Privacy-Preserving Distributed Movement Data Aggregation

Anna Monreale, Wendy Hui Wang, Francesca Pratesi, Salvatore Rinzivillo, Dino Pedreschi, Gennady Andrienko and Natalia Andrienko

Abstract We propose a novel approach to privacy-preserving analytical processing within a distributed setting, and tackle the problem of obtaining aggregated information about vehicle traffic in a city from movement data collected by individual vehicles and shipped to a central server. Movement data are sensitive because people's whereabouts have the potential to reveal intimate personal traits, such as religious or sexual preferences, and may allow re-identification of individuals in a database. We provide a privacy-preserving framework for movement data aggregation based on trajectory generalization in a distributed environment. The proposed solution, based on the differential privacy model and on sketching techniques for efficient data compression, provides a formal data protection safeguard. Using real-life data, we demonstrate the effectiveness of our approach also in terms of data utility preserved by the data transformation.

A. Monreale (✉) · F. Pratesi · D. Pedreschi
University of Pisa, Pisa, Italy
e-mail: annam@di.unipi.it

F. Pratesi
e-mail: prafra@yahoo.it

D. Pedreschi
e-mail: pedre@di.unipi.it

W. H. Wang
Stevens Institute of Technology, Hoboken, NJ, USA
e-mail: Hui.Wang@stevens.edu

S. Rinzivillo
ISTI-CNR, Pisa, Italy
e-mail: salvatore.rinzivillo@isti.cnr.it

G. Andrienko · N. Andrienko
Fraunhofer IAIS, Sankt Augustin, Germany
e-mail: gennady.andrienko@iais.fraunhofer.de

N. Andrienko
e-mail: natalia.andrienko@iais.fraunhofer.de

D. Vandenbroucke et al. (eds.), *Geographic Information Science at the Heart of Europe*, Lecture Notes in Geoinformation and Cartography,
DOI: 10.1007/978-3-319-00615-4_13,

1 Introduction

The widespread availability of low cost GPS devices enables collecting data about movements of people and objects at a large scale. Understanding of the human mobility behavior in a city is important for improving the use of city space and accessibility of various places and utilities, managing the traffic network, and reducing traffic jams. Generalization and aggregation of individual movement data can provide an overall description of traffic flows in a given time interval and their variation over time. Chapter Andrienko and Andrienko (2011) proposes a method for generalization and aggregation of movement data that requires having all individual data in a central station. This centralized setting entails two important problems: (a) the amount of information to be collected and processed may exceed the capacity of the storage and computational resources, and (b) the raw data describe the mobility behavior of the individuals with great detail that could enable the inference of very sensitive information related to the personal private sphere.

In order to solve these problems, we propose a privacy-preserving distributed computation framework for the aggregation of movement data. We assume that on-board location devices in vehicles continuously trace the positions of the vehicles and can periodically send derived information about their movements to a central station, which stores it. The vehicles provide a statistical sample of the whole population, so that the information can be used to compute a summary of the traffic conditions on the whole territory. To protect individual privacy, we propose a data transformation method based on the well-known differential privacy model. To reduce the amount of information that each vehicle transmits to the central station, we propose to apply the sketch techniques to the differentially private data to obtain a compressed representation. The central station, that we call *coordinator*, is able to reconstruct the movement data represented by the sketched data that, although transformed for guaranteeing privacy, preserve some important properties of the original data that make them useful for mobility analysis.

The remainder of the chapter is organized as follows. Section 2 introduces background information and definitions. Section 3 describes the system architecture and states the problem. Section 4 presents our privacy-preserving framework. In Sect. 5, we discuss the privacy analysis. Experimental results from applying our method to real-world data are presented and discussed in Sect. 6. Section 7 discusses the related work and Sect. 8 concludes the chapter.

2 Preliminaries

2.1 Movement Data Representation

Definition 1 (*Trajectory*) A Trajectory or spatio-temporal sequence is a sequence of triplets $T = <l_1, t_1>, \ldots, <l_n, t_n>$, where t_i $(i = 1 \ldots n)$ denotes a timestamp such that $\forall_{1 \leq i < n}\ t_i < t_{i+1}$ and $l_i = (x_i, y_i)$ are points in $\mathbf{R^2}$.

Intuitively, each pair $\langle l_i, t_i \rangle$ indicates that the object is in the position $l_i = \langle x_i, y_i \rangle$ at time t_i.

We assume that the territory is subdivided in cells $C = \{c_1, c_2, \ldots, c_p\}$ which compose a partition of the territory. During a travel a user goes from a cell to another cell. We use g to denote the function that applies the spatial generalization to a trajectory. Given a trajectory T this function generates the generalized trajectory $g(T)$, i.e. a sequence of *moves* with temporal annotations, where a *move* is an pair (l_{c_i}, l_{c_j}), which indicates that the moving object moves from the cell c_i to the *adjacent* cell c_j. Note that, l_{c_i} denotes the pair of spatial coordinates representing the centroid of the cell c_i; in other words $l_{c_i} = \langle x_{c_i}, y_{c_i} \rangle$. The *temporal annotated move* is the quadruple $(l_{c_i}, l_{c_j}, t_{c_i}, t_{c_j})$ where l_{c_i} is the location of the origin, l_{c_j} is the location of the destination and the t_{c_i}, t_{c_j} are the time information associate to l_{c_i} and l_{c_j}. As a consequence, we define a generalized trajectory as follows.

Definition 2 (*Generalized Trajectory*) Let $T = \langle l_1, t_1 \rangle, \ldots, \langle l_n, t_n \rangle$ be a trajectory. Let $C = \{c_1, c_2, \ldots, c_p\}$ be the set of the cells that compose the territory partition. A generalized version of T is a sequence of temporal annotated moves

$$T_g = (l_{c_1}, l_{c_2}, t_{c_1}, t_{c_2})(l_{c_2}, l_{c_3}, t_{c_2}, t_{c_3}) \ldots (l_{c_{m-1}}, l_{c_m}, t_{c_{m-1}}, t_{c_m})$$

where $m \leq n$.

Now, we show how a generalized trajectory can be represented by a frequency distribution vector. First, we define the function *Move Frequency MF* that given a generalized trajectory T_g, a move (l_{c_i}, l_{c_j}) and a time interval computes how many times the move appears in T_g by considering the temporal constraint. More formally, we define the *Move Frequency* function as follows.

Definition 3 (*Move Frequency*) Let T_g be a generalized trajectory and let (l_{c_i}, l_{c_j}) be a move. Let τ be a temporal interval. The *Move Frequency* function is defined as:

$$MF(T_g, (l_{c_i}, l_{c_j}), \tau) = |\{(l_{c_i}, l_{c_j}, t_i, t_j) \in T_g | t_i \in \tau \wedge t_j \in \tau\}|.$$

This function can be easily extended for taking into consideration a set of generalized trajectories $\mathscr{T}^{\mathscr{G}}$. In this case, the information computed by the function represents the total number of movements from the cell c_i to the cell c_j in a time interval in the set of trajectories.

