonymization of movement data*," Information Science, 208, 2012.

[33] O. Abul, F. Bonchi, M. Nanni, "*Never walk alone: Uncertainty for anonymity in moving objects databases*," IEEE ICDE, 2008.

[34] B.C.M. Fung, M. Cao, B.C. Desai, H. Xu, "*Privacy protection for RFID data*," ACM SAC, 2009.

[35] Y. Song, D. Dahlmeier, S. Bressan, "*Not So Unique in the Crowd: a Simple and Effective Algorithm for Anonymizing Location Data*," PIR, 2014.

[36] O. Abul, F. Bonchi, M. Nanni, "*Anonymization of moving objects databases by clustering and perturbation*," Information Systems, 35(8), 2010.

[37] C. Dwork "*Differential privacy*," ICALP, 2006.

[38] R. Chen, G. Acs, C. Castelluccia "*Differentially private sequential data publication via variable-length n-grams*," ACM CCS, 2012.

[39] K. Chatzikokolakis, C. Palamidessi, M. Stronati, "*A Predictive Differentially-Private Mechanism for Mobility Traces*," PETS, 2014.

[40] M.E. Andrés, N.E. Bordenabe, K. Chatzikokolakis, C. Palamidessi, "*Geo-indistinguishability: differential privacy for location-based systems*," ACM CCS, 2013.