Privacy-preserving Publication of Trajectories Using Microaggregation

Josep Domingo-Ferrer, Michal Sramka, and Rolando Trujillo-Rasúa
Universitat Rovira i Virgili
UNESCO Chair in Data Privacy
Department of Computer Engineering and Mathematics
Av. Països Catalans 26, E-43007 Tarragona, Catalonia
{josep.domingo,michal.sramka,rolando.trujillo}@urv.cat

ABSTRACT

Huge amounts of movement data are automatically collected by technologies such as GPS, GSM, RFID, etc. Publishing such data is essential to improve transportation, to understand the dynamics of the economy in a region, etc. However, there are obvious threats to the privacy of individuals if their trajectories are published in a way which allows re-identification of the individual behind a trajectory. We contribute to the literature on privacy-preserving publication of trajectories by presenting: i) a distance measure for trajectories which naturally considers both spatial and temporal aspects of trajectories, is computable in polynomial time, and can cluster trajectories not defined over the same time span (something that previously proposed methods could not do); ii) a method to replace a cluster of trajectories by synthetic data that preserve all the visited locations and the number of original trajectories, among other features; iii) a comparison of our method with (k, δ)-anonymity [1] using trajectories generated by the Brinkhoff's generator [4] in the city of Oldenburg.

Categories and Subject Descriptors

K.4 [**Computers and Security**]: Privacy—*anonymization*; H.3.3 [**Information Search and Retrieval**]: Clustering, Information filtering—*clustering and transforming movement data*

General Terms

Security, Algorithms

Keywords

Movement data, Trajectories, Distance, Data privacy, Anonymization, Microaggregation

SPRINGL'10 November 2, 2010, San Jose, CA, USA

1. INTRODUCTION

Various technologies such as GPS, RFID, GSM, etc. can sense and track whereabouts of objects. The ever increasing capacity to store data allows such object movements data to be collected in huge spatio-temporal databases. The study of a database of spatio-temporal data (trajectories) can lead to useful or previously unknown knowledge. Therefore, it is beneficial to share and publish such databases and let the analysts obtain useful knowledge —for example, knowledge that can be applied in intelligent transportation, traffic monitoring and planing, congestion trends, supply chain management, etc. However, the privacy of individuals may be affected by publishing the database or outsourcing it for analysis.

Several kinds of threats to the privacy of individuals associated with publishing databases of trajectories exist. Simple de-identification realized by removing identifying attributes is insufficient to protect the privacy of individuals. The biggest threat with trajectories is the "sensitive location disclosure". In this scenario, knowing the times in which an individual visited a few locations can help an adversary to identify the individual's trajectory in the published database, and therefore learn the individual's other locations at other times. Privacy-preservation in this context means that no sensitive location can be linked to an individual.

The risks of sensitive location disclosure is also affected by how much the adversary knows. The adversary may have access to auxiliary information [11], also sometimes called side knowledge, background knowledge and external knowledge. The adversary can link such prior knowledge from other sources to information in the published database. Capturing the amount and extent of auxiliary information available to the adversary is a challenging task.

There are quite a few differences between spatio-temporal data and microdata, and the real difference becomes apparent when considering privacy. Unfortunately, the traditional anonymization and sanitization methods for microdata [9] cannot be directly applied to spatio-temporal data without considerable expenses in computation time and information loss. Hence, there is a need for specific anonymization methods to thwart privacy attacks and therefore reduce privacy risks associated with publishing trajectories.

1.1 Our contribution

We present a method for preserving the privacy of individuals when releasing spatio-temporal data about their

trajectories. The microaggregation approach [7] has been successfully used in microdata anonymization to achieve k-anonymity [15, 16]. We use a similar approach, first to cluster the trajectories into clusters of size at least k based on their similarity and then transforming the trajectories inside each cluster to preserve privacy.

For clustering purposes, we propose a distance measure for trajectories which naturally considers both spatial and temporal aspects. Many distance measures have been proposed for trajectories and time series (cf. [5]), however none of them seems to satisfy the specifics required in anonymization. The novelty of our distance measure lies in the ability to compare trajectories that are not defined over the same time span, something that previously proposed anonymization methods based on similar principles could not do. Our distance measure can compare trajectories that are timewise overlapping only partially or not at all. It may seem at first sight that the distance computation is exponential in the terms of all considered trajectories, but we show that it is in fact computable in polynomial time.

After forming clusters of trajectories, we propose a *Swap-Triples* method which effectively anonymizes the trajectories in a given cluster. The method swaps locations in space and time. This results in location preservation and as our experiments show also minimal space distortion.

We theoretically and experimentally compare our proposed anonymization method with a recent trajectory anonymization called (k, δ)-anonymity [1]. Theoretical results show that the privacy preservation of our method is the same as that of (k, δ)-anonymity when dealing with trajectories having the same time span, while we can also guarantee privacy for trajectories *not* having the same time span. Our experimental results on 1,000 synthetic trajectories generated by the Brinkhoff's generator [4] in the German city of Oldenburg indicate that, when considering the same distortion, our re-identification probability is smaller.

In summary, our contributions are:

- A distance measure for trajectories which naturally considers both spatial and temporal aspects of trajectories, is computable in polynomial time, and can cluster trajectories not defined over the same time span (something that previously proposed methods could not do);
- A method to replace a cluster of trajectories with synthetic data that preserve all the visited locations and the number of original trajectories, among other features;
- A comparison with (k, δ)-anonymity [1] using trajectories generated by the Brinkhoff's generator [4] in the city of Oldenburg.