Definition 4 (*Global Move Frequency*) Let $\mathscr{T}^{\mathscr{G}}$ be a set of generalized trajectories and let (c_i, c_j) be a move. Let τ be a time interval. The *Global Move Frequency* function is defined as:

$$GMF(\mathscr{T}^{\mathscr{G}}, (c_i, c_j), \tau) = \sum_{\forall T_g \in \mathscr{T}^{\mathscr{G}}} MF(T_g, (c_i, c_j), \tau).$$

The number of movements between two cells computed by either the function *MF* or *GMF* describes the amount of traffic flow between the two cells in a specific time interval. This information can be represented by a frequency distribution vector.

Definition 5 (*Vector of Moves*) Let $C = \{c_1, c_2, \ldots, c_p\}$ be the set of the cells that compose the territory partition. The *vector of moves* M with $s = |\{(c_i, c_j) | c_i \textit{ is adjacent to } c_j\}|$ positions is the vector containing all possible moves. The element $M[i] = (l_{c_i}, l_{c_j})$ is the *move* from the cell c_i to the adjacent cell c_j.

Definition 6 (*Frequency Vector*) Let $C = \{c_1, c_2, \ldots, c_p\}$ be the of the cells that compose the territory partition and let M be the vector of moves. Given a set of generalized trajectories in a time interval τ $\mathscr{T}^{\mathscr{G}}$. The corresponding *frequency vector* is the vector f with size $s = |\{(c_i, c_j) | c_i \textit{ is adjacent to } c_j\}|$ where each $f[i] = GMF(\mathscr{T}^{\mathscr{G}}, M[i], \tau)$.

The definition of *frequency vector of a trajectory set* is straightforward; it requires to compute the function *GMF* instead of *MF*.

Note that the above definitions are based on the assumption that consecutive locations can be contained in the same cell or in adjacent cells. In some cases (for example, because of GPS problems) this fact would not be true. In order to avoid illegal moves (i.e., moves that are not present in the Frequency Vector) a reasonable solution is to reconstruct the missing part of the trajectories, e.g. by interpolation.

2.2 Differential Privacy

Differential privacy implies that adding or deleting a single record does not significantly affect the result of any analysis. Intuitively, differential privacy can be understood as follows. Let a database D include a private data record d_i about an individual i. By querying the database, it is possible to obtain certain information I(D) about all data and information I(D $-$ d_i) about the data without the record d_i. The difference between I(D) and I(D $-$ d_i) may enable inferring some private information about the individual i. Hence, it must be guaranteed that I(D) and I(D $-$ d_i) do not significantly differ for any individual i.

The formal definition (Dwork et al. 2006) is the following. Here the parameter, ε, specifies the level of privacy guaranteed.

Definition 7 (*ε-differential privacy*) A privacy mechanism A gives ε-differential privacy if for any dataset D_1 and D_2 differing on at most one record, and for any possible output D' of A we have

$$Pr[A(D_1) = D'] \leq e^{\varepsilon} \times Pr[A(D_2) = D']$$

where the probability is taken over the randomness of A.

Two principal techniques for achieving differential privacy have appeared in the literature, one for real-valued outputs (Dwork et al. 2006) and the other for outputs of arbitrary types (McSherry and Talwar 2007). A fundamental concept of both techniques is the global sensitivity of a function mapping underlying datasets to (vectors of) reals.

Definition 8 (*Global Sensitivity*) For any function $f : D \rightarrow R^d$, the sensitivity of f is

$$\Delta f = max_{D_1,D_2} ||f(D_1) - f(D_2)||_1$$

for all D_1, D_2 differing in at most one record.

For the analysis whose outputs are real, a standard mechanism to achieve differential privacy is to add Laplace noise to the true output of a function. Dwork et al. (2006) propose the Laplace mechanism which takes as inputs a dataset D, a function f, and the privacy parameter ε. The magnitude of the noise added conforms to a Laplace distribution with the probability density function $p(x|\lambda) = \frac{1}{2\lambda} e^{-|x|/\lambda}$, where λ is determined by both the global sensitivity of f and the desired privacy level ε.

Theorem 1 (Dwork et al. 2006) *For any function $f : D \rightarrow R^d$ over an arbitrary domain D, the mechanism A $A(D) = f(D) + Laplace(\Delta f/\varepsilon)$ gives ε-differential privacy.*

A relaxed version of differential privacy discussed in Michael and Sebastian (2012) allows claiming the same privacy level as Definition 7 in the case there is a small amount of privacy loss due to a variation in the output distribution for the privacy mechanism A is as follows.

Definition 9 [(ε, δ)-*differential privacy*] A privacy mechanism A gives (ε, δ)-differential privacy if for any dataset D_1 and D_2 differing on at most one record, and for any possible output D' of A we have

$$Pr[A(D_1) = D'] \leq e^{\varepsilon} \times Pr[A(D_2) = D'] + \delta$$

where the probability is taken over the randomness of A.

Note that, if $\delta = 0$, $(\varepsilon; 0)$-differential privacy is ε-differential privacy. In the remaining of this chapter we will refer to this last version of differential privacy.

3 Problem Definition

3.1 System Architecture

We consider a system architecture as that one in described in Cormode and Garofalakis (2008). In particular, we assume a distributed-computing environment comprising a collection of k (trusted) remote sites (nodes) and a designated coordinator site. Streams of data updates arrive continuously at remote sites, while the coordinator site is responsible for generating approximate answers to periodic user queries posed over the unions of remotely-observed streams across all sites. Each remote site exchanges messages only with the coordinator, providing it with state information on its (locally observed) streams. There is no communication between remote sites.

In our scenario, the coordinator is responsible for computing the aggregation of movement data on a territory by combining the information received by each node. In order to obtain the aggregation of the movement data in the centralized setting we need to generalize all the trajectories by using the cells of a partition of the territory. In our distributed setting we assume that the partition of the territory, i.e., the set of cells $C = \{c_1, \ldots, c_p\}$ useful for the generalization, is known by both all the nodes and the coordinator. Each node, that represents a vehicle that moves in this territory, in a given time interval observes a sequence of spatio-temporal points (trajectory), generalizes it and contributes to the computation of the global vector.

Formally, each remote site $j \in \{1, \ldots, k\}$ observes local update streams that incrementally render a collection of (up to) s distinct frequency distribution vectors (equivalently, multi-sets) $f_{1,j}, \ldots, f_{s,j}$ over data elements from corresponding integer domains $[U_i] = \{0, \ldots, U_{i_1}\}$, for $i = 1, \ldots, s$; that is, $f_{i,j}[v]$ denotes the frequency of element $v \in [U_i]$ observed locally at remote site j.

