Geoinformatica (2014) 18:819–842
DOI 10.1007/s10707-014-0205-7

Location privacy models in mobile applications: conceptual view and research directions

Maria Luisa Damiani

Received: 5 June 2013 / Revised: 10 January 2014 /
Accepted: 3 February 2014 / Published online: 13 March 2014

Abstract Location privacy in mobile, location-aware applications is a prominent research topic spanning across different disciplines and with strong societal implications and expectations. The tumultuous growth of the mobile applications market over the past few years has however hindered the development of a systematic organization and classification of location privacy concepts. In this paper we focus on one of the key concepts, i.e. *location privacy metric*. We survey existing approaches to the measurement of location privacy and propose a classification framework. The notion of location privacy metric, however, cannot be fully understood without describing the context in which these metrics are used. To that extent we elaborate on the notions of *application model* and *privacy model*. The ultimate goal is to contribute to the specification of a conceptual framework for location privacy.

Keywords Location privacy · Location privacy metric · Location-aware applications

1 Introduction

Individual location is an enabling factor in a variety of mobile applications such as location-based services (LBS), mobile sensing, geo-location services, location sharing in social networks. In all such applications the user's location is communicated to a third party. This raises challenging issues for privacy because location and in particular the history of locations (*trajectories*) can reveal details about an individual's personal life. Sharing location data with potentially untrustworthy parties, e.g. service providers and even members of a community, may thus result in a loss of control that exposes data to possible abuses.

In most countries, location data cannot be collected without providing users with privacy guarantees in compliance with data protection legislation which contains obligations for data controllers (e.g. application providers) and rights and guarantees for data subjects

M. L. Damiani (✉)
Department of Computer Science, University of Milan, Via Comelico 39, 20135 Milan, Italy
e-mail: damiani@dico.unimi.it

(i.e. users). For example, in Europe, the ePrivacy Directive[1] indicates that location data other than traffic data (e.g. public communications networks logs) may only be processed when users "are made anonymous or with prior consent and only for the duration necessary for the provision of a value added service". Unfortunately, privacy regulations cannot protect personal information, e.g. location, against malicious or curious parties deliberately trying to access such data without the user's consent. This is instead the goal of location privacy technologies.

Current research on location privacy technologies focuses on two major application scenarios: (1) trajectory data publishing; and (2) on-line information services. In the former case, the goal is typically to protect the business organizations willing to capitalize on customers' private data asset by releasing trajectories data to third parties without incurring privacy violations. Trajectory data can regard for example the vehicles covered by a pay-as-you-drive insurance or the customers of a telecommunication company. By contrast, in on-line applications, the goal is to protect end-users when dynamically exchanging their location for information services. Of these two scenarios, the latter is technically (and socially) more challenging: i) the users' trajectories are only partially known (only the current and past locations are known but not the future ones); (ii) the service must be provided without delays; (iii) privacy requirements can differ from user to user. Following common terminology, in what follows we refer to this class of applications as location-aware applications.

Location-aware applications are rapidly evolving. Beyond traditional LBS offering search services, new classes of services are emerging, such as mobile sensing, volunteered geography, geo-location services and location sharing, each presenting different, though not necessarily disjoint, location privacy requirements. Unfortunately the lack of a unifying framework offering a systematic organization of concepts has led to a proliferation of location privacy solutions across the various domains which are difficult to relate to each other [49]. This makes it difficult to achieve a common understanding across the various research communities as well as the development of engineering practices, for example to support the elicitation and fulfillment of privacy requirements [21]. In this view, a conceptual approach to the analysis of location privacy is definitely critical.

With this motivation, in this paper we focus on the key concept of *location privacy metric*. A location privacy metric enables the estimation of the level of protection offered to users. We survey relevant approaches to the measurement of location privacy and propose a possible classification. However, these approaches cannot be fully understood without taking into account the context in which the privacy metrics can be used. To overcome the heterogeneity of application scenarios, we abstract relevant aspects of applications in the concept of *application model*.

An application model is an abstraction of the information flows between the various parties involved in location-aware applications. Understanding the information flows is key to understanding the location privacy issue. We start from this notion to define the concepts of privacy goal, privacy mechanism, and privacy metrics, collectively summarized in the term of *privacy model*. The ultimate objective of this discussion is to contribute to the definition of a unifying conceptual framework for location privacy.

The rest of the paper is organized as follows. Section 2 presents the reference architecture along with some preliminary assumptions, regarding as well the meaning of location privacy. Section 3 introduces the notion of application model and applies such an abstraction

[1] Directive 2002/58/EC-Article 9. See: http://eur-lex.europa.eu

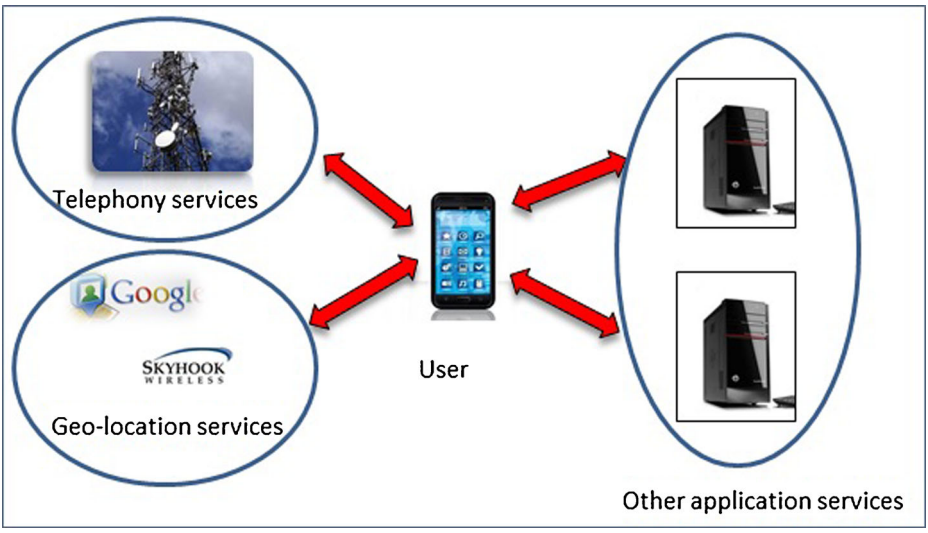

Fig. 1 Reference framework: users reveal their whereabouts to different service providers

to characterize major classes of location-aware applications. Section 4 introduces the concept of privacy model along the three dimensions, i.e. privacy goal, mechanism and metric. Privacy mechanisms are briefly surveyed in Section 5 while the notion of privacy metric is discussed in Section 6. Finally Section 7 concludes the article presenting possible directions for future research.

2 Preliminaries: reference architecture and assumptions

We begin describing the reference architectural framework. Next we analyze common definitions of location privacy.

2.1 Reference architecture

We assume the distributed, client-server architecture illustrated in Fig. 1. Users are equipped with high-end devices, e.g. gps and wifi-enabled smartphones/tablets and can access various services offered by different providers. In particular users can access two main kind of services, i.e. communication services and application services.

In both cases, the service providers are aware of users' location, as described next:

– Communication services enable the acquisition of users' location from communication data. Services of this category include mobile telephony services and Internet communication services. For example, users have normally access to a mobile telephony network, e.g. GSM network. A typical GMS network is structured into cells of known size and location, each served by a single base tranceiver station (BTS). To establish a connection to the mobile phone in the case of an incoming connection request, the system has to know whether the mobile phone is available and in which cell it is currently located [45]. This information can be provided by the mobile phone itself either peri

odically or when the cell changes, or, alternatively, be determined by the network. In any case, the service provider knows the user's location. The location can achieve an accuracy up to 50 meters (in case of emergency) [2].

Another simple way to obtain location information from communication data is through the IP address. The IP address can be matched against one of the numerous datasets reporting the correspondence between IP addresses and geographical coordinates, to obtain a coarse location at the granularity of city or even campus.