1.2 Related work

Several anonymity notions and methods for trajectories have been proposed [10, 1, 13, 17, 14, 18, 12]. Closest to our approach are the k-anonymity like methods that are also similar to the microaggregation approach [7] in their use of clustering. Trajectory (k, δ)-anonymity [1] separately anonymizes trajectories defined over the same time interval by spatial translation —the points of trajectories in a cluster are moved toward the δ radius of the cluster average trajectory. When $\delta = 0$, this is the traditional microaggregation (which replaces trajectories by the cluster centroid) over trajectories having the same time span. Trajectory k-anonymity by [13, 14] considers trajectories consisting of ranges of coordinates and times that are generalized into bounding boxes. Synthetic trajectories are then sampled point-by-point from the obtained boxes. Trajectory k-anonymity of [12] considers trajectories stripped of the time information that are anonymized by first considering regioning and then publishing the centroids of the regions passed by the trajectories. Similarly, the method of [17] considers only sequences of points without the time information.

In contrast to these methods, we perform traditional microaggregation over all original trajectories —we do not specially and separately consider trajectories having the same time span and we consider trajectories over points, not ranges, without stripping the time information. We publish synthetic trajectories which are analogous to condensed/microaggregation-based hybrid data for microdata [2, 6] and for strings [3]. However, our synthetic trajectories can be randomly sampled or can preserve the locations covered by the original trajectories.

Additional related work about anonymization of spatio-temporal data can be found in the literature about location-based services. Anonymity is enforced on individual sensitive locations and the data are anonymized on a per-request basis. Here, we focus on publishing whole spatio-temporal databases rather than providing anonymity for services and protecting individuals from location-based service provides.

2. MICROAGGREGATION-BASED ANONYMIZATION

Our method to anonymize trajectories is partially based on microaggregation [7], and partially on hybrid data [6] and condensation [2]. The method starts by clustering the trajectories based on their similarity and proceeds by transforming the obtained clusters in order to obtain anonymized trajectories. The objective of the clustering (aggregation) is to minimize the similarity measure among trajectories and the constraint is to consider at least k trajectories in a cluster.

Clustering trajectories requires defining a similarity measure —a distance between two trajectories. Because trajectories are distributed over space and time, a distance that considers both spatial and temporal aspects of trajectories is needed. Many distance measures have been proposed in the past for both trajectories of moving objects and for time series (cf. [5]). They include among others the Euclidean distance, Dynamic Time Warp (DTW), Edit distances with real penalty (EDP) or on real sequences (EDR), and Longest common subsequences (LCSS) distance. None of these measures can compare trajectories that do not overlap over time. Therefore we define our own distance measure which can compare trajectories that are timewise overlapping only partially or not at all. We believe this is the right approach in clustering trajectories for anonymization purposes.

2.1 Distance between two trajectories

Let *location* be a triple (t, x, y) with t being a timestamp and (x, y) a point in $\mathbb{R}^2$. Intuitively, the triple denotes that at time t an object is in the position (x, y). A sequence of triples over time form a trajectory.

Definition 1 (Trajectory)**.** A *trajectory* is an ordered set

of locations

$$T = \{(t_1, x_1, y_1), \ldots, (t_n, x_n, y_n)\} \ , \quad (1)$$

where $t_i < t_{i+1}$ for all $1 \leq i < n$.

Definition 2 (p%-contemporary trajectories)**.** Two trajectories

$$T_i = \{(t_1^i, x_1^i, y_1^i), \ldots, (t_n^i, x_n^i, y_n^i)\}$$

and

$$T_j = \{(t_1^j, x_1^j, y_1^j), \ldots, (t_m^j, x_m^j, y_m^j)\}$$

are said to be p%-contemporary if

$$p = 100 \cdot \min(\frac{I}{t_n^i - t_1^i}, \frac{I}{t_m^j - t_1^j})$$

with $I = \max(\min(t_n^i, t_m^j) - \max(t_1^i, t_1^j), 0)$.

Intuitively, two trajectories are 100%-contemporary if and only if they start at the same time and end at the same time; two trajectories are 0%-contemporary trajectories if and only if they occur during non-overlapping time intervals. Denote the overlap time of two trajectories T_i and T_j as $ot(T_i, T_j)$.

Definition 3 (Synchronized trajectories)**.** Given two p%-contemporary trajectories T_i and T_j for some $p > 0$, both *trajectories are said to be synchronized* if they have the same number of triples timestamped within $ot(T_i, T_j)$ and these correspond to the same timestamps. A *set of trajectories is said to be synchronized* if all pairs of p%-contemporary trajectories in it are synchronized, where $p > 0$ may be different for each pair.

We assume that between two locations of a trajectory of an object, the object is moving along a straight line between the locations at a constant speed. Interpolating new locations on a given trajectory is then straightforward. Trajectories can be then synchronized in the sense that if one trajectory has a location at time t, then other trajectories defined at that time will also have a (possibly interpolated) location at time t. This rule guarantees that the set of new locations interpolated in order to synchronize trajectories is of minimum cardinality. Algorithm 1 describes this process.