The coordinator for each $i \in \{1, \ldots, s\}$ computes the *global frequency distribution vector* $f_i = \sum_{j=1}^{k} f_{i,j}$.

3.2 Privacy Model

In our setting we assume that each node in our system is secure; in other words we do not consider attacks at node level. Instead, we take into consideration possible attacks from any intruder between the node and the coordinator (i.e., attacks during the communications), and any intruder at coordinator site, so our privacy preserving technique has to guarantee privacy even against a malicious behavior of

the coordinator. We consider sensitive information as any information from which the typical mobility behavior of a user may be inferred. This information is considered sensitive for two main reasons: (1) typical movements can be used to identify the drivers who drive specific vehicles even when a simple de-identification of the individual in the system is applied; and (2) the places visited by a driver could identify specific sensitive areas such as clinics, hospitals, the user's home.

Therefore, we need to find effective privacy mechanisms on the real count associate to each move, in order to generate uncertainty. As a consequence, the goal of our framework is to compute a distributed aggregation of movement data for a comprehensive exploration of them while preserving privacy.

Definition 10 (*Problem Definition*)

Given a set of cells $C = \{c_1, \ldots, c_p\}$ and a set $V = \{V_1, \ldots, V_k\}$ of vehicles, the *privacy-preserving distributed movement data aggregation problem* (DMAP) consists in computing, in a specific time interval τ the $f^{\tau}_{DMAP}(V) = [f_1, f_2, \ldots, f_s]$, where each $f_i = GMF(\mathscr{T}^{\mathscr{G}}, M[i], \tau)$ and $s = |\{(c_i, c_j) | c_i \textit{ is adjacent to } c_j\}|$, while preserving privacy. Here, $\mathscr{T}^{\mathscr{G}}$ is the set of generalized trajectories related to the k vehicles V in the time interval τ and M is the vector of moves defined on the set of cells C.

4 Our Solution

Clearly, in order to guarantee the privacy within this framework we may apply many privacy-preserving techniques depending on the privacy attack model and the background knowledge of the adversary that we want to consider in this scenario. In this chapter, we provide a solution based on the differential privacy, that is a very strong privacy model independent on the the background knowledge of an adversary. In this section, we describe the details of our solution, including the computation of each node and the coordinator in the system.

The pseudo code of our algorithm is shown in Algorithm 1. Each node represents a vehicle that moves in a specific territory and this vehicle in a given time interval observes sequences of spatio-temporal points (trajectories) and computes the corresponding frequency vector to be sent to the coordinator. The computation of the frequency vector requires four steps described in Algorithm 1: (a) trajectory generalization; (b) frequency vector construction; (c) frequency vector transformation for privacy guarantees and (d) vector sketching for compressing the information to be transmitted.

Algorithm 1: NodeComputation(ε, τ, M, T^G, w, d)

Input: A privacy budget ε, a time interval τ, the vector of the moves M, a set of trajectories T^G, the dimension of the sketch w and d.

Output: The sketch vector representing the privacy-preserving frequency vector $sk(\tilde{f}^{V_j})$.

foreach *observed trajectory* T **do**
  // Trajectory Generalization (Sec. 4.1)
  $T_g = TrajectoryGeneralization(M, T)$;
  // Update of the Frequency Vector f^{V_j} **(Sec. 4.2)**
  foreach *move* $(l_{c_i}, l_{c_j}) \in T_g$ **do**
    $n = MF(T_g, (l_{c_i}, l_{c_j}), \tau)$;
    $f^{V_j}[(l_{c_i}, l_{c_j})] += n$;
// Privacy Transformation (Sec. 4.3)
foreach *vector element* $f^{V_j}[i]$ **do**
  $noise = Laplace(0, \frac{1}{\varepsilon})$;
  if $noise > f^{V_j}[i]$ **then**
    $noise = f^{V_j}[i]$;
  if $noise < -f^{V_j}[i]$ **then**
    $noise = -f^{V_j}[i]$;
  $\tilde{f}^{V_j}[i] = f^{V_j}[i] + noise$;
// Generation of Sketch Vector (Sec. 4.4)
$sk(\tilde{f}^{V_j}) = CountMin(\tilde{f}^{V_j})$;
return $sk(\tilde{f}^{V_j}, w, d)$

4.1 Trajectory Generalization

Given a specific division of the territory, a trajectory is generalized in the following way. We apply place-based division of the trajectory into segments. The area c_1 containing its first point l_1 is found. Then, the second and following points of the trajectory are checked for being inside c_1 until finding a point l_i not contained in c_1. For this point l_i, the containing area c_2 is found.

The trajectory segment from the first point to the i-th point is represented by the vector (c_1, c_2). Then, the procedure is repeated: the points starting from l_{i+1} are checked for containment in c_2 until finding a point l_k outside c_2, the area c_3 containing l_k is found, and so forth up to the last point of the trajectory.

In the result, the trajectory is represented by the sequence of moves $(c_1, c_2, t_1, t_2)(c_2, c_3, t_2, t_3) \ldots (c_{m-1}, c_m, t_{m-1}, t_m)$. Here, in a specific quadruple t_i is the time moment of the last position in c_i and t_j is the time moment of the last position in c_j. There may be also a case when all points of a trajectory are contained in one and the same area c_1. Then, the whole trajectory is represented by the sequence $\{c_1\}$. Since, globally we want to compute aggregation of moves we discard this kind of trajectories. Moreover, as most of the methods for analysis of trajectories are suited to work with positions specified as points, the areas $\{c_1, c_2, \ldots, c_m\}$ are replaced, for practical purposes, by the sequence $l_{c_1}, l_{c_2}, \ldots, l_{c_m}$ consisting of the centroids of the areas $\{c_1, c_2, \ldots, c_m\}$.

4.2 Frequency Vector Construction

After the generalization of a trajectory, the node computes the *Move Frequency* function for each move (l_{c_i}, l_{c_j}) in that trajectory and updates its frequency vector f^{V_j} associated to the current time interval τ. Intuitively, the vehicle populates the frequency vector f^{V_j} according the generalized trajectory observed. So, at the end of the time interval τ the element $f^{V_j}[i]$ contains the number of times that the vehicle V_j moved from m to n in that time interval, if $M[i] = (m, n)$.