– The application services rely on the location information transmitted by users. A popular class of application services are the LBS responding to queries such as *which is the closest point of interest*. A more recent class of services consists of the geo-location services for the estimation of the user's location across indoor and outdoor settings. Geo-location services are used alternatively or in conjunction with GPS-based positioning. They ensure a broader coverage and lower energy consumption than GPS, though at the price of a lower location accuracy, and for this they are widely used. Note that the user's coordinates are determined by matching the communication network infrastructure sensed in proximity of the mobile device, i.e. wi-fi access points and BTS, against a proprietary database of geo-referenced networking elements [14, 34]. Thus the service providers know the user's location. The resulting location can reach an accuracy of a few tens of meters.

In the rest of the paper, we assume that the information flows of interest for the discussion on privacy are: (a) those that convey fine-grained locations; (b) those that convey location data to application services providers, i.e. the communication services are assumed trusted and secure. In reality, whether e.g. telephony services are trusted or conversely may endanger privacy is a debatable question (see for example [45]). We base our assumption on the fact that in certain countries, e.g. Europe, communication services are subject to more restrictive data protection norms than application services [4], and this can mitigate the privacy risk.

2.2 Location privacy: what definition?

The most widely used notion of privacy is that of control over personal information. This theory of privacy is also known as information privacy. One of the most influential and quoted definitions goes back to 1967 and has been developed by Alan Westin, professor of Public Law and Government at the Columbia University [56]: "Privacy is the claim of individuals, groups, or institutions to determine for themselves when, how, and to what extent information about them is communicated to others." Information privacy underlies existing privacy regulations. Law provides people with a set of rights to enable them to make decisions about how to manage their data where these rights consist primarily of rights to notice, access, and consent regarding the collection, use, and disclosure of personal data [50]. Building on this theory of privacy, Duckham and Kulik (2006) define *location privacy* as a special type of information privacy *which concerns the claim of individuals to determine for themselves when, how, and to what extent location information about them is communicated to others* [22]. To our knowledge this is the only abstract conceptualization of location privacy proposed so far in the computer science realm. In reality, the relation between privacy and control is a debated question in social sciences (the interested reader can refer to e.g. Solove [50] and Nissembaum [42] for a recent discussion). For example, Tavani and Moor [52] argue that identifying the concept of privacy with control is misleading because it would greatly reduce what can be private, in other terms personal information

ought to be private even if the owner is not in a position to control it. For example a patient should not lose his right to have his medical rights protected when he is under anesthesia. While it is beyond the scope of this work to discuss the philosophical and legal dimension of privacy, it is worth noting that there exists a tension between privacy and control. As we will see later on, this tension somehow reflects itself on the technologies for privacy protection.

3 Application model

Location-aware applications are continuously evolving, opening up novel privacy concerns. Because of the diversity of these applications, defining a unique reference model around which to frame privacy problems would result in an oversimplification of the privacy issues. To preserve the specificity of applications, the idea is to build abstractions, i.e. application models, using a common set of concepts, grounded on the notion—key for privacy—of *information flow*. In what follows we first present a characterization of locations (information) flows, then we discuss major classes of applications in the light of this characterization.

3.1 Meta-model

The *application model* is an abstraction of the information flows typical of a class of applications. Specifically an application model, denoted as pair (A, I), specifies a set A of application service providers and a set I of information flows between the user and the parties in A. In what follows, we call *users* both the mobile devices and the device holders. Let $u \in U$ be a representative member of the set U of users. We define the application model (A, I) as follows:

- The set $A = \{a_1, .., a_n\}$ denotes the service providers (simply *providers* hereinafter) jointly contributing to offer an information service to user u. We distinguish between the applications in which there is only one provider, i.e. $|A| = 1$ and those in which there can be multiple providers, i.e. $|A| > 1$. We call these applications as *single-party* and *multi-party*, respectively. As we will see, this distinction is relevant for privacy.
- The set $I = \{I_{a_1}, .., I_{a_n}\}$ describes the information flows between user u and providers $a_1, .., a_n$. Every information flow I_{a_i} has a *direction*, a *request content* and a *frequency*:
  - Direction. The information flow can be either *query-based or transaction-based* depending on whether the communication is two-way or one-way respectively. In the former case the provider a_i returns an immediate answer in response to a request, thus the communication is two-way. In the latter case, the user conveys a request, for example to transfer data to a_i, without waiting for an answer, thus the communication is one-way.
  - Request content. A generic service request takes the form: $< id, t, loc, c >$ with the attributes denoting the user's identifier, time, location and content, respectively. Of these, the content is specific to the application class, e.g. it can be a spatial query, a function, a measurement or be empty.
  - Frequency. The information flow can be either *continuous* or *sporadic*. It is continuous when the service requests are closely distributed in time. In such

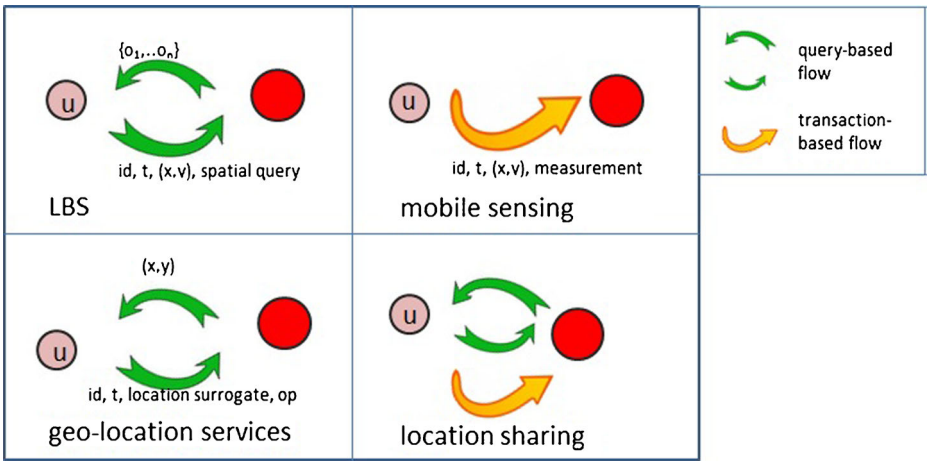

Fig. 2 Information flows in example application models between the user U and the provider. The arrows indicate the information flow direction

a case, subsequent locations of the requester are interrelated, i.e. the user's location at time t_i is dependent on the user's location at time t_{i-1}. Conversely, if the next location does not depend on the previous one, the information flow is sporadic.

In the most general case, there can be more than one information flow between user u and provider a_i. For example, if a_i is the provider of a location sharing application (e.g. a find-friend application), the user can continuously convey his location to a_i and at the same time sporadically query a_i about the location of other users. In such a case, there are two information flows, the first continuous and transaction-based, and the other sporadic and query-based. Conventionally, I_a^j denotes the jth information flow.

3.2 Examples of single-party application models

We now use the above framework to propose a concise description of representative single-party applications models, in particular: LBS, geo-location services, mobile sensing, and location sharing services. Note that the goal is not to provide a survey of location-aware applications, but simply to propose an abstract, privacy-oriented characterization of popular applications. The application models take the simple form (a, I_a), where a is the unique provider. The information flow is peculiar to each application model. Figure 2 illustrates the four cases described in the following.

LBS. We restrict attention to popular search-based LBS requesting for example the closest points of interest. The information flow is:

- Query-based
- The content of the request is a spatial query, e.g. k-nearest neighbors. Upon a request, the provider returns a set of geo-referenced and application-dependent objects i.e. $\{(o_i, x_i, y_i)\}_{i\geq 0}$
- LBS can be either sporadic or continuous. If continuous, the result of the query is repeatedly updated until the query is explicitly ended.

Geo-location services. We recall that these services can estimate the geographical location of the user (i.e. the coordinates) based on an environmental footprint. The information flow is:

- Query-based
- Request content: the location conveyed to provider a is a location surrogate e.g. the set of wi-fi access points near to the user. The content can detail the geo-location operation. The provider returns the coordinates of the estimated point (possibly along with an accuracy measure)
- Sporadic and continuous services can be requested, for example, through the W3C geo-location API (Application Programming Interface) standard [18, 55].

Mobile sensing. These services enable the acquisition of geo-referenced data, i.e. pollution measurements, from sensors installed on the mobile device [35].