Algorithm 1 Trajectory synchronization

Require: $\mathcal{T} = \{T_1, \ldots, T_N\}$ a set of trajectories to be synchronized, where each $T_i \in \mathcal{T}$ is of the form:

$$T_i = \{(t_1^i, x_1^i, y_1^i), \ldots, (t_{n^i}^i, x_{n^i}^i, y_{n^i}^i)\}$$

1: Let $TS = \{t_j^i \mid (t_j^i, x_j^i, y_j^i) \in T_i \ : \ T_i \in \mathcal{T}\}$ be all timestamps from all locations of all trajectories.
2: **for all** $T_i \in \mathcal{T}$ **do**
3: **for all** $ts \in TS$ with $t_1^i < ts < t_{n^i}^i$ **do**
4: **if** location having timestamp ts is not in T_i **then**
5: insert new location to T_i having the timestamp ts and coordinates interpolated from the two timewise-neighboring locations
6: **end if**
7: **end for**
8: **end for**

Definition 4 (Distance between trajectories)**.** Consider a set of synchronized trajectories $\mathcal{T} = \{T_1, \ldots, T_N\}$ where each trajectory is written as

$$T_i = \{(t_1^i, x_1^i, y_1^i), \ldots, (t_{n^i}^i, x_{n^i}^i, y_{n^i}^i)\} \ .$$

The *distance between trajectories* is defined as follows. If $T_i, T_j \in \mathcal{T}$ are p%-contemporary with $p > 0$, then

$$d(T_i, T_j) = \frac{1}{p} \sqrt{\sum_{t_\ell \in ot(T_i, T_j)} \frac{(x_\ell^i - x_\ell^j)^2 + (y_\ell^i - y_\ell^j)^2}{|ot(T_i, T_j)|^2}} \ .$$

If $T_i, T_j \in \mathcal{T}$ are 0%-contemporary but there is at least one subset of $\mathcal{T}$

$$\mathcal{T}^k(ij) = \{T_1^{ijk}, T_2^{ijk}, \ldots, T_{n^{ijk}}^{ijk}\} \subseteq \mathcal{T}$$

such that $T_1^{ijk} = T_i$, $T_{n^{ijk}}^{ijk} = T_j$ and T_ℓ^{ijk} and $T_{\ell+1}^{ijk}$ are p_ℓ%-contemporary with $p_\ell > 0$ for $\ell = 1$ to $n^{ijk} - 1$, then

$$d(T_i, T_j) = \min_{\mathcal{T}^k(ij)} \left(\sum_{\ell=1}^{n^{ijk}-1} d(T_\ell^{ijk}, T_{\ell+1}^{ijk}) \right)$$

Otherwise $d(T_i, T_j)$ is not defined.

The computation of the distance between every pair of trajectories is not exponential as it could seem from the definition. Polynomial-time computation of a distance graph containing the distances between all pairs of trajectories can be done as follows.

Definition 5 (Distance graph)**.** A *distance graph* is a weighted graph where

(i) Nodes represent trajectories,
(ii) two nodes T_i and T_j are adjacent if the corresponding trajectories are p%-contemporary for some $p > 0$, and
(iii) the weight of the edge (T_i, T_j) is the distance between the trajectories T_i and T_j.

Now, given the distance graph for $\mathcal{T} = \{T_1, \ldots, T_N\}$, the distance $d(T_i, T_j)$ for two trajectories is easily computed as the minimum cost path between the nodes T_i and T_j, if such path exists. The inability to compute the distance for all possible trajectories (the last case of Definition 4) naturally splits the distance graph into connected components. The connected component that has the majority of the trajectories should be kept, while the remaining components present outlier trajectories that are discarded in order to preserve privacy. Finally, given the connected component of the distance graph having the majority of the trajectories of $\mathcal{T}$, the distance $d(T_i, T_j)$ for *any two* trajectories on this connected component is easily computed as the minimum cost path between the nodes T_i and T_j. In terms of computational costs, the minimum cost path between every pair of nodes can be compute using the Floyd-Warshall algorithm with computational cost $O(N^3)$, *i.e.*, in polynomial time.

2.2 Clustering algorithm

Any constrained clustering algorithm can now be used with this distance in order to cluster trajectories, as outlined in Algorithm 2. We limit ourselves to clustering algorithms that try to minimize the sum of the intra-cluster distances or approximate the minimum. The constraints are that each cluster should be of size at least k and at most $2k-1$, where k is an input parameter.

The purpose of limiting the size of each cluster between k and $2k-1$ is twofold. First, the purpose is to provide privacy by performing anonymization on at least k trajectories. Second, the purpose is also to minimize information loss by capping the maximum cluster size at $2k-1$ trajectories in order to control the possible distortion: indeed, a cluster of size $2k$ can be split into two clusters of size k, whose separate anonymization fulfills the privacy requirement and leads to better data utility. Examples of clustering algorithms include the Greedy Clustering (described in Section 4 below) and Maximum Distance to Average Vector (MDAV) methods [8].

Algorithm 2 Cluster-based trajectory anonymization

Require: $\mathcal{T} = \{T_1, \ldots, T_N\}$ a set of trajectories such that $d(T_i, T_j)$ is defined for all $T_i, T_j \in \mathcal{T}$.