4.3 Vector Transformation for Achieving Privacy

As we stated in Sect. 3.2, if a node sends the frequency vector without any data transformation any intruder may infer the typical movements of the vehicle represented by the node. As an example, he could learn his most frequent move; this information can be considered very sensitive because the cells of this move usually correspond to user's home and his work place. Clearly, the generalization step can help the privacy user but it depends on the density of the area; specifically, if the area is not so dense it could identify few places and in that case the privacy is at risk. *How can we hide the event that the user moved from a location a to a location b in the time interval* τ? We propose a solution based on a very strong privacy model called ε-differential privacy (Sect. 2.2). As explained above the key point of this model is the definition of the sensitivity. Given a move (a, b) its sensitivity is straightforward: releasing its frequency have sensitivity 1 as adding or removing a single flow can affect its frequency by at most 1. Thus adding noise according to $Lap(\frac{1}{\varepsilon})$ to the frequency of each of the moves in the frequency vector satisfies ε-differential privacy. As a consequence, at the end of the time interval τ, before sending the vector to the coordinator, for each position of the vector (i.e., for each move) has to generate the noise by the Laplace distribution with zero mean and scale $\frac{1}{\varepsilon}$ and then has to add it to the value in that position of the vector. At the end of this step the node transforms f^{V_j} into $\tilde{f}^{V_j}$.

Differential privacy must be applied with caution because in some context it could lead to the destruction of the data utility because of the added noise that, although with small probability, can reach values of arbitrary magnitude. Moreover, adding noise drawn from the Laplace distribution could generate negative values for the flow in a move and negative flows does not make sense. To prevent this two problems we decided to draw the noise from a cutting version of the Laplace distribution. In particular, for each value x of the vector f^{V_j} we draw the noise from $Lap(\frac{1}{\varepsilon})$ bounding the noise value to the interval $[-x, x]$. In other words, if we have the original flow $f^{V_j}[i] = x$ in the perturbed version we obtain a flow value in the interval $[0, 2x]$. The use of a truncated version of the Laplace distribution can lead to privacy leaks and in Sect. 5 we show that our privacy mechanism satisfies (ε, δ)-differential privacy, where δ represents this privacy loss.

4.4 Vector Sketching for Compact Communications

In a system like ours an important issue to be considered is the amount of data to be communicated. In fact, real life systems usually involve 1,000 vehicles (nodes) that are located in any place of the territory. Each vehicle has to send to the coordinator the information contained in its frequency vector that has a size depending on the number of cells that represent the partition of the territory. The number of cells in a territory can be very huge and this can make each frequency vector too big. As an example, in the dataset of real-life trajectories used in our experiments we have about 4,200 vehicles and a frequency vector with about 15,900 positions. Therefore, the system has to be able to handle both a very large number of nodes and a huge amount of the global information to be communicated. These considerations make the reduction of the information communicated necessary. We propose the application of a sketching method (Cormode et al. 2012a) that allows us to apply a good compression of the information to be communicated. In particular, we propose the application of *Count-Min* sketch algorithm (Cormode and Muthukrishnan 2005). In general, this algorithm maps the frequency vector onto a more compressed vector. In particular, the sketch consists of an array C of $d \times w$ counters and for each of d rows a pairwise independent hash functions h_j, that maps items onto $[w]$. Each item is mapped onto d entries in the array, by adding to the previous value the new item. Given a sketch representation of a vector we can estimate the original value of each component of the vector by the following function $f[i] = min_{1 \leq j \leq d} C[j, h_j(i)]$. The estimation of each component j is affected by an error, but it is showed that the overestimate is less than n/w, where n is the number of components. So, setting $d = \log \frac{1}{\gamma}$ and $w = O(\frac{1}{\alpha})$ ensures that the estimation of $f[i]$ has error at most αn with probability at least $1 - \gamma$. Here, α indicates the accuracy (i.e. the approximation error), and γ represents the probability of exceeding the accuracy bounds.

4.5 Coordinator Computation

The computation of the coordinator is composed of two main phases: (1) computation of the set of moves and (2) computation of the aggregation of global movements.

Move Vector Computation. The coordinator in an initial setting phase has to send to the nodes the *vector of moves* (Definition 5). The computation of this vector depends on the set of cells that represent the partition of the territory. This partition can be a simple grid or a more sophisticated territory subdivision such as the Voronoi tessellation. The sharing of vector of moves is a requirement of the whole process because each node has to use the same data structure for allowing the coordinator the correct computation of the global flows.

Global Flow Computation. The coordinator has to compute the global vector that corresponds to the global aggregation of movement data in a given time interval τ by composing all the local frequency vectors. It receives the sketch vector $sk(\tilde{f}^{V_j})$ from each node; then it reconstructs each frequency vector from the sketch vector, by using the estimation described in Sect. 4.4. Finally, the coordinator computes the global frequency vector by summing the estimate vectors component by component. Clearly the estimate global vector is an approximated version of the global vector obtained by summing the local frequency vectors after the only privacy transformation.

5 Privacy Analysis

As pointed out in Kifer and Machanavajjhala (2011) differential privacy must be applied with caution. The privacy protection provided by differential privacy relates to the data generating mechanism and deterministic aggregate level background knowledge. As in our problem the trajectories in the raw database are independent of each other, and no deterministic statistics of the raw database will ever be released, we are ready to show that Algorithm 1 satisfies (ε, δ)-differential privacy.

Let F and F' be the frequency distribution before and after adding Laplace noise. We observe that bounding the Laplace noise will lead to some privacy leakage on some values. For instance, from the noisy frequency values that are large, the attacker can infer that these values should not be transformed from small ones. To analyze the privacy leakage of our bound-noise approach, we first explain the concept of *statistical distance*. Statistical distance is defined in Michael and Sebastian (2012). Formally, given two distributions X and Y, the *statistical distance* between X and Y over a set U is defined as

$$d(X, Y) = max_{S \in U}(Pr[X \in S] - Pr[Y \in S]).$$

Michael and Sebastian (2012) also shows the relationship between (ε, δ)-differential privacy and the statistical distance.

Lemma1 (Michael and Sebastian 2012) *Given two probabilistic functions F and G with the same input domain, where* F *is* (ε, δ_1)*-differentially private. If for all possible inputs x we have that the statistical distance on the output distributions of F and G is*:

$$d(F(x), G(x)) \leq \delta_2,$$

then G *is* $(\varepsilon, \delta_1 + (e^{\varepsilon} + 1)\delta_2)$*-differentially private*.

Let F and F' be the frequency distribution before and after adding Laplace noise. We can show that the statistical distance between F and F' can be bounded as follows:

Lemma 2 (Michael and Sebastian 2012) *Given an* (ε, δ)-*differentially private function* F *with* $F(x) = f(x) + R$ *for a deterministic function* f *and a random variable* R. *Then for all* x, *the statistical distance between* F *and its throughput-respecting variant* F' *with the bound* b *on* R *is at most*

$$d(F(x) - F'(x)) \leq Pr[|R| > b].$$

Michael and Sebastian (2012) has the following lemma to bound the probability $Pr[|R| > b]$.