The information flow is as follows:

- Transaction-based
- The request content typically specifies the measurement at a point. The provider is also called *aggregator*
- The information flow can be either continuous or sporadic. *Location tracking* can be seen as special case of mobile sensing where the request content is empty.

Location sharing. Location sharing is a major functionality of geo-social networks [37]. A typical find-friend application has been described in a previous example. It consists of two information flows, that we can represent as: $\left(a, \left\{I_a^1, I_a^2\right\}\right)$ where I_a^1 is a sporadic LBS, i.e. a query-based flow for retrieving the friends' locations; and I_a^2 a continuous location tracking service, i.e. a transaction-based flow reporting the user's location.

3.3 Example of multi-party application models

In general, applications may exhibit complex information flows involving more than one provider. Figure 3 shows the two-party scenario that, in recent years, has become widely popular, following the launch of commercial geo-location services by e.g. Google. The application work-flow is as follows: the mobile device first requests a geo-location service to the provider a_1 to obtain the coordinated point, say p; next p is forwarded along with the service request to provider a_2 for example to access a LBS. We omit the details of the application model. Note that, in this specific scenario, the geo-location service provider can be accessed by different location-aware applications. The provider is thus in the position of collecting huge amounts of location data.

An aspect of the multi-party scenario that deserves some attention for its implications for privacy, is the fact that the different services can be provided by the same or closely linked business organizations, e.g. a_1 and a_2 are provided by the same company. In this case we say that the providers may *collude*. As we will see, colluding providers can significantly complicate location privacy protection.

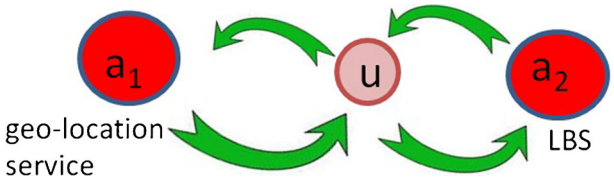

Fig. 3 Two-party information flow: the user requests services to two providers, offering geo-location and social networking services, respectively

4 Privacy model

We now inject into the application model the location privacy issue. The privacy concern arises because locations are collected by providers. Location represents personal data and thus the collection and use of such data is subject to privacy regulations. We use the term *data controller* to designate any entity, individuals or companies, which collects and uses personal data where personal data is defined as any information relating to an identified or identifiable natural person[2] . The obligations are addressed to the data controller. In our scenario a provider is a controller.

4.1 Hard vs. soft privacy technologies

A key distinction is between trusted and untrusted controllers. Trusted controllers are expected to behave in compliance with privacy regulations, e.g. a credit card company. Untrusted controllers are those that may attempt to gain benefit from the processing of personal data without user's consent, i.e. controllers are assumed to be honest but curious. Whether a controller is trusted or not, depends on various factors, not least the reputation of the party. The distinction between trusted and untrusted controllers is at the heart of the two major technological approaches to privacy protection, well synthesized in the concepts of *soft privacy* and *hard privacy*, respectively [21], where:

- Soft privacy solutions apply when the controller is trusted. The goal is to ensure that users are aware of how their personal information is used. These techniques supplement the controller's legal notice. Soft privacy solutions include a variety of techniques, such as the specification of machine enforceable location privacy policies (see early work on extensions of the P3P standard [41]), location privacy preferences languages [24, 53] and development of privacy-aware API [55].
- Hard privacy solutions apply when at least one of the controllers is untrusted. The goal of hard privacy is to minimize the transfer of data so as to reduce the need for trust. Hard privacy solutions tend to automate or significantly limit user's intervention in the configuration and management of privacy.

It can be seen that the amount of control that users can exercise is generally different for the two classes of solutions. We argue that the dualism hard vs. soft privacy mirrors the juxtaposition between privacy and control discussed in Section 2.2. More specifically, hard privacy is aligned with the idea that privacy is a value by itself and has to be protected independently from the control the user can exercise.

[2]European Data Protection Directive (95/46/EC)

Hard privacy is where the discussion in the article naturally fits in. Notice that there is no a-priori superiority of one class of solutions against the other, i.e. the convenience depends on the application context. Moreover, as we will see in Section 5.4, there are cases in which it might be convenient to combine hard and soft privacy in hybrid solutions.

4.2 Privacy model dimensions

We call *privacy model* the abstraction of a hard privacy solution. The untrusted controller is the *adversary*. We characterize a privacy model in terms of the following dimensions:

(i) privacy goal
(ii) privacy mechanism
(iii) privacy metric

where (i) specifies the user's privacy requirement; (ii) the technique used to achieve the privacy goal, and (iii) the criteria for the quantification of the level of protection provided by the privacy mechanism. The privacy goals are described in the following, while the concepts of privacy mechanism and the privacy metric will be discussed in the next two sections.

4.2.1 Privacy goals

There are two classical privacy goals in location-aware applications, namely identity protection and location protection [10, 32]. In addition to these two, we mention the less emphasized, although definitely relevant, protection of user's behavior [15]. These goals are briefly summarized as follows:

- **Identity protection**. The goal is to protect user's anonymity. Both users and providers can benefit from the user's anonymity. For example the users of a LBS offered to the members of a community potentially subject to discrimination, e.g. gay community, might understandably ask to remain anonymous. Moreover if users are anonymous, providers are not compelled by law to obtain users' prior consent and limit data usage. That is especially relevant when data is obtained from a potentially large population of users and/or data is potentially useful for purposes difficult to predict in advance. Ensuring anonymity is however challenging. A major challenge comes from the fact that simply replacing the user's identifier with a pseudonym is not sufficient because location can be easily associated with external knowledge, in the simplest case a map, causing the disclosure of personal details, e.g. home address, unveiling the user's identity. Protecting location is therefore necessary to provide anonymity guarantees.
- **Location protection**. The goal is to protect the user's true location, independently of any other consideration, because location is valuable by itself. This is the goal commonly pursued in non-anonymous location-aware applications, such as most LBS and social networks. In such applications, users might not wish to disclose their detailed whereabouts while, for example, they request a LBS. Unfortunately users have limited control on the way location is communicated; at most they can specify their location at some coarse predefined granularity. The issue is how to provide more flexible solutions, without compromising the application utility. In general, location protection targets the undifferentiated protection of *every* location, independently from the location meaning.
- **Behavior protection.** The goal is to protect sensitive mobility patterns. We call mobility pattern the abstraction of an event or fact from the observation of the user's

movement. For example, a mobility pattern can be the presence in a place, e.g. in a restaurant, or the user's activity, e.g. shopping. A mobility pattern describes the user's behavior. We say that a mobility pattern is sensitive when it reveals a behavior that, for some reason, i.e. by law or personal preference, should be kept private. The importance of this form of protection is underlined by institutional bodies such as, in Europe, the Article 29 Data Protection Working Party that emphasizes the need of protecting specific patterns, i.e."visits to hospitals and religious places, presence at political demonstrations or presence at other specific locations revealing data about for example sex life" [4]. Note that the user's behavior can be inferred from the observation of a variety of contextual factors, e.g. time, geographical context, frequency of visits, and the presence of other individuals nearby [44]. Therefore, protecting the user's location using one of the privacy models discussed above, simply does not work [15].

5 Privacy mechanism

The privacy mechanism is the technique used to achieve the privacy goal. Abstractly, it can be defined as a transformation which maps every user's request into a different request before it becomes observable to the adversary. In the most general case, the transformation consists of two functions, the first for transforming the user's identity into a pseudonym based on some criteria (should the user be anonymous), and the latter for transforming the location [47]. For the sake of brevity, we restrict our attention to location transformation. We define a location transformation as a mapping from the reference space S, where the movement takes place, onto a possibly different space S', the space visible to the adversary, i.e.:

$$T : S \rightarrow S'$$

Location privacy mechanisms have been extensively investigated since the pioneering works of Gruteser et al. and Beresford et al. [6, 30]. The reader can refer to the excellent survey by Ghinita [25] for a comprehensive and detailed analysis of the state of the art. In this section we look at the question from a different angle. In particular we limit ourselves to present a possible set of criteria for the classification of location transformations. Such criteria will be used next to contextualize the discussion on privacy metrics. Specifically the criteria are:

(i) The reference space
(ii) The paradigm
(iii) The architecture

where (i) specifies the domain of the transformation; (ii) the kind of transformation; and (iii) the agent that applies the transformation.