1: Use any clustering algorithm to cluster the trajectories of $\mathcal{T}$, while minimizing the sum of intra-cluster distances measured with the distance defined in Definition 4 and ensuring that minimum cluster size is at least k.
2: Let $C_1, C_2, \ldots, C_{n(\mathcal{T})}$ be the resulting clusters.
3: **for all** clusters C_i **do**
4: $C_i^\star = \text{SwapTriples}(C_i, R_i^t, R_i^s)$ // Algorithm 3
5: **end for**

2.3 Trajectory anonymization

Every trajectory anonymization algorithm must combine utility and privacy. However, utility and privacy are two largely antagonistic concepts. What is useful in a set of trajectories is application-dependent, so for each utility feature probably a different anonymization algorithm is needed. The utility features that are usually considered are: (i) length preservation, (ii) shape preservation, (iii) time preservation, and (iv) minimization of the number of discarded trajectories. We include another utility feature that is really meaningful in city scenarios: *location preservation*. This essentially means that the set of locations visited by the original trajectories is preserved: all locations visited by the original trajectories are also visited by the anonymized trajectories. Without location preservation, an adversary may be able to distinguish true locations and fake locations that are inserted by some anonymization methods. Hence, location preservation is not just good for utility but it is also good for privacy: the adversary cannot discard any fake locations, because the true original locations are preserved. To achieve this anonymization, we propose the SwapTriples method, presented in Algorithm 3.

SwapTriples takes a cluster C_i as input and it outputs a cluster $C_i^\star$ containing the transformed trajectories obtained from the original ones in cluster C_i. We denote the transformed trajectories, triples, and sets with a star. We refer to $\mathcal{T}$ as the *original trajectories* and to $\mathcal{T}^\star$ as the *anonymized trajectories*. Analogously, $T^\star \in \mathcal{T}^\star$ is called an *anonymized trajectory*. To obtain a transformed trajectory corresponding to a given original trajectory, each triple in the original trajectory can be swapped with a triple in another original trajectory provided that the timestamps of both triples differ no more than a time threshold R_i^t and the spatial coordinates of both triples are no more distant than a space threshold R_i^s.

Algorithm 3 SwapTriples(C_i, R_i^t, R_i^s)

Require: C_i a cluster of trajectories to be transformed, R_i^t a time threshold, and R_i^s a space threshold.

1: Mark all trajectories in C_i as "available".
2: Mark all triples in the trajectories of C_i as "unswapped".
3: **while** "available" trajectories are left in C_i **do**
4: Let T' be the longest "available" trajectory in C_i.
5: Let n' be the length of T'.
6: **for** $j = 1$ to n' **do**
7: **if** j-th triple (t'_j, x'_j, y'_j) of T' is "unswapped" **then**
8: Let S'_j be the set of "unswapped" triples in C_i of the form (t^s_j, x^s_j, y^s_j) with

$$|t^s_j - t'_j| \leq R_i^t$$

$$\sqrt{(x^s_j - x'_j)^2 + (y^s_j - y'_j)^2} \leq R_i^s \ .$$

9: **if** $S'_j = \emptyset$ **then**
10: Leave (t'_j, x'_j, y'_j) unaltered and mark it as "swapped".
11: **else**
12: Swap (t'_j, x'_j, y'_j) with a triple randomly chosen in S'_j and mark both swapped triples as "swapped".
13: **end if**
14: Mark T' as "unavailable".
15: **end if**
16: **end for**
17: **end while**
18: Return a cluster $C_i^\star$ containing the transformed trajectories.

3. PRIVACY AND UTILITY

We argue that k-anonymity applied to trajectories cannot always be realized: in particular, if the time and/or space thresholds used in Algorithm 3 are too small w.r.t. to the spread of original trajectories, often there will be less than k different trajectories with which a certain original trajectory can swap triples. Therefore we use the location re-identification probability as our privacy risk measure. We compare this privacy guarantee with the one of (k, δ)-anonymity [1] and show that when trajectories are defined over the same time span, we provide the same level of privacy.

Anonymizing trajectories needs to preserve utility as much as possible. In particular, we deal with several concerns regarding utility – the utility features that are usually considered by the users of the anonymized data and are discussed above in the previous section. We recall and present metrics for measuring the utility of anonymized trajectories.

3.1 Deficiency of k-anonymity for anonymizing trajectories

The notion of k-anonymity [15, 16] has been originally proposed for microdata, *i.e.*, transactional and relational databases. If the data satisfy the k-anonymity definition, an individual represented by a record in the microdata cannot be identified with probability higher than $1/k$ because his/her record cannot be distinguished within a group of at least k records. The k-anonymity notion can be extended to other types of data, including trajectories [1, 13, 14, 12]. Un-

fortunately, a direct extension of k-anonymity cannot protect the privacy of trajectories as shown in the following example.

Suppose that the adversary has trajectory T consisting of only one location – individual's home at 8am. Assume straightforward k-anonymity has been achieved, in such a way that there are at least k anonymized trajectories in $\mathcal{T}^\star$ having an anonymized version of T as a sub-trajectory, where a sub-trajectory is a selection of triples of a trajectory. This means that there will be k anonymized trajectories containing the single location of T. However, not all of these anonymized trajectories start at the single location of T. Since an individual's home at 8am is likely to be the first location of any individual's original trajectory, those anonymized trajectories that do not start at the single location of T (just pass through it) can be filtered out by an adversary and only the remaining trajectories are considered. In this way, using side knowledge the adversary identifies less than k anonymized trajectories compatible with the original trajectory T. Hence, straightforward k-anonymity does not imply actual trajectory k-anonymity.

We believe that our anonymization approach, which considers location re-identification, better fits the spatio-temporal data anonymization.

3.2 Re-identification probability

Because k-anonymity is insufficient in the trajectories setting, we measure the privacy risk as the *location re-identification probability*. It represents the average risk, measured as a probability, of re-identification of a location by an adversary in the anonymized trajectories $\mathcal{T}^\star$ obtained with SwapTriples from the original trajectories $\mathcal{T}$. Because the swap options to anonymize a location depend on each particular cluster, we consider the average location re-identification probability that cumulatively captures probabilities over all clusters.