Lemma 3 (Michael and Sebastian 2012) *Given a function* F *with* $F(x) = f(x) + Lap(\frac{\Delta f}{\varepsilon})$ *for a deterministic function* f, *the probability that the Laplacian noise* $Lap(\frac{\Delta f}{\varepsilon})$ *applied to* f *is larger than* b *is bounded by*:

$$Pr(Lap(\frac{\Delta f}{\varepsilon}) > b) \leq \frac{2(\Delta f)^2}{b^2\varepsilon^2}.$$

We stress that in our approach, the bound b of each frequency value x is not fixed. Indeed, $b = x$. Therefore, each frequency value x has different amounts of privacy leakage. Our approach thus achieves different degree of (ε, δ)-differentially privacy guarantee on each frequency value x. Theorem 2 shows more details.

Theorem 2 *Given the total privacy budget* ε, *for each frequency value* x, *Algorithm* 1 *ensures* $(\varepsilon, (e^{\varepsilon} + 1)\frac{2}{x^2\varepsilon^2})$-d*ifferential privacy*.

Proof Algorithm 1 consists of four steps, namely *TrajectoryGeneralization*, *FrequencyVectorUpdate*, *PrivacyTransformation*, and *SketchVectorGeneration*. The steps of *TrajectoryGeneralization* and *FrequencyVectorUpdate* mainly prepare the frequency vectors for privacy transformation. Hence we focus on the privacy guarantee of *PrivacyTransformation* and *SketchVectorGeneration* steps. For each frequency value x, the *PrivacyTransformation* step can achieve $(\varepsilon, (e^{\varepsilon} + 1)\frac{2(\Delta f)^2}{x^2\varepsilon^2}$-differentially privacy. This can be easily proven by Lemma 1 and Lemma 3. Note that the the frequency vectors with Laplace noise (without truncation) satisfies $(\varepsilon, 0)$-differentially privacy. In our approach, $\Delta f = 1$. Thus the *PrivacyTransformation* step can achieve $(\varepsilon, (e^{\varepsilon} + 1)\frac{2}{x^2\varepsilon^2})$-differentially privacy. For the *SketchVectorGeneration* step, it only accesses a differentially private frequency vector, not the underlying database. As proven by Michael et al. (2010), a post-processing of differentially private results remains differentially private. Therefore, Algorithm 1 as a whole maintains $(e^{\varepsilon} + 1)\frac{2}{x^2\varepsilon^2})$-differentially privacy. □

6 Experiments

6.1 Dataset

For our experiments we used a large dataset of GPS vehicles traces, collected in a period from 1st May to 31st May 2011. In our simulation, the coordinator collects the FV from all the vehicles to determine the Global Frequency Vector (GFV), i.e. the sum all the trajectories crossing any link, at the end of each day, so we defined a series of time intervals τ_i, where each τ_i spans over a single day. In the following we show the resulting GFV for the 25th May 2011, but similar accuracy is observed also for the other days. The GPS traces were collected in the geographical areas around Pisa, in central Italy, and it counts for around 4,200 vehicles, generating around 15,700 trips.

6.2 Space Tessellation

The generalization and aggregation of movement data is based on space partitioning. Arbitrary territory divisions, such as administrative districts or regular grids, do not reflect the spatial distribution of the data. The resulting aggregations may not convey the essential spatial and quantitative properties of the traffic flows over the territory. Our method for territory partitioning extends the data-driven method suggested in chapter Andrienko and Andrienko (2011). Using a given sample of points (which may be, for example, randomly selected from a historical set of movement data), the original method finds spatial clusters of points that can be enclosed by circles with a user-chosen radius. The centroids of the clusters are then taken as generating seeds for Voronoi tessellation of the territory. We have modified the method so that dense point clusters can be subdivided into smaller clusters, so that the sizes of the resulting Voronoi polygons vary depending on the point density: large polygons in data-sparse areas and small polygons in data-dense areas. The method requires the user to set 3 parameters: maximal radius R, minimal radius r, and minimal number of points N allowing a cluster to be subdivided. In our experiments, we used a tessellation with 2,681 polygons obtained with R = 10 km, r = 500 m, N = 80.

6.3 Utility Evaluation

In the proposed framework, the coordinator collects the Frequency Vectors from all the vehicles in the time interval τ and aggregate them to obtain the resulting GFV, representing the flow values for each link of the spatial tessellation. Each FV received from the vehicles is perturbed by means of a two-step transformation:

Table 1 Reduced sizes of the FV for different values of α and γ

	α	γ	Columns (w)	Rows (d)	$w \times d$
CM_{5k}	0.0008	0.1	2,500	2	5,000
CM_{7k}	0.00078	0.05	2,564	3	7,692
CM_{10k}	0.00057	0.05	3,508	3	10,524

privacy transformation—with the objective of protecting sensitive information–, and sketches summarization—to reduce the volume of communication to be sent. These two transformations are regulated by two set of parameters: ε for the differential privacy transformation, and α and γ for the Count-Min Sketch summarization. When ε tends to 1 very little perturbation is introduced and this yields a low privacy protection. On the contrary, better privacy guarantees are obtained when ε tends to zero. The two parameters α and γ regulate the compression of the FV to be sent to the coordinator. Table 1 shows how the choice of these two parameters influences the final size of the FV. For example, for $\alpha = 0.0008$ and $\gamma = 0.1$ the original FV of 16k entries is reduced to a vector of 5k cells.

Since the two transformations operate on the entries of the FV, and hence on the flows, we compare two measures: (1) the *flow per link* (*fpl*), i.e. the directed volume of traffic between two adjacent zones; (2) the *flow per zone* (*fpz*), i.e. the sum of the incoming and outgoing flows in a zone. Figure 1 shows the resulting distributions of different privacy transformation with $\varepsilon = 0.9, 0.5, 0.3$. Figure 1 (*left*) shows the reconstructed flows per link: fixed a value of flow (x) we count the number of links (y) that have that flow. Figure 1 (*right*) shows the distribution of sum of flows passing for each zone: given a flow value (x) it shows how many zones (y) present that total flow.

From the distribution we can notice how the privacy transformation preserves very well the distribution of the original flows, even for more restrictive values of the parameter ε.

When we consider several flows together, like those incident to a given zone [Fig. 1 (*right*)], the distribution curves present several local variations, however the general shape is preserved for all the privacy transformations. Since the global distributions are comparable, we choose a value 0.3 for ε for the following discussions, in order to obtain a better privacy protection.

Fixed the privacy transformation parameter, we can evaluate the error introduced by the Count-Min sketch summarization. In Fig. 2 we can appreciate how a large compression of the FV yields a precise reconstruction of the transformed flows. In fact, we can observe that the general shape of the distribution curves are also preserved after the application of sketching techniques.