5.1 Reference space

Transformation functions can be defined over different spaces. A popular distinction is between *unconstrained* and *constrained* space. The Euclidean space is the typical unconstrained space. Locations are 2D or 3D points. Accordingly a trajectory is a sequence of timestamped points in the free space. In the real world, however, the actual movement is generally confined by roads boundaries and other physical barriers. This complicates the

management of privacy, because, in general, the transformations defined for the unconstrained space, cannot be straightforwardly transposed into such a complex environment. This motivates the research of solutions tailored to constrained spaces.

A popular constrained space is the road network (actually any transportation network). In its basic form, a road network is represented by a graph $H = (V, E)$, where the vertices in V denote road intersections, and the edges in E road segments, embedded in the Euclidean plane. The road network is used as reference space, for example, in [40, 43, 54] where [40, 43] address identity protection in LBS and location tracking, respectively, and [54] location protection in LBS, e.g. route planning services. In all of these cases, the typical users are car drivers, thus move outdoor and their location is acquired through GPS. In the real world, however, users can be tracked pervasively, along roads and inside buildings, for example through their wifi-enabled smartphones. Confining the movement to the road network, as current transformations do, limits the generality of the privacy mechanism.

A more realistic urban space is considered in [58]. In this case the privacy goal is the protection of a specific mobility pattern, i.e. the presence in certain buildings reachable through a road network. The reference space is described by an augmented graph called *city annotated network*. An annotated city network is a graph $G = (V, E, pop, \mathcal{L}, tt)$ where the vertices in V include buildings or finer-grained *places* and the edges in E, entrances connecting places to road junctions. In addition, the functions $pop(.)$, $\mathcal{L}(.)$ and $tt(.)$ specify the popularity of places, the place semantics and the travel time respectively. The augmented graph is the domain of a specific transformation which maps a vertex $v \in V$ onto a properly defined subgraph $G' = (V', E')$ representing the obfuscated location.

In general, privacy mechanisms may not only require the definition of a suitable reference space but also of appropriate distance functions. For example, the approach in [26] uses two different distance functions, i.e. the Hausdorff metric, and the maximal distance between two regions—a pseudometric—to ensure that two consecutive transformed locations are at sufficient distance to guarantee the protection of the actual locations.

5.2 Paradigm

A paradigm identifies a class of location transformations. Privacy mechanisms are built on a limited number of paradigms, such as: location obfuscation, perturbation, confusion and suppression, and cryptographic techniques. These paradigms are briefly described in the following. For each paradigm, we also indicate the application classes in which the paradigm is mostly used and mention a few representative works. Let S be our reference space (constrained or unconstrained space), $loc \in S$ a user's location and T the location transformation.

– **Location obfuscation.** The transformation maps location loc onto a region r of space (i.e. a subset of space). It takes the form:

$$T(loc) = r \in 2^S \text{with } loc \in T(loc)$$

The technique is widely used across different applications, spaces and with different privacy goals, i.e. identity protection as in [11, 30], location protection e.g. [1, 10] and behavior protection e.g. [17, 36]. Note that obfuscating user's every location can compromise the utility of the applications demanding high positional accuracy, such as location sharing. An approach that can offer a good compromise between the need to preserve location quality and privacy, is *selective* obfuscation. Selective obfuscation means that only certain locations are obfuscated, typically those that satisfy certain

conditions. Selective obfuscation has been used for behavior protection in [17]. It will be recalled later on in the article.

– **Location perturbation.** The transformation maps *loc* onto a different location in proximity of the true location. The range of the transformation is thus the reference space itself, i.e.:

$$T(loc) = loc' \in S$$

The transformation can take various forms. For example it can be a deterministic function, e.g. a function computing the centroid of a set of points, or a random function returning a noisy location. Perturbation has been used in location tracking applications such as in the pioneering work [31]. This technique is not suited in those applications demanding high quality location data, such as geo-social networks [46].

– **Location confusion.** The true location is confused in a set of dummy locations. In particular the transformation maps location into a set of n locations of which one is the true location, i.e.:

$$T(loc) = \{loc_i\}_{i\in[1,n]} \text{ such that } \exists loc_i = loc$$

The use of this technique is generally confined to LBS applications. In response to the set of requests reporting the different locations, the provider returns a set of results, from which the user extracts the answer that corresponds to the true query. The paradigm is introduced in [33]. To reduce the communication overhead, an approach is to obtain the answer through an iterative process that stops when the user recognizes the answer corresponding to the query as in [59]. In transaction-based applications, e.g. mobile sensing, the use of this technique can lead to the collection of large amounts of noisy data thus compromising the application utility.

– **Location suppression.** This transformation simply withdraws the service request. We represent the transformation as follows:

$$T(loc) = \perp$$

where $\perp$ stands for *undefined*. Naturally this technique can be only applied sporadically, otherwise the service would have little sense. Location suppression is often used in conjunction with other privacy mechanisms, in particular location obfuscation and perturbation, such as in [1, 26, 29]. Location suppression is also at the basis of the mix-zones privacy model [7]: when an user enters a mix-zone area, the communication with the provider is suspended for the time the user is inside the area; at the exit the user is assigned a new pseudonym and the communication reactivated. Moreover, suppression is the only technique, we are aware of, employed for the protection of the location in geo-location services [19].

– **Cryptographic techniques.** The information flow relies on a cryptographic protocol. As a result the controller has no access to the user's plain data, i.e. the actual content of the request as well as the actual answer, if any. Cryptographic protocols have been originally defined for LBS, such as [27], where the approach is tailored to the classes of sporadic LBS supporting *k-nearest neighbor* queries. Protocols have been also specified in the context of location sharing applications, such as proximity-based services e.g. [38] and transaction-based applications. An approach combining cryptographic techniques with perturbation methods in transaction-based applications has been recently developed by Brown et al. [8]. The problem addressed in this work is to collect traffic data (e.g. vehicles speed) through crowdsourcing in a privacy preserving manner. Traffic data are first collected using a cryptographic voting protocol, then data are privately aggregated using perturbation methods based on differentially private techniques.

5.3 Architecture

The third dimension of the privacy mechanism is the architecture. We recall that the location transformation has to be necessarily performed by a trusted component. This can be achieved in two main ways:

(i) The transformation is performed locally by the mobile device
(ii) The transformation is performed by another *trusted entity*

Here, the term *entity* is deliberately generic and stands for anything but the single user. For example, the trusted entity can be a centralized third-party acting as proxy, i.e. it intercepts the users' request, applies the transformation and forwards the transformed request to the provider. This architecture is quite popular (see for example [30, 39]). Alternatively, the trusted entity can be a group of users offering their collaboration to enable the computation of the transformed location such as in [12, 28]. At a more abstract level, the choice of which of the two general architectures is more appropriate - client-based or trusted-entity based - depends on whether the transformation function needs to know the location of other users. For example the trusted-entity approach is extensively used for anonymity protection whereas the user's location needs to be confused with the location of nearby users. The client-centric approach is more popular in those applications in which the privacy goal is location or behavior protection, typically LBS such as in [10, 17], and mobile sensing such as in [1, 13].

5.4 Privacy mechanisms for multi-party applications

So far we have seen the privacy mechanisms for protecting the single information flow. Now let us go a step further and consider the general case in which there are $n > 1$ adversaries (i.e. the application model is multi-party), namely, for using the application the user has to disclose his location to n providers.

A straightforward approach to privacy protection is to define a privacy mechanism for each of the information flows taken singularly. This means to define a set of transformations, e.g. $\{T_1, .., T_n\}$ one for each adversary. The drawback of this approach is that it falls short when the adversaries collude. In what follows we briefly describe a motivating scenario and a first attempt to deal with such a problem in a specific two-party application context [19].