Theorem 1. *Given a set $\mathcal{T} = \{T_1, \ldots, T_N\}$ of trajectories anonymized as $\mathcal{T}^\star = \{T_1^\star, \ldots, T_N^\star\}$ using Algorithm 2, the average location re-identification probability is $\frac{1}{S(\mathcal{T}, \mathcal{T}^\star)}$, where $S(\mathcal{T}, \mathcal{T}^\star)$ is the average number of swap options that each location in the original trajectories had in the swapping process (the number of swap options depends on the trajectories and also on the parameters R_i^t's and R_i^s's used to call Algorithm 3).*

We recall the (k, δ)-anonymity [1] definition in order to compare our privacy guarantee with it under the assumption that all trajectories are defined over the same time span.

Definition 6 ((k, δ)-anonymity [1])**.** A set C of trajectories defined over the timestamps TS having the form

$$T_i = \{(t, x_t^i, y_t^i) \,|\, t \in TS\}$$

satisfies (k, δ)-*anonymity* if $|C| \geq k$ and $\forall T_i, T_j \in C$ and $\forall t \in TS$ holds

$$\sqrt{(x_t^i - x_t^j)^2 + (y_t^i - y_t^j)^2} \leq \delta \ .$$

Our privacy guarantee can be the same as the one of (k, δ)-anonymity when considering trajectories defined over the same time span. We note however that our anonymization approach and privacy guarantee are also applicable to trajectories with partial or even no time overlap.

Theorem 2. *Let C be a set (cluster) of at least k trajectories defined over the same time span. There exist δ such that the trajectories anonymized by SwapTriples satisfy (k, δ)-anonymity.*

Proof. For any space threshold R^s and any time threshold R^t, the anonymized trajectories of the set $C^\star = SwapTriples(C, R^s, R^t)$ will collectively span the same time and some space. By assumption, all trajectories in C and therefore all trajectories in $C^\star$ are defined over the same set of timestamps TS. Let δ be the maximal Euclidean distance between any two points in two different trajectories in $C^\star$ sharing the same timestamp $t \in TS$. Then obviously, $C^\star$ satisfies (k, δ)-anonymity (of course, the constructed δ may not be minimal). □

3.3 Utility measures

Counting the number of removed trajectories as well as the number of removed locations, whether during pre-processing, clustering or cluster anonymization is easy.

Measuring the length of a trajectory is done in the obvious way, as the length of the poly-line of a trajectory projected on its spatial coordinates. We then compare the average lengths of trajectories in original clusters and in distorted clusters.

To capture the distortion of the trajectory shape, we use the space distortion metric [1, Sec.VI.B], which also allows to cumulatively consider the total space distortion of all anonymized trajectories from original ones.

Definition 7 (Space distortion [1])**.** The space distortion of an anonymized trajectory $T^\star$ with respect to its original trajectory T at time t when T has location (t, x, y) and $T^\star$ has possible location $(t, x^\star, y^\star)$, is

$$SD_t(T, T^\star) = \begin{cases} \Delta((x, y), (x^\star, y^\star)) & \text{if } (x^\star, y^\star) \text{ defined at } t \\ \Omega & \text{otherwise} \end{cases}$$

where Δ is a distance (*e.g.*, Euclidean), and Ω a constant that penalizes for removed points. The space distortion of an anonymized trajectory $T^\star$ from its original T is then

$$SD(T, T^\star) = \sum_{t \in TS} SD_t(T, T^\star) \ ,$$

where TS are all the timestamps where T is defined. In particular, if T is discarded during anonymization, $T^\star$ is empty, and so $SD(T, T^\star) = n\Omega$, where $n = |TS|$ is the number of locations of T. Finally, the space distortion of a set of trajectories $\mathcal{T}$ from its anonymized set $\mathcal{T}^\star$ is

$$TotalSD(\mathcal{T}, \mathcal{T}^\star) = \sum_{T \in \mathcal{T}} SD(T, T^\star) \ ,$$

where $T^\star \in \mathcal{T}^\star$ (which may be empty) corresponds to $T \in \mathcal{T}$.

4. EXPERIMENTAL RESULTS

We used 1,000 synthetic trajectories generated with the Brinkhoff's generator [4] visiting 45,505 locations in the German city of Oldenburg. Our results indicate that: (i) we discard significantly fewer trajectories than (k, δ)-anonymity, due to being able to consider also trajectories that do not have the same time span; (ii) our algorithm falls short of entirely preserving the lengths of trajectories, even though the lengths of anonymized trajectories are strongly correlated

to the lengths of original trajectories; (iii) when considering the same distortion achieved by (k, δ)-anonymity [1], our location re-identification probability is smaller; and (iv) our anonymization algorithm preserves all original locations, by design.

4.1 Synthetic data generation

In practice, testing anonymization algorithms with real data sets of trajectories is problematic, especially because data sets having a significant number of trajectories are hard to obtain. Hence, most anonymization algorithms are usually tested on synthetic data, which have the additional advantage of being easily transferable to other authors in view of experiment reproducibility.

We used Brinkhoff's generator [4] to obtain trajectories of moving objects and evaluate and compare our anonymization with other proposals. Synthetic data generated with Brinkhoff's generator have also been used for evaluation of approaches like [1, 13, 14, 18].