To maintain the data utility for mobility density analysis, we want to preserve the relative density distribution over the zones, i.e. it is desirable that former zones with low (high) traffic still present low (high) traffic after the transformations. To check this property, we show in Fig. 3 the correlation plots to compare the original flows with the transformed ones. From the charts we can notice how the

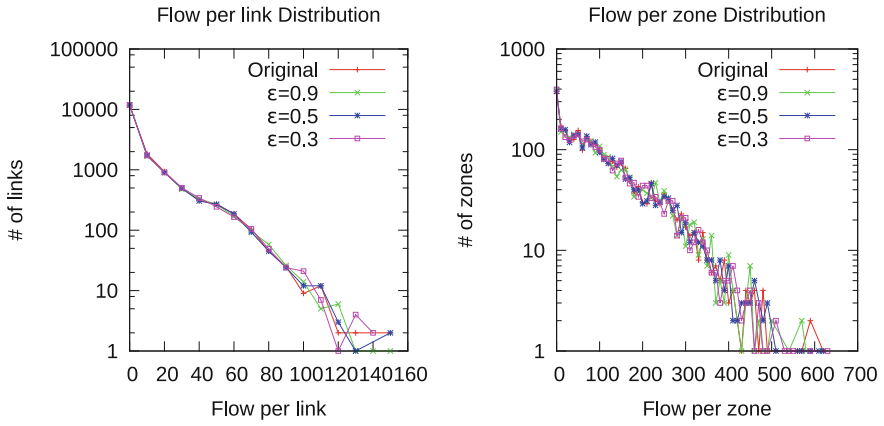

Fig. 1 Distribution of flow per link (*left*) and flow per zone (*right*)

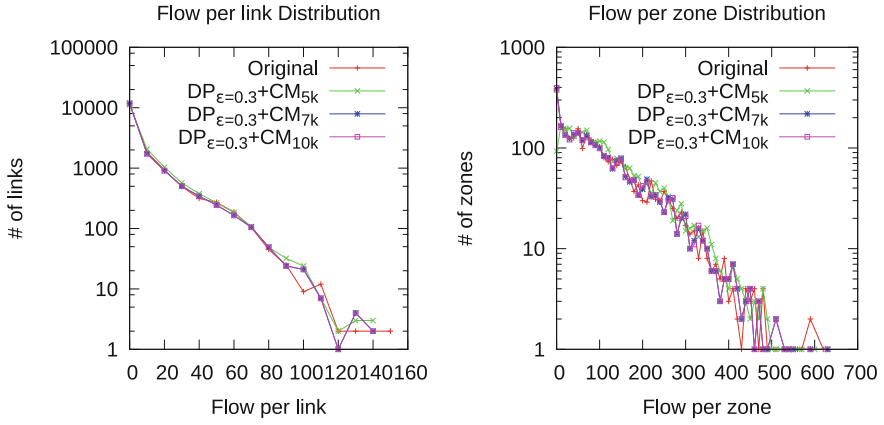

Fig. 2 Distribution of flow per link (*left*) and flow per zone (*right*) after the Count-Min sketch transformation

transformed flows maintain a very strong correlation with the original ones, enabling relative flows comparisons also in the transformed data.

Qualitatively, Fig. 4 shows a visually comparison of each Sketch summarization with the original flows. Each flow is draw with arrows with thickness proportional to the volume of trajectories observed on a link. From the figure it is evident how the relevant flows are preserved in all the transformed GFV, revealing the major highways and urban centers.

Similarly, the flow per zone is also preserved, as it is shown in Fig. 5, where the flow per each cell is rendered with a circle of radius proportional to the difference from the median value of each GFV. The maps allow us to recognize the dense areas (red circles, above the median) separated by sparse areas (blue circle below

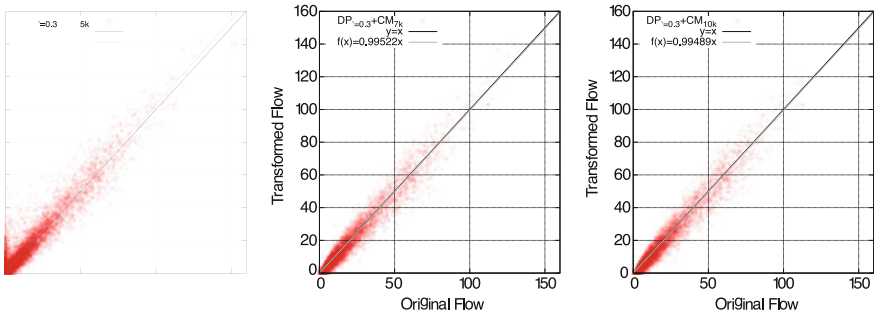

Fig. 3 Correlation between original flows and transformed flows with DP $\varepsilon = 0.3$ and CM_{5k} (*first*), CM_{7k} (*second*), CM_{10k} (*third*)

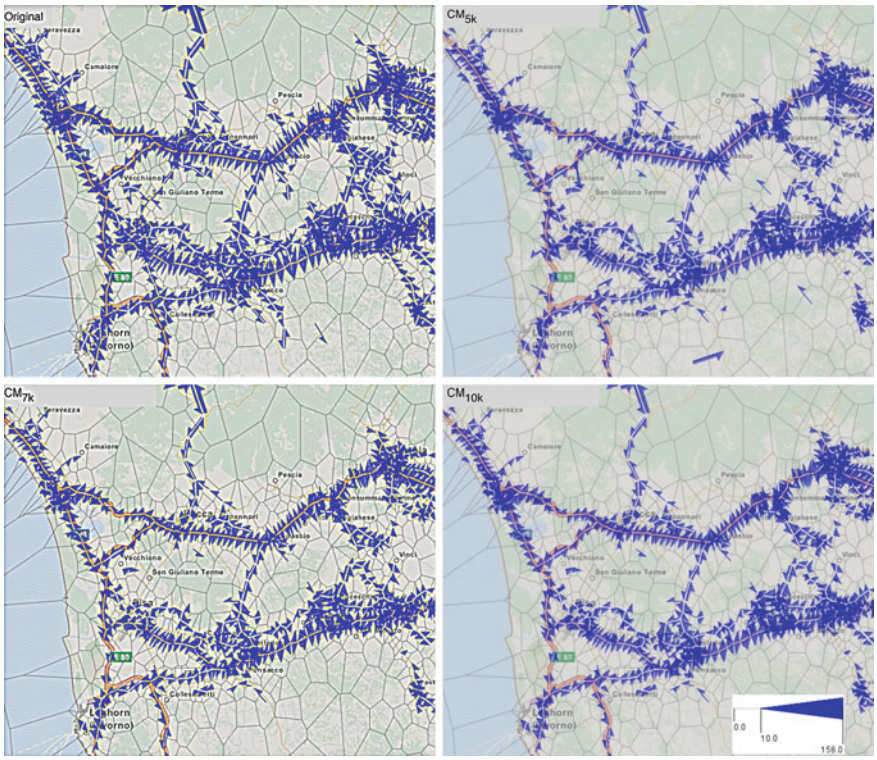

Fig. 4 Comparison of the original flows (**a**) with the GMF obtained with $\varepsilon = 0.3$ and CM_{5k} (**b**), CM_{7k} (**c**), and CM_{10k} (**d**)

the median). The high density traffic zones follow the highways and the major city centers along their routes.