5.4.1 The two-party scenario

We refer to the application scenario in Section 3.3. In such a scenario there are two providers: one, say a_1, computes the user's location, and the other, a_2 provides an LBS. We recall the work-flow: first the user requests the location to a_1, next such information is used to request a service to a_2. There are thus two information flows. Now assume that both a_1 and a_2 are the same organization.[3] This scenario presents two novel challenges: (a) how to protect the location from the geo-location provider; and (b) how to protect the location from the two colluding parties.

The approach in [19] combines two privacy mechanisms: (a) the first mechanism is location suppression, in order to minimize the information flow between the user and a_1. In

[3]The scenario is realistic. For example one of the pioneering location sharing applications, i.e. Google Latitude, was coupled with the geo-location services provided by Google itself

particular the location is suppressed when the user is recognized to be inside a *private place*. A private place is for example *home*. (b) The second mechanism is in reality based on soft privacy i.e. the user can specify the location transformation to apply when specific contextual conditions are met such as the presence in a private place. A major novelty of this approach is that it attempts to bridge hard and soft privacy techniques. The implications of such a choice are still to be analyzed in depth.

6 Privacy metric

Eventually, we turn to consider the notion of privacy metric. A *privacy metric* is a formal criteria for the quantification of the degree of protection provided by a privacy mechanism with respect to a privacy goal. Privacy metrics provide the basis for evaluating and comparing privacy models as well as for tailoring such models to the privacy demands of users. Intuitively, a privacy metric measures how difficult it is for the adversary to guess the actual data from the transformed data [47]. This means that the level of privacy depends on what an adversary knows, or, put differently, how smart the adversary is. The capabilities of the adversary are specified in the *adversary model* which describes: a) the auxiliary knowledge of the adversary (background knowledge); b) the inference capabilities [47].

In this section we overview major classes of metrics. In particular we introduce three classification criteria that we call: subject, usage, and paradigm, explained in the following:

(i) Subject. The subject specifies whether the metric refers to the single user or, conversely, a group of users. The subject thus specifies the social granularity of the privacy measurement. To distinguish the two cases, we say that the metric is *individual* and *collective*, respectively.

(ii) Usage. The usage specifies at what stage of the application the metric is used and for which purpose. We consider the following two cases:

- The metric is used at the end of an observation period, to assess the level of privacy provided to the subject by the privacy mechanism in such a time frame. We say that the metric is *ex-post*.
- The metric is used before the privacy mechanism is applied, to tailor the transformation to the required degree of protection. We say that the metric is *ex-ante*.

(iii) Paradigm. Privacy metrics typically rely on paradigms such: k-anonymity, entropy-based metrics, error-based metrics, metrics based on probabilistic models, and differential privacy.

We now discuss in more detail, representative metrics for each of the above paradigms. Where possible we emphasize the distinction between collective and individual metric, specifying as well whether the metric is ex-ante or ex-post. We assume a unique information flow.

6.1 Location *k*-anonymity

Grounded on the well known metric defined in the area of data publishing [51], *location* k-*anonymity* measures the degree of user anonymity where the information flow is sporadic e.g. [11, 30]. The adversary can re-identify seemingly anonymous users by linking their location with some external source referring to the same location and revealing the user's identity. Location k-anonymity is naturally associated with a location obfuscation mechanism. An obfuscated region r is k-anonymous if r is indistinguishable from the location of other k-1 users in the vicinity. Therefore the probability for a user to be re-identified based on location is $\frac{1}{k}$. The value of k is used to generate the obfuscated region, therefore the metric is used ex-ante. In addition, the value of k can be either unique for the whole set of users or be personalized as in [23]. Therefore the metric can be either collective or individual.

The condition of location indistinguishability, however, only serves to protect the association between users and queries, while it does not prevent the disclosure of the association between users and locations. For example if all the k users are located in a small region, e.g. a bar, location is easily revealed. Therefore, in spite of its popularity, location k-anonymity does not ensure a real protection of location [16, 49]. For this, another metric - location l-diversity - has been defined.

L-diversity supplements k-anonymity to prevent the identification of the semantic locations, i.e. places, in which the user can be located within an obfuscated region. A first definition of location l-diversity is given in [5]: a region is l-diverse when it contains l different physical locations where a physical location is identified by a symbolic address. A different definition, which accounts for the semantics of location, is due to [57]; a region r is l-diverse if the semantic locations contained in r are of at least l different types. Note that location k-anonymity and its variants assume a simple adversary model where the adversary has limited knowledge of the geographical context.

6.2 Entropy-based metrics

Entropy is a measure of the average degree of uncertainty associated with a set of events. Formally, the entropy of a set of N events is:

$$h = -\sum_{1}^{N} p_i \log p_i$$

where p_i is the probability of occurrence of event i. In the following we describe two representative entropy-based metrics. The former has been defined for evaluating the level of privacy offered by the mix-zones privacy mechanism [7]; the latter, the level of privacy in feeling based techniques.

Mix-zones We briefly recall the mix-zones privacy mechanism, described for example in [6, 43]. A mix-zone is a region of the reference space. When inside the mix-zone, the user is assigned a new pseudonym that identifies the user until the next mix-zone is entered. For the whole time the user is inside a mix-zone, no location information is disclosed (i.e the location is suppressed). The goal is to prevent the tracking of the user's long-term movement, i.e. should a pseudonym be violated, only the movement corresponding to the trace associated with that pseudonym would be revealed. An example of mix zone is reported in Fig. 4.

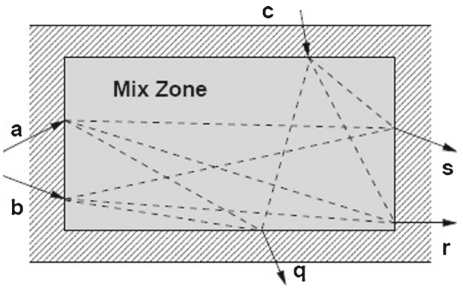

Fig. 4 A mix zone: a, b, c are the entering pseudonyms; q, r, s, are the newly generated pseudonyms [7]

Entropy comes into play in order to estimate the uncertainty of the correlation between old and newly generated pseudonyms [7]. Consider N users entering mix zone X with pseudonym $P = \{p_1, p_2, ..p_n\}$ and assume that N users exit from X after time δt with pseudonyms $Q = \{q_1, q_2, ..., q_n\}$. Entropy measures the uncertainty of the association between the elements of P and Q over δt. Formally, given a set $M = m_1, ..m_k$ of possible mappings between the elements of P and Q, the entropy is given by:

$$h = -\sum_i Pr(m_i|M) \log Pr(m_i|M)$$

Note that the entropy measure depends on the number of users entering the mix-zone and that such number is only known at run time. Therefore the privacy metric, defined in these terms, can be only utilized ex-post, as collective metric.

Feeling-based location privacy The privacy goal is to prevent users from being re-identified based on their presence (footprint) in a restricted region, such as home or office, where individuals can be easily identified. To that end, the privacy mechanism replaces the actual location with an obfuscated region whose size depends on the level of popularity specified by the user. The higher the level of popularity, the higher the level of privacy. Entropy serves to quantify the popularity of a region. Intuitively the popularity of a region depends on the number of individuals leaving their footprint in the region. More formally, given a region R and a set of k users with footprints in R, the entropy of R is

$$E(R) = -\sum_1^k \frac{n_i}{N} \log \frac{n_i}{N}$$

where n_i is the number of footprints of user-i in R and $N = \sum_1^k n_i$. The popularity $Pop(R)$ of region R is expressed as: $2^{E(R)}$. Users willing to personalize their privacy can specify the desired entropy of the obfuscated region by indicating a public region with similar popularity, e.g. a shopping mall. The obfuscation algorithm then generates a region which contains the user's current position, has the desired popularity and is as small as possible. This mechanism can be applied to both continuous and sporadic LBS. It can be observed, that unlike the previous case, the feeling-based privacy metrics is defined at individual level and used ex-ante.