The generation parameters over the map of Oldenburg in Germany were: 6 moving object classes and 3 external object classes; 10 moving objects and 1 external object generated per time stamp; 100 timestamps; speed 250; and "probability" 1,000. This resulted in 1,000 trajectories containing 45,405 points. The maximum trajectory length was 100 points, the average length was 45.4 points, and the median length was 44 points.

4.2 Details of our approach

In our approach, a trajectory is removed only if the distance between it and some other trajectory cannot be computed. In fact, given the distance graph G (see Definition 5), our approach can only be used within one of the connected components of G. During the construction of the distance graph for the synthetic data we found 11 connected components, 10 of them of size 1. Therefore, we removed these 10 trajectories in order to obtain a new distance graph with just one connected component. In this way, we preserved 99% percent of all trajectories. The removed trajectories were in fact trajectories of length one, *i.e.*, just one location in each one.

We implemented the standard Greedy Clustering method and executed it for $k = 2$, 4, 6, 8, 10, and 15. Greedy Clustering first creates clusters of size k and disperses the up to $k-1$ unclustered trajectories to existing clusters while minimizing the intra-cluster distance. This algorithm incurs no additional discarding of trajectories.

For the anonymization of trajectories in the cluster using our SwapTriples method, we initially set for every i a time threshold $R_i^t = 100$ and a space threshold $R_i^s = 317$. It is important to remark that $R_i^t = 100$ is the entire time span of the synthetic data (100 timestamps at 100 consecutive time units), and $R_i^s = 317$ is five times the average distance between every pair of consecutive points inside the trajectories. For successive anonymizations aimed at comparing disclosure risk with (k, δ)-anonymity, we selected R_i^s and R_i^t in a way to obtain roughly the same total space distortion values as in (k, δ)-anonymity (see Table 1).

4.3 Implementing (k, δ)-anonymity for comparison with our method

We compared our approach with (k, δ)-anonymity [1]. Because the (k, δ)-anonymity only works over trajectories having the same time span, first a pre-processing step that partitions the trajectories is needed. By superimposing the begin and end times of the trajectories by reducing times modulo a parameter π, it is not always possible to find at least k trajectories having the same time span or it may happen that a trajectory disappears because the new reduced end time lies before the new reduced begin time.

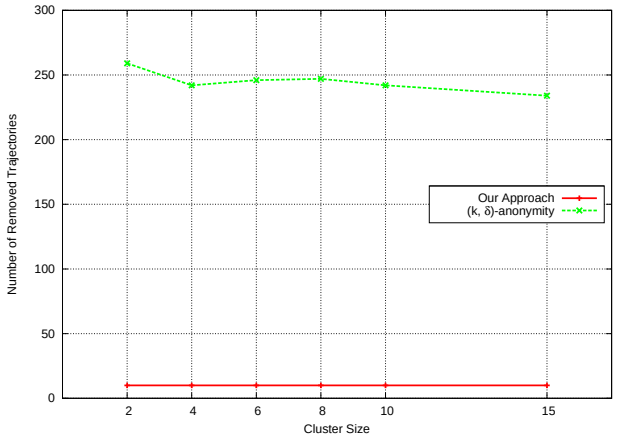

Figure 1: The number of removed trajectories in our approach and in (k, δ)-anonymity for cluster size k. Our approach discarded only 10 outlier trajectories while computing the distances among trajectories. (k, δ)-Anonymity discarded 227 trajectories in the pre-processing step and additional ones during clustering.

We have used $\pi = 3$ which kept the maximum (and so discarded the minimum) trajectories. From the 1,000 synthetic trajectories, 40 were discarded because the end time was less than the begin time and 187 were discarded because there were less than or equal to 4 trajectories having the same time span. In total, 227 (22.7%) trajectories were discarded just in the pre-processing step. The remaining 773 trajectories were in 32 sets having the same time span, each set containing a minimum of 15 trajectories and 24 on average.

We performed (k, δ)-anonymization for $k = 2$, 4, 6, 8, 10, and 15 and $\delta = 0$, 1000, 2000, 3000, 4000 and 5000. Because of the pre-processing step, using a higher k was impossible without causing a significant number of additional trajectories to be discarded.

4.4 Comparison, evaluation and discussion

Removed trajectories. Our approach discarded 10 outlier trajectories (1%) because the distance among them and the others could not be determined. On the other hand, (k, δ)-anonymity discarded 227 trajectories in the pre-processing step, because their time span could not match other trajectories, and discarded additional outlier trajectories during clustering, altogether discarding more than 24% of trajectories. These results are depicted in Figure 1, and it is clear that our approach outperforms (k, δ)-anonymity. This is because we are able to consider also trajectories that do not have the same time span, while (k, δ)-anonymity needed to discard 22.7% of trajectories in the pre-processing step.