The two comparisons proposed above give the intuition that, while the transformations protect individual sensitive information, the utility of data is preserved.

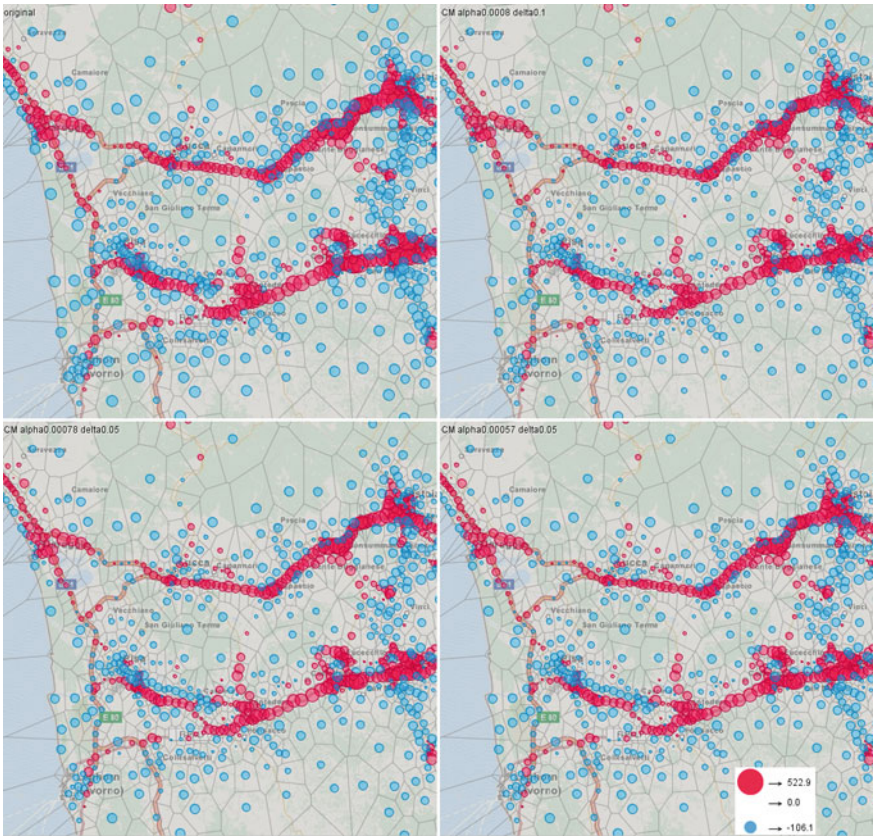

Fig. 5 Traffic flow per zone drawn with *circles* proportional to the difference from the median for each transformation with privacy transformations with different parameters (**a**), without privacy transformation; $\varepsilon = 0.3$ and CM_{5k} (**b**), CM_{7k} (**c**), and CM_{10k} (**d**)

7 Related Work

The existing methods of privacy-preserving publishing of trajectories can be categorized into two classes: (1) generalization/suppression based data perturbation, and (2) differential privacy.

Generalization/suppression based data perturbation techniques. There have been some recent works on privacy-preserving publishing of spatio-temporal moving points by using the generalization/suppression techniques. The mostly widely used privacy model of these work is adapted from what so called k-anonymity (Samarati and Sweeney 1998a, b), which requires that an individual should not be identifiable from a group of size smaller than k based on their quasi-identifies (QIDs), i.e., a set of attributes that can be used to uniquely identify the individuals. Abul et al. (2008) proposes the (k, δ)-anonymity model that exploits

the inherent uncertainty of the moving object's whereabouts, where δ represents possible location imprecision. Terrovitis and Mamoulis (2008) assume that different adversaries own different, disjoint parts of the trajectories. Their anonymization technique is based on *suppression* of the dangerous observations from each trajectory. Yarovoy et al. (2009) consider timestamps as the quasi-identifiers, and define a method based on *k -anonymity* to defend against an attack called *attack graphs*. Monreale et al. (2010) propose a spatial generalization approach to achieve *k*-anonymity. A general problem of these *k*-anonymity based privacy preserving techniques is that these techniques assume a certain level of background knowledge of the attackers, which may not be available to the data owner in practice.

Differential privacy. The recently proposed concept of *differential privacy* (DP) (Dwork et al. 2006) addresses the above issue. There are two popular mechanisms to achieve differential privacy, *Laplace* mechanism that supports queries whose outputs are numerical (Dwork et al. 2006) and *exponential mechanism* that works for any queries whose output spaces are discrete (McSherry and Talwar 2007). The basic idea of the Laplace mechanism is to add noise to aggregate queries (e.g., counts) or queries that can be reduced to simple aggregates. The Laplace mechanism has been widely adopted in many existing work for various data applications. For instance, Xiaokui et al. (2011), Cormode et al. (2012b) present methods for minimizing the worst-case error of count queries; Barak et al. (2007), Ding et al. (2011) consider the publication of data cubes; Michael et al. (2010), Xu et al. (2012) focus on publishing histograms; and Mohammed et al. (2011), Ninghui et al. (2012) propose the methods of releasing data in a differential private way for data mining. On the other hand, for the analysis whose outputs are not real or make no sense after adding noise, the exponential mechanism selects an output from the output domain, $r \in R$, by taking into consideration its score of a given utility function q in a differentially private manner. It has been applied for the publication of audition results (McSherry and Talwar 2007), coresets (Feldman et al. 2009), frequent patterns (Bhaskar et al. 2010) and decision trees (Friedman and Schuster 2010).

Regarding publishing differentially private spatial data, Chen et al. (2012) propose to release a prefix tree of trajectories with injected Laplace noise. Each node in the prefix tree contains a doublet in the form of $< tr(v), c(v) >$, where $tr(v)$ is the set of trajectories of the prefix v, and $c(v)$ is a version of $|tr(v)|$ with Laplace noise. Compared with our work, the prefix tree in Chen et al. (2012) is *data-dependent*, i.e., it should have a different structure when the underlying database changes. In our work, the frequency vector is *data-independent*. Cormode et al. (2012b) present a solution to publish differentially private spatial index (e.g., quadtrees and kd-trees) to provide a private description of the data distribution. Its main utility concern is the accuracy of multi-dimensional range queries (e.g., how many individuals fall within a given region). Therefore, the spatial index only stores the count of a specific spatial decomposition. It does not store the movement information (e.g., how many individuals move from location i to location j) as in

our work. In another chapter, Cormode et al. (2012c) proposes to publish a contingency table of trajectory data. The contingency table can be indexed by specific locations so that each cell in the table contains the number of people who commute from the given source to the given destination. The contingency table is very similar to our frequency vector structure. However, Cormode et al. (2012c) has a different focus from ours: we investigate how to publish the frequency vector in a differential privacy way, while Cormode et al. (2012c) address the sparsity issue of the contingency table and presents a method of releasing a compact summary of the contingency table with Laplace noise.