6.3 Error-based metrics

Location privacy is measured in terms of distance between the transformed location, or a location derived from it, and the actual location. In the following we describe two

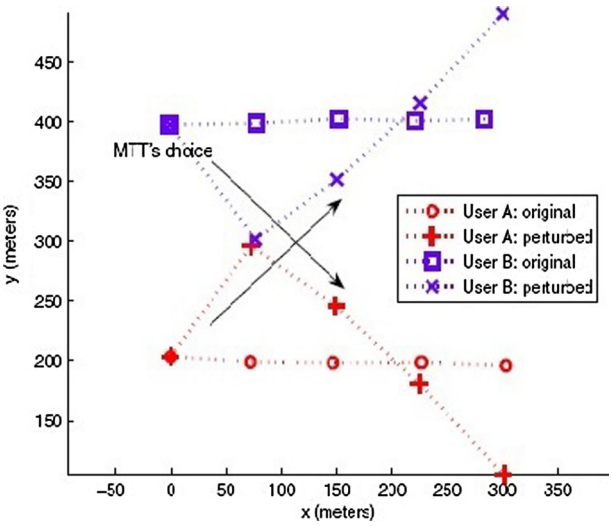

Fig. 5 Mixing the traces of two users [31]

representative metrics: the *expectation of distance error* defined for the path confusion privacy mechanism [31]; and the *relevance*, the metric defined for the location obfuscation mechanism in [3].

Expectation of distance error The expectation of distance error captures how accurately an adversary can estimate a user's position. This metric has been defined for the path confusion privacy mechanism. The objective of such a mechanism is to prevent long term user's tracking. The adversary has the capability of predicting with high confidence the user's next location based on the movement history, for example using the Multi Target Tracking (MTT) algorithm.

The approach is to confuse the pseudonyms: every time two users' traces meet (i.e. users are in close proximity) the location is perturbed so as to increase the chance for the adversary to confuse the tracks and follow the wrong user. An example is reported in Fig. 5. The figure shows two users moving in parallel. The algorithm perturbs the parallel segment into a crossing segment, while the MTT algorithm (the adversary) gets confused and follows the wrong user.

Given N users and an observation time of M timestamps, the expectation of distance error for the trajectory of user u is defined as follows [25]:

$$E(u) = \frac{1}{NM} \sum_{i=1}^{M} \sum_{j=1}^{I_i} p_j(i) d_j(i)$$

where I_i is the total number of assignment hypotheses for user u at timestamp i, $p_j(i)$ is the probability associated with hypothesis j at timestamp i, and $d_j(i)$ is the distance between the actual and estimated position of u at timestamp i.

Relevance The second metric, called *relevance*, is an individual metric. The privacy mechanism is based on location obfuscation. No assumption is made on the adversary's

knowledge. This metric takes into account the intrinsic uncertainty of location measurements. In particular, a location *loc* is defined by a circle of origin x, y and radius r, while the relevance of the location, denoted R_{loc}, is the ratio:

$$R_{loc} = \frac{r_0^2}{r^2}$$

where r_0 is the radius of the area that would be returned if the positioning system would estimate the location with optimal accuracy. R thus specifies the relative accuracy loss of the location with respect to the accuracy provided by the positioning technology. Privacy is given by the difference $1 - R$. The privacy mechanism provides a set of operators for the generation of obfuscated locations. For example, such operators can change the radius or shift the origin. The obfuscation operators can then be composed using certain criteria to obtain obfuscated locations having the desired relevance. The metrics is thus used ex-ante. Note that, unlike the previous metrics, the notion of relevance has exclusively a geometric meaning, i.e. it does not take into account what the adversary may know and infer. Moreover there is no distinction between the notion of privacy and that of location quality.

6.4 Probabilistic metrics

Probabilistic metrics describe the adversary in probabilistic terms. The adversary leverages both the background knowledge and the observed transformed locations, to infer the true user's location. The background knowledge is modelled in terms of random variables, the inference capabilities as statistical inferences. Two representative metrics are described below: the first is called *correctness-based* and is defined within a formal framework for location protection; the second, called *location sensitivity* targets behavior protection. Both these metrics are individual metrics and are used ex-ante.

Correctness-based metric This metric is used in [48] for sporadic LBS. The adversary observes the transformed location and, knowing the privacy mechanism (i.e. location perturbation) and the users' *access profile* (i.e. the background knowledge), tries to infer some information of interest about the actual locations. The privacy metric measures how close the location estimated by the adversary is from the actual location.

The concepts of access profile and privacy mechanism are formally defined as follows. The *access profile* of user u, $\psi_u^\tau(r)$ specifies the probability that user u is in location r in a gridded space in time period τ, for any location and time period. The user's profile is thus time dependent. The *privacy mechanism* is a function that transforms location r at time t into location r' by sampling from the following probability distribution:

$$f(r'|r) = Pr\{o(t) = (r', t)|a(t) = (r, t)\}$$

where $o(t)$ is the observed location and $a(t)$ the actual location.

The other two key concepts are those of attack and privacy metric. The adversary runs an *attack* to infer the actual location $a(t)$ from the observed location $o(t)$. Formally the result of an attack can be described as posterior probability distribution $h(.)$ such that:

$$h(\hat{r}|r') = Pr\{o(t) = (\hat{r}, t)|a(t) = (r, t)\}$$

Note that $h(\hat{r}|r')$ is an estimate of the actual posterior probability $h(r|r')$, because the adversary has finite resources. The expected distance between the actual location and the one

estimated by the adversary defines the *privacy metric*. Formally the expectation over all location r, r', and $\hat{r}$ is defined as follows:

$$Privacy(\psi, f, h, d_p) = \sum_{r,\hat{r},r'} \psi(r), f(r'|r), h(\hat{r}|r') d_p(\hat{r}, r)$$

where d_p is the distance function, e.g. the Euclidean distance.

The metric has been used to find the privacy mechanism, namely the function $f(.)$ that maximizes the user's location privacy, given a maximum tolerable service-quality loss specified through the distance function $d_p(.)$. This problem is formulated as instance of the zero-sum Bayesian Stackelberg game. The metric is used in the Stackelberg game to find the optimal privacy mechanism for each user. This privacy framework represents an important contribution in the direction of a sound theory of location privacy. However, it raises a number of issues, first of all the computational cost and the practical deployment of the solution in real applications.

Location sensitivity metric This metric has been defined for a *selective* location obfuscation mechanism [20]. The privacy goal is to prevent the extraction of the following mobility pattern, i.e the user's presence in a set of regions termed *sensitive*. For example an user may wish to share accurate locations with friends everywhere but in hospitals. The reference space is a grid space $\{c_1, ..c_n\}$ of n cells. The space contains places of types $\{lt_1, ...lt_k\}$, where a place has a spatial extent. A subset of those types denotes sensitive places. The adversary has background knowledge and inference capabilities. Moreover the user's location is disclosed sporadically.

The background knowledge consists of two components: (a) the prior knowledge of the distribution of the user's locations, expressed by the function D_u, e.g. $D_u(c_i)$ is the probability of finding the user in cell c_i in the time interval the distribution refers to; (b) the geographical context, i.e. the semantic locations. The obfuscation mechanism is represented by a function $F_u(.)$ that, given the actual location c_i returns either c_i or a region r obfuscating c_i. Note that the obfuscation is *selective*, i.e. it is not indifferently applied to every location.

The adversary tries to infer whether the user located in an obfuscated location is in a sensitive place based on the background knowledge and knowing the obfuscation mechanism. The *location sensitivity* of an arbitrary region r with respect to place type lt_i, denoted as $Sens(r, lt)$, is defined as posterior probability that a random user in r is in any place of sensitive type lt_i. In the discrete space such value can be expressed as ratio of the probability that the user is inside a cell of type lt and the probability that the location is in r; if r is unreachable for the user, for example for some physical barrier, the sensitivity is 0.

$$Sens(r, lt) = \begin{cases} \frac{\sum_{c_i \in r} S^{lt}(c_i) D_u(c_i)}{\sum_{c_i \in r} D_u(c_i)} & \text{if } \sum_{c_i \in r} D_u(c_i) \neq 0 \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

This metric is used for the generation of the obfuscated locations. In order to prevent possible inferences on the correlation between the actual location and the obfuscated location, the obfuscation process is divided in two phases, off-line and on-line, respectively. The off-line process generates a set of obfuscated locations $\{r_1, ..r_k\}$ covering the set of sensitive places and satisfying the constraint that the sensitivity of those regions must not exceed a threshold value, i.e. $Sens(r_i, lt_j) < \tau$ while minimizing the loss of spatial accuracy. The on-line process checks whether the user is located in any obfuscated region r_i. If so, r_i is disclosed otherwise the actual location is not obfuscated.