Length of trajectories. Figure 2 shows that lengths of the anonymized trajectories are proportional to the length of

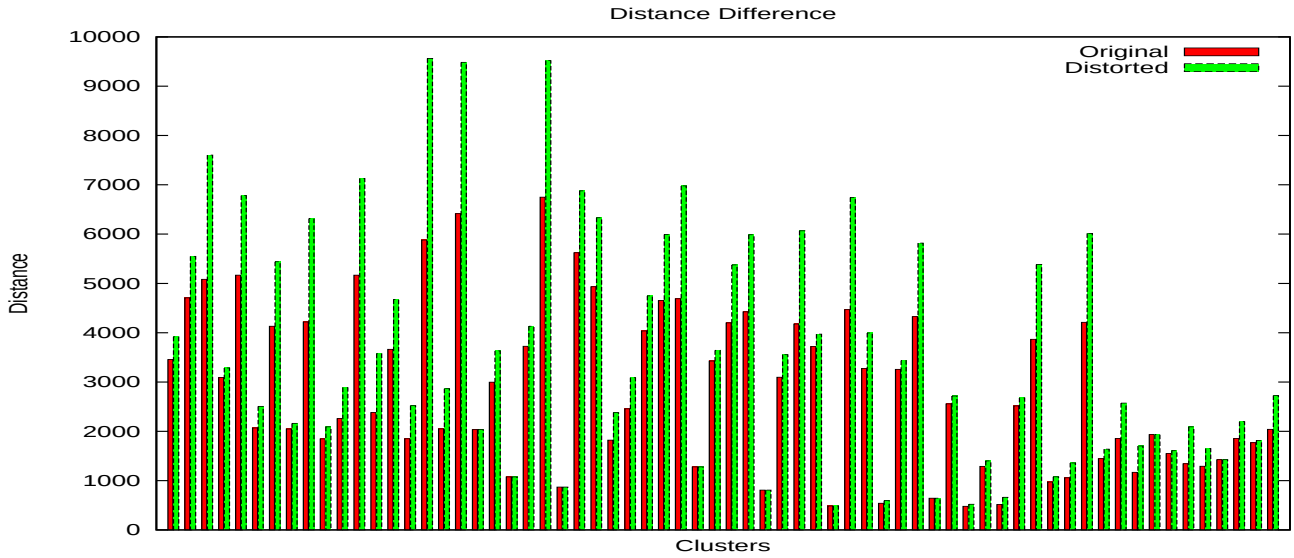

Figure 2: Average distance of the original trajectories and average distance of the distorted trajectories in each cluster when $k = 15$ and SwapTriples used $R_i^s = 317$ and $R_i^t = 100$. A similar situation happens with other k values, but there are too many clusters to show.

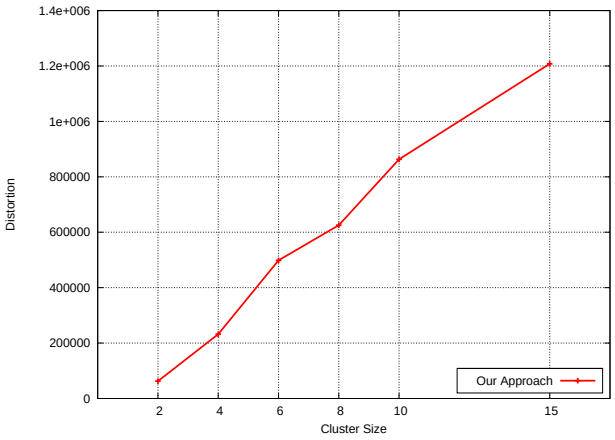

Figure 3: Total space distortion (TotalSD) of our approach with different cluster size k and using SwapTriples$(\cdot, R_i^s = 317, R_i^t = 100)$ for every cluster.

the original trajectories. This is acceptable from a statistical point of view, but we conclude that our approach does not preserve the lengths of trajectories.

Space distortion. Figure 3 depicts total space distortion of our approach with different cluster sizes k and using space threshold $R_i^s = 317$ and time threshold $R_i^t = 100$. In comparison to (k, δ)-anonymity, whose total space distortion is in Table 1, our approach achieves significantly lower distortion. Furthermore, the penalty constant Ω was set to 0 in the computation of the space distortion, partly because we already considered the number of removed trajectories separately. However, with any penalty $\Omega > 0$, the TotalSD would show even bigger gap between the two methods.

Re-identification probability. The two approaches we are comparing use different parameters, and so we fix the utility in order to compare privacy. Namely, for various k,

$\delta \setminus k$	2	4	6	8	10	15
0	48e6	93e6	120e6	143e6	165e6	199e6
1,000	19e6	60e6	86e6	109e6	131e6	165e6
2,000	4e6	32e6	56e6	78e6	99e6	133e6
3,000	.9e6	14e6	32e6	52e6	71e6	104e6
4,000	.2e6	5e6	16e6	32e6	48e6	79e6
5,000	.03e6	2e6	7e6	18e6	31e6	58e6

Table 1: Total space distortion (TotalSD) of (k, δ)-anonymity in millions. Compare with Figure 3 where our approach has TotalSD significantly smaller.

we consider the total space distortion of (k, δ)-anonymity as presented in Table 1. We subsequently choose time and space thresholds (R_i^t and R_i^s) for the particular cases in a way so that our approach achieves roughly the same total space distortion values.

Equipped with these privacy parameters, we present location re-identification probabilities in Figure 4. For (k, δ)-anonymity it is $\frac{1}{k}$, while for our approach it is mostly below this probability. Furthermore, the higher the k, the lower the location re-identification probability of our method compared to $\frac{1}{k}$. That is, while the utility is the same for both methods, the higher the aimed privacy, the more our method outperforms (k, δ)-anonymity in what regards privacy preservation.

5. CONCLUSIONS

First, we define a new distance between two trajectories that naturally combines time and space. We show that the use of this distance considerably reduces the number of locations or trajectories that must be removed from the data set.