There are some work on publishing time-series data with differential privacy guarantee (McSherry and Mahajan 2010; Rastogi and Nath 2010). Since we only consider spatial data, these work are complement to our work.

8 Conclusion

In this chapter, we have studied the problem of computing movement data aggregation based on trajectory generalization in a distributed system while preserving privacy. We have proposed a method based on the well-known notion of differential privacy that provides very nice data protection guarantees. In particular, in our framework each vehicle, before sending the information about its movements within a time interval, applies to the data a transformation for achieving privacy and then, creates a summarization of the private data (by using a sketching algorithm) for reducing the amount of information to be communicated. The results obtained in our experiments show that the proposed method preserves some important properties of the original data allowing the analyst to use them for important mobility data analysis.

Future investigations could be directed to explore other methods for achieving differential privacy; as an example, it would be interesting to understand the impact of the use of the geometric mechanism instead of the Laplace one for achieving differential privacy.

Acknowledgments This work has been partially supported by EU FET-Open project LIFT (FP7-ICT-2009-C n. 255951) and EU FET-Open project DATA SIM (FP7-ICT 270833)

References

Abul O, Bonchi F, Nanni M (2008) Never walk alone: uncertainty for anonymity in moving objects databases. In: Proceedings of the 2008 IEEE 24th international conference on data engineering (ICDE), pp 376–385

Andrienko N, Andrienko G (2011) Spatial generalization and aggregation of massive movement data. IEEE Trans Visual Comput Graphics 17:205–219

Backes M, Meiser S (2012) Differentially private smart metering with battery recharging. IACR cryptology ePrint archive, p 183

Barak B, Chaudhuri K, Dwork C, Kale S, McSherry F, Talwar K (2007) Privacy, accuracy, and consistency too: a holsistic solution to contingency table release. In: Proceedings of the 26th ACM SIGMOD-SIGACT-SIGART symposium on principles of database systems (PODS), pp 273–282

Bhaskar R, Laxman S, Smith A, Thakurta A (2010) Discovering frequent patterns in sensitive data. In: Proceedings of the 16th ACM SIGKDD international conference on knowledge discovery and data mining (KDD), pp 503–512

Chen R, Fung BCM, Desai BC, Sossou NM (2012) Differentially private transit data publication: a case study on the montreal transportation system. In: Proceedings of the 18th ACM SIGKDD international conference on knowledge discovery and data mining (KDD), pp 213–221

Cormode G, Muthukrishnan S (2005) An improved data stream summary: the count-min sketch and its applications. J Algorithms 55(1):58–75

Cormode G, Garofalakis MN (2008) Approximate continuous querying over distributed streams. ACM Trans Database Syst 33(2)

Cormode G, Garofalakis MN, Haas PJ, Jermaine C (2012a) Synopses for massive data: samples, histograms, wavelets, sketches. Found Trends Databases 4(1–3):1–294

Cormode G, Procopiuc CM, Srivastava D, Shen E, Yu T (2012b) Differentially private spatial decompositions. In: ICDE, pp 20–31

Cormode G, Procopiuc CM, Srivastava D, Tran TTL (2012c) Differentially private summaries for sparse data. In: ICDT, pp 299–311

Ding B, Winslett M, Han J, Li Z (2011) Differentially private data cubes: optimizing noise sources and consistency. In: Proceedings of the 2011 ACM SIGMOD international conference on management of data, pp 217–228

Dwork C, McSherry F, Nissim K, Smith A (2006) Calibrating noise to sensitivity in private data analysis. In: Proceedings of the 3rd conference on theory of cryptography (TCC), pp 265–284

Feldman D, Fiat A, Kaplan H, Nissim K (2009) Private coresets. In: Proceedings of the 41st annual ACM symposium on theory of computing (STOC), pp 361–370

Friedman A, Schuster A (2010) Data mining with differential privacy. In: Proceedings of the 16th ACM SIGKDD international conference on knowledge discovery and data mining, pp 493–502

Hay M, Rastogi V, Miklau G, Suciu D (Sep 2010) Boosting the accuracy of differentially private histograms through consistency. Proc VLDB Endow 3(1–2):1021–1032

Kifer D, Machanavajjhala A (2011) No free lunch in data privacy. In: Sellis TK, Miller RJ, Kementsietsidis A, Velegrakis Y (eds) ACM-SIGMOD conference, pp 193–204

Li N, Qardaji WH, Su D, Cao J (2012) Privbasis: frequent itemset mining with differential privacy. PVLDB 5(11):1340–1351

McSherry F, Mahajan R (2010) Differentially-private network trace analysis. In: Proceedings of the ACM SIGCOMM 2010 conference, pp 123–134

McSherry F, Talwar K (2007) Mechanism design via differential privacy. In: Proceedings of the 48th annual IEEE symposium on foundations of computer science (FOCS), pp 94–103

Mohammed N, Chen R, Fung BCM, Yu PS (2011) Differentially private data release for data mining. In: Proceedings of the 17th ACM SIGKDD international conference on knowledge discovery and data mining

Monreale A, Andrienko GL, Andrienko NV, Giannotti F, Pedreschi D, Rinzivillo S, Wrobel S (2010) Movement data anonymity through generalization. Trans Data Priv 3(2):91–121

Rastogi V, Nath S (2010) Differentially private aggregation of distributed time-series with transformation and encryption. In: SIGMOD, pp 735–746

Samarati P, Sweeney L (1998a) Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppresion. In: Proceedings of the IEEE symposium on research in security and privacy, pp 384–393

Samarati P, Sweeney L (1998b) Generalizing data to provide anonymity when disclosing information(abstract). In: Proceedings of the 17th ACM symposium on principles of, database systems (PODS)

Terrovitis M, Mamoulis N (2008) Privacy preservation in the publication of trajectories. In: Proceedings of the 9th international conference on mobile data management (MDM)

Xiao X, Wang G, Gehrke J (Aug 2011) Differential privacy via wavelet transforms. IEEE Trans Knowl Data Eng 23(8):1200–1214

Xu J, Zhang Z, Xiao X, Yang Y, Yu G (2012) Differentially private histogram publication. In: ICDE, pp 32–43

Yarovoy R, Bonchi F, Lakshmanan LVS, Wang WH (2009) Anonymizing moving objects: how to hide a mob in a crowd? In: EDBT, pp 72–83