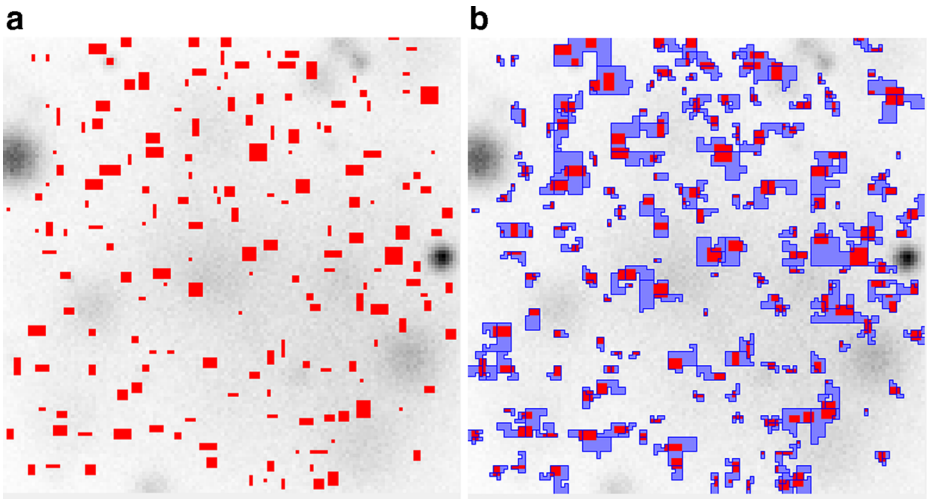

Fig. 6 **a** The *rectangles* are the sensitive regions; the *gray* background indicates the probability distribution; **b** The *blue* polygons are the obfuscated regions, obtained from the $Sens_{hilb}$ algorithm

A number of heuristics have been developed for the generation of obfuscated regions [20]. Figure 6 illustrates the outcome of $Sens_{hilb}$ obfuscation algorithm. The algorithm generates obfuscated regions by progressively aggregating the cells in the neighborhood of sensitive cells in the linear space represented by the Hilbert filling curve.

6.5 Differential location privacy

Originally developed in the area of statistical databases, differential privacy is being increasingly deployed in a variety of domains to support privacy-preserving data analysis, including social data mining and spatial indexing. Interestingly, a recent stream of research attempts to bridge differential privacy and location privacy. Before proceeding, we briefly overview key concepts of differential privacy.

Consider a database containing sensitive data about a sample set of individuals. The goal is to prevent analysts to inferring single individual's information from aggregated data obtained by querying the database. An algorithm M that takes as inputs the database and a query, and returns a noisy response is *differentially private* if for any pair of databases X and X' with $|X| = |X'|$ differing for only one record, and for any query q , it holds that the probability of obtaining result r from database X differs from the probability of obtaining r from database X' of at most a predefined amount. Formally, let $Pr(M(X, q)) = r$ the probability of obtaining answer r from database X upon query q and using M. It must hold that:

$$\frac{Pr[M(X,q) = r]}{Pr[M(X',q) = r]} < e^{\epsilon}$$

Recent research has started investigating relaxed models of differential privacy. For example, [9] departs from the database-centric view of differential privacy to propose a

framework for the protection of location privacy in LBS. The goal is to define a principled approach to location perturbation. In essence the idea is to perturb a location x, by generating a random noise, in such a way that the probability of reporting x' in place of x differ at most by a multiplicative factor $e^{-\epsilon d(x,x')}$ where $d(x, x')$ is the Euclidean distance. Whether this form of relaxation of differential privacy is really effective is an open issue.

7 Conclusions and research directions

In this work, we propose a conceptual framework for the classification of hard privacy techniques along three dimensions, i.e. privacy goal, privacy mechanism and privacy metric, across different classes of location-aware applications.

Defining a conceptual framework for location privacy is a complex task. The main problem stems from the fact that mobile applications, legal frameworks, privacy theories in social sciences, are all interrelated, and all of them are rapidly evolving. This motivates the need to define a core of general and shared concepts. On the other hand, the on-going evolution opens up new research opportunities. In what follows we elaborate on three broad research directions.

The first research direction is towards the definition of formal and general frameworks centered on the notion of privacy metric enabling the evaluation and comparison of hard privacy solutions. Probabilistic privacy models may represent an interesting direction. A complementary effort that should accompany the development of formal theories regards privacy usability, i.e. how to make privacy techniques usable in real applications. The definition of such a framework can facilitate the cooperation among different research communities and the sharing of results in a wider and socially relevant community.

The second direction is towards real and more ambitious privacy goals, e.g. to protect behavior patterns in real time and with incomplete knowledge of the user's movement. The world of analytics is, however, an open world, where novel techniques are continuously discovered. Therefore one could legitimately wonder whether there can exist an equilibrium point between the competing needs of business organizations, willing to capitalize on the enormous amount of personal data they possess, and the individuals right to privacy. The problem cannot be solved uniquely through the use of technology.

The third research challenge is related to the development of engineering principles supporting privacy requirement analysis and solutions development. The principles of *privacy-by-design* are likely too generic and far from real and practical problems. The approach we have proposed for the modeling of location-aware applications, based on the notion of information flow, is a first, small contribution in the direction of location privacy engineering.

References

1. Agir B, Papaioannou TG, Narendula R, Aberer K, Hubaux JP (2013) User-side adaptive protection of location privacy in participatory sensing. Geoinformatica, to appear
2. Andrienko G, Gkoulalas-Divanis A, Gruteser M, Kopp C, Liebig T, Rechert K (2013) Report from Dagstuhl: the liberation of mobile location data and its implications for privacy research. ACM SIGMOBILE Mob Comput Commun Rev 17(2):7–18
3. Ardagna CA, Cremonini M, Damiani E, di Vimercati S, Samarati P (2007) Location privacy protection through obfuscation-based techniques. In: 21st annual IFIP WG 11.3 working conference on data and applications security