Second, we propose a new method for anonymization of spatio-temporal movement data via microaggregation that

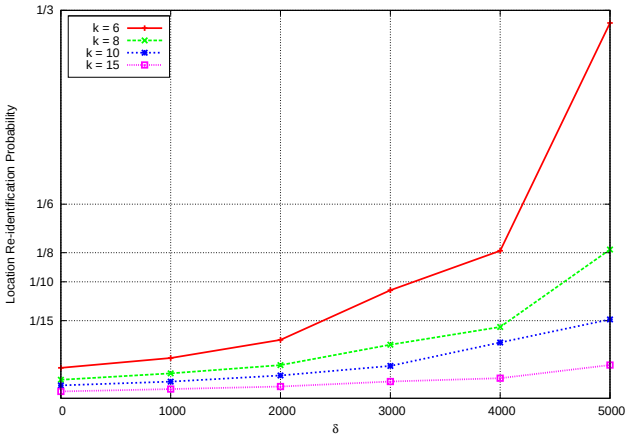

Figure 4: Location re-identification probability of our approach when considering total space distortion (TotalSD) values shown in Table 1 and cluster size $k = 6, 8, 10, 15$. Mostly our re-identification probabilities are below the $\frac{1}{k}$ re-identification probabilities offered by (k, δ)-anonymity. The trend is that the higher the aimed privacy – the higher the k, the lower the re-identification probability of our method vs (k, δ)-anonymity.

uses the proposed distance. The most obvious contributions of our method are: (i) it can deal with trajectories with partial or no time overlap; (ii) it preserves the visited locations after the anonymization process; (iii) it substantially reduces the proportion of discarded trajectories vs (k, δ)-anonymity (a similar approach, except that it only works over trajectories having a 100% time overlap). Moreover, we show that using the same levels of space distortion as (k, δ)-anonymity, we reach lower location re-identification probabilities, *i.e.*, for roughly the same levels of utility we achieve higher levels of privacy; and (iv) although length preservation is not exactly ensured, there is a strong correlation between the length of original trajectories and the length of the anonymized trajectories.

Acknowledgments

This work was partly supported by the Spanish Government through projects TSI2007-65406-C03-01 "E-AEGIS", CONSOLIDER INGENIO 2010 CSD2007-00004 "ARES" and TSI-020100-2009-720 "everification", and by the Government of Catalonia under grant 2009 SGR 01135.

The authors are with the UNESCO Chair in Data Privacy, but they are solely responsible for the views expressed in this paper, which do not necessarily reflect the position of UNESCO nor commit that organization.

6. REFERENCES

[1] O. Abul, F. Bonchi, and M. Nanni. Never walk alone: uncertainty for anonymity in moving objects databases. In *ICDE*, pages 376–385, 2008.

[2] C. C. Aggarwal and P. S. Yu. A condensation approach to privacy preserving data mining. In *EDBT*, pages 183–199, 2004.

[3] C. C. Aggarwal and P. S. Yu. On anonymization of string data. In *SIAM SDM*, pages 419–424, 2007.

[4] T. Brinkhoff. Generating traffic data. *IEEE Data Eng. Bull.*, 26(2):19–25, 2003.

[5] L. Chen, M. T. Ozsu, and V. Oria. Robust and fast similarity search for moving object trajectories. In *SIGMOD*, pages 491–502, 2005.

[6] J. Domingo-Ferrer and U. González-Nicolás. Hybrid microdata using microaggregation. *Inform. Sciences*, 180(15):2834–2844, 2010.

[7] J. Domingo-Ferrer and J. M. Mateo-Sanz. Practical data-oriented microaggregation for statistical disclosure control. *IEEE Trans. Knowl. Data Eng.*, 14(1):189–201, 2002.

[8] J. Domingo-Ferrer and V. Torra. Ordinal, continuous and heterogenerous k-anonymity through microaggregation. *Data Min Knowl Disc*, 11(2):195–212, 2005.

[9] B. Fung, K. Wang, R. Chen, and P. S. Yu. Privacy-preserving data publishing: a survey on recent developments. *ACM Comput. Surv.*, 42(4):to appear, 2010.

[10] B. Hoh, M. Gruteser, H. Xiong, and A. Alrabady. Preserving privacy in GPS traces via uncertainty-aware path cloaking. In *CCS*, pages 161–171, 2007.

[11] E. Kaplan, T. B. Pedersen, E. Savas, and Y. Saygin. Discovering private trajectories using background information. *Data Knowl. Eng.*, 69(7):723–736, 2010.

[12] A. Monreale, G. Andrienko, N. Andrienko, F. Giannotti, D. Pedreschi, S. Rinzivillo, and S. Wrobel. Movement data anonymity through generalization. *Trans. Data Privacy*, 3(2):91–121, 2010.

[13] M. E. Nergiz, M. Atzori, and Y. Saygin. Towards trajectory anonymization: a generalization-based approach. In *SPRINGL*, pages 52–61, 2008.

[14] M. E. Nergiz, M. Atzori, Y. Saygin, and B. Guc. Towards trajectory anonymization: a generalization-based approach. *Trans. Data Privacy*, 2(1):47–75, 2009.

[15] P. Samarati and L. Sweeney. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical Report SRI-CSL-98-04, SRI Computer Science Laboratory, 1998.

[16] L. Sweeney. k-anonymity: a model for protecting privacy. *Int. J. Uncertain. Fuzz.*, 10(5):557–570, 2002.

[17] M. Terrovitis and N. Mamoulis. Privacy preservation in the publication of trajectories. In *MDM*, pages 65–72, 2008.

[18] R. Yarovoy, F. Bonchi, L. V. S. Lakshmanan, and W. H. Wang. Anonymizing moving objects: how to hide a mob in a crowd? In *EDBT*, pages 72–83, 2009.