4. Article_29_Data_Protection_Working_Party (2011) WP185 Opinion 13/2011 on Geolocation services on smart mobile devices. http://ec.europa.eu/justice/data-protection/article-29/documentation/opinion-recommendation/files/2011/wp185_en.pdf. Accessed Feb 2014
5. Bamba B, Liu L, Pesti P, Wang T (2008) Supporting anonymous location queries in mobile environments with PrivacyGrid. In: Proceedings of WWW
6. Beresford AR, Stajano F (2003) Location privacy in pervasive computing. IEEE Pervasive Comput 2(1):46–55
7. Beresford AR, Stajano F (2004) Mix zones: user privacy in location-aware services. In: Proceedings of the 2nd IEEE annual conference on pervasive computing and communications workshops
8. Brown J, Ohrimenko O, Tamassia R (2013) Haze: privacy-preserving real-time traffic statistics. arXiv:1309.3515v1 [cs.CR] 13
9. Chatzikokolakis K, Andrés ME, Bordenabe NE, Palamidessi C (2013) Broadening the scope of differential privacy using metrics. In: Symposium HotPets 2013. OnLine version: http://freehaven.net/anonbib/papers/pets2013/paper_57.pdf
10. Cheng R, Zhang Y, Bertino E, Prabhakar S (2006) Preserving user location privacy in mobile data management infrastructures. In: Proceedings of the 6th workshop on privacy enhancing technologies
11. Chow C, Mokbel MF, Aref WG (2009) Casper*: query processing for location services without compromising privacy. ACM Trans Database Syst (34)4
12. Chow CY, Mokbel MF, Liu X (2006) A peer-to-peer spatial cloaking algorithm for anonymous location-based service. In: Proceedings of ACM GIS
13. Cornelius C, Kapadia A, Kotz D, Peebles D, Shin M, Triandopoulos N (2008) Anonysense: privacy-aware people-centric sensing. In: Proceedings of ACM MobiSys
14. Damiani ML (2011) Third party geolocation services in LBS: privacy requirements and research issues. Trans Data Priv 4(2):55–72
15. Damiani ML (2013) European data protection: coming of age? In: Privacy enhancing techniques for the protection of mobility patterns in LBS: research issues and trends. Springer
16. Damiani ML, Bertino E, Silvestri C (2008) Protecting location privacy through semantics-aware obfuscation techniques. In: Proceedings of IFIPTM
17. Damiani ML, Bertino E, Silvestri C (2010) The PROBE framework for the personalized cloaking of private locations. Trans Data Priv 3(2):123–148
18. Damiani ML, Cuijpers C (2012) Privacy-aware geolocation interfaces for volunteered geography: a case study. In: Proceedings of ACM GEOCROWD
19. Damiani ML, Galbiati M (2012) Handling user-defined private contexts for location privacy in LBS. In: Proceedings of ACM GIS
20. Damiani ML, Silvestri C, Bertino E (2011) Fine-grained cloaking of sensitive positions in location-sharing applications. IEEE Pervasive Comput 10(4):64–72
21. Deng M, Wuyts K, Scandariato R, Preneel B, Joosen W (2011) A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements. J Requir Eng Spec Issue Digit Priv Theory Policies Technol 16(1):3–32
22. Duckham M, Kulik L (2006) Location privacy and location aware computing. In: Drummond J (ed) Dynamic & mobile GIS: investigating change in space and time. CRC Press, Boca Rator
23. Gedik B, Liu L (2005) Location privacy in mobile systems: a personalized anonymization model. In: Proceedings of ICDCS
24. GEOPRIV. http://www.ietf.org/html.charters/geopriv-charter.html
25. Ghinita G (2013) Privacy for location-based services. Morgan & Claypool Publishers, San Rafael
26. Ghinita G, Damiani ML, Silvestri C, Bertino E (2009) Preventing velocity-based linkage attacks in location-aware applications. In: Proceedings of ACM GIS
27. Ghinita G, Kalnis P, Khoshgozaran A, Shahabi C, Tan K-L (2008) Private queries in location based services: anonymizers are not necessary. In: Proceedings of ACM SIGMOD
28. Ghinita G, Kalnis P, Skiadopoulos S (2007) PRIVE: anonymous location-based queries in distributed mobile systems. In: Proceedings of WWW
29. Goetz M, Nath S, Gehrke J (2012) MASKIT: privately releasing user context streams for personalized mobile applications. In: Proceedings of ACM SIGMOD
30. Gruteser M, Grunwald D (2003) Anonymous usage of location-based services through spatial and temporal cloaking. In: Proceedings of ACM Mobysis
31. Hoh B, Gruteser M (2005) Protecting location privacy through path confusion. In: Proceedings of international conference on security and privacy for emerging areas in communications networks
32. Jensen CS, Lu H, Yiu ML (2009) Location privacy techniques in client-server architectures. In: Privacy in location-based applications: research issues and emerging trends. Springer

33. Kido H, Yanagisawa T, Satoh Y (2005) Protection of location privacy using dummies for location-based services. In: Proceedings of ICDEW
34. LaMarca A, de Lara E (2008) Location systems. Morgan and Claypool Publishers, San Rafael
35. Lane ND, Miluzzo E, Lu H, Peebles D, Choudhury T, Campbell AT (2010) A survey of mobile phone sensing. IEEE Commun Mag 48(9):140–150
36. Lee B, Oh J, Yu J, Kim H (2011) Protecting location privacy using location semantics. In: Proceedings of ACM SIGKDD
37. Li N, Chen G (2010) Sharing location in online social networks. IEEE Netw 24(5):20–25
38. Mascetti S, Freni D, Bettini C, Wang XS, Jajodia S (2011) Privacy in geo-social networks: proximity notification with untrusted service providers and curious buddies. VLDB J 20(4):541–566
39. Mokbel MF, Chow WG, Aref C-Y (2006) The new Casper: query processing for location services without compromising privacy. In: Proceedings of VLDB, pp 763–774
40. Mouratidis K, Yiu ML (2010) Anonymous query processing in road networks. IEEE Trans Knowl Data Eng 22(1):2–15
41. Myles G, Friday A, Davies N (2003) Preserving privacy in environments with location-based applications. IEEE Pervasive Comput 2:56–64
42. Nissembaum H (2011) A contextual approach to privacy online. Dedalus, J Am Acad Arts Sci 140(4):32–48
43. Palanisamy B, Liu L (2011) Mobimix: protecting location privacy with mix-zones over road networks. In: Proceedings of IEEE ICDE
44. Parent C, Spaccapietra S, Renso C, Andrienko G, Andrienko N, Bogorny V, Damiani ML, Gkoulalas-Divanis A, Macedo J, Pelekis N, Theodoridis Y, Yan Z (2013) Semantic trajectories modeling and analysis. ACM Comput Surv 45(4):42:1–42:32
45. Rechert K, Meier K, Zahoransky R, Wehrle D, von Suchodoletz D, Greschbach B, Wohlgemuth S, Echizen I (2013) Reclaiming location privacy in mobile telephony networks—effects and consequences for providers and subscribers. IEEE Syst J 7(2):211–222
46. Ruiz-Vicente C, Freni D, Bettini C, Jensen CS (2011) Location-related privacy in geo-social networks. IEEE Internet Comput 15:20–27
47. Shokri R, Theodorakopoulos G, Le Boudec JY, Hubaux JP (2011) Quantifying location privacy. In: IEEE symposium on security and privacy
48. Shokri R, Theodorakopoulos G, Troncoso C, Hubaux JP, Le Boudec JY (2012) Protecting location privacy: optimal strategy against localization attacks. In: Proceedings of CCS
49. Shokri R, Troncoso C, Diaz C, Freudiger J, Hubaux JP (2010) Unraveling an old cloak: k-anonymity for location privacy. In: Proceedings of WPES
50. Solove D (2013) Privacy self-management and the consent dilemma. Harv Law Rev 123:1880–1902
51. Sweeney L (2002) Achieving k-anonymity privacy protection using generalization and suppression. Int J Uncertain Fuzziness Knowl-Based Syst 10:571–588
52. Tavani HT, Moor JH (2001) Privacy protection, control of information, and privacy-enhancing technologies. ACM SIGCAS Comput Soc 31(1):6–11
53. Toch E, Ravichandran R, Cranor LF, Drielsma PH, Hong J, Kelley PG, Sadeh N, Tsai JY (2009) Analyzing use of privacy policy attributes in a location sharing application. In: Proceedings of symposium on usable privacy and security (SOUP)
54. Vicente CR, Assent I, Jensen CS (2011) Effective privacy-preserving online route planning. In: Proceedings of MDM
55. W3C (2012) Geolocation API specification. http://dev.w3.org/geo/api/spec-source.html
56. Westin A (1970) Privacy and freedom. Bodley Head
57. Xue M, Kalnis P, Pung HK (2009) Location diversity: enhanced privacy protection in location based services. In: Proceedings of international symposium on location and context awareness (LoCA)
58. Yigitoglu E, Damiani ML, Abul O, Silvestri C (2012) Privacy-preserving sharing of sensitive semantic locations under road-network constraints. In: IEEE MDM
59. Yiu ML, Jensen CS, Huang X, Lu H (2008) SpaceTwist: managing the trade-offs among location privacy, query performance, and query accuracy in mobile services. In: Proceedings of ICDE

Maria Luisa Damiani received the PhD in Computer Science from EPFL (CH) and the MSc in Computer Science from the University of Pisa (I). Main research interests are location-based security, location privacy, and spatio-temporal modelling, in particular semantic and symbolic trajectories. Before joining the University of Milan, Italy, she worked for several years in public and private organizations, first as researcher, then as R&D Coordinator, and finally as co-founder of a start-up company operating in the area of geospatial data management in road and traffic applications. She has been coorganizer of the ACM Springl Workshop series on Security and Privacy in Location Based Services and GIS from 2008 to 2011 and of the IEEE Workshop on Privacy and Security of Moving Objects, 2013. She has been recipient of the Best Paper (Runner up) award at ACM GIS 2009 and Best Demonstration Award at ACM GIS 2012 and 3 DASFAA 2013.