Privacy preservation in the dissemination of location data

Manolis Terrovitis
Institute for the Management of Information Systems (IMIS)
Research Center "Athena"
Athens, Greece
mter@imis.athena-innovation.gr

ABSTRACT

The rapid advance in handheld communication devices and the appearance of smartphones has allowed users to connect to the Internet and surf on the WWW while they are moving around the city or traveling. *Location based services* have been developed to deliver content that is adjusted to the current user location. Social networks have also responded to the challenge of users who can access the Internet from any place in the city, and *location based social-networks* like Foursquare have become very popular in a short period of time. The popularity of these applications is linked to the significant advantages they offer: users can exploit live location-based information to take dynamic decisions on issues like transportation, identification of places of interest or even on the opportunity to meet a friend or an associate in nearby locations. A side effect of sharing location-based information is that it exposes the user to substantial privacy related threats. Revealing the user's location carelessly can prove to be embarrassing, harmful professionally, or even dangerous.

Research in the data management field has put significant effort on anonymization techniques that obfuscate spatial information in order to hide the identity of the user or her exact location. Privacy guaranties and anonymization algorithms become increasingly sophisticated offering better and more efficient protection in data publishing and data exchange. Still, it is not clear yet what are the greatest dangers to user privacy and which are the most realistic privacy breaching scenarios. The aim of the paper is to provide a brief survey of the attack scenarios, the privacy guaranties and the data transformations employed to protect user privacy in *real time*. The paper focuses mostly on providing an overview of the privacy models that are investigated in literature and less on the algorithms and their scaling capabilities. The models and the attack scenarios are classified and compared, in order to provide an overview of the cases that are covered by existing research.

1. INTRODUCTION

The recent explosive growth in the usage of smart phones and the increased availability of wireless and GSM connection has allowed users to connect to the Internet, use web services or other types of custom services, send and receive data from any place and at any time. Moreover, the developments in positioning technologies like GPS, wireless positioning, GSM etc has allowed for tracing users' location, storing it, processing and even easily rendering it on a map. Collecting and processing location based information has significant benefits to users who can get information adjusted to their current location, to service providers who can better understand user behavior and offer them customized location aware services(usually termed as *Location Based Services* (*LBS*)) and finally to researchers and public authorities who can exploit user movement data in order to plan better the future development of services and infrastructure.

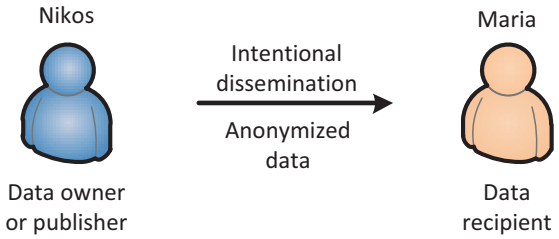

Figure 1: Basic *PPDD* scenario

The monitoring, processing and storing of users' locations in an unprecedented scale has naturally attracted attention to privacy related issues. Location information can reveal sensitive information about users, like health related issues, commercial practices, sexual preferences and can cause embarrassment, financial damage or even expose users to physical dangers. In the data management and knowledge discovery field the focus of research is on privacy preservation in the publication and exchange of data. Numerous privacy preserving techniques have been proposed that allow publishing and exchanging data without breaching the privacy of the users. In the last few years many works have proposed methods that protect the privacy of users in scenarios where location based information is involved. The majority of works has focused on protecting the privacy of users when they are communicating with *LBS* providers that are not trusted. These techniques provide protection to the user in *real time* against location disclosure (they hide the exact location of the user from the untrusted entities in the communication scenario), against identity disclosure (they hide the identity of the user from adversaries that already know other identifying information, like the exact location) and against disclosure of sensitive information (they stop the adversary from inferring that a user visited a certain place or made use of a sensitive service).

Despite the concern about user privacy both in research and in practice, a vast number of users choose to share their location with *LBS* providers and friends in the rapidly growing field of location based social networks. At the same time high quality maps and semantic information on them is freely available in the Internet (e.g., Google Maps). The locations of hospitals, schools, companies is usually marked on the map and even the location of unexpected events becomes very soon available on-line. All these freely available data do not only allow malicious adversaries to collect information about user movement in the city, but also the semantically rich map background allows them to infer her work place, her home address, even information not directly associated with location like religious and sexual preferences. Possibly dangerous attacks are quite easy to perform. To raise awareness over the privacy problems of sharing location information in *LBSN*s, *PleaseRobMe* (`http://pleaserobme.com/`) demonstrated how it can easily infer that someone is out of home and his apartment is empty. The attack was possible by combining data from Foursquare (`http://4sq.com`) and Twitter (`http://twitter.com`).

There are several research works which show that people do not put great value on their privacy, and in general they are willing to share their location data [20; 3; 7; 14]. On the other hand, privacy is something that is valuable when missed. Since sharing location data on this scale is something relatively new, the effects on user privacy are not fully understood yet. Adversaries might appear in the future that will be able to collect data from past years. Even now, malicious parties might be collecting data that can be used in the future. The scale of the threat on user privacy is something that will be understood in long run, but given that user data are made available now, it might be too late.

The aim of the paper is to provide an overview on the privacy breaching scenarios that have been studied in data management and knowledge discovery research literature. The paper explores what are the usual assumptions about the adversaries background knowledge, what are the communication architectures that make an attack scenario possible and what information is the target of the attack in each case. The privacy guaranties of the different anonymization methods are categorized and the basic data transformations and the information loss they introduce are explored. The focus in this work is to present a broad picture of the base assumptions,the dangers and the privacy guaranties that have been devised to address them, and less emphasis is given to algorithmic issues. In this way, it is aimed to provide the background for assessing whether the privacy problems studied in data management literature are the same with those that are or will be encountered in practice.

2. SCOPE OF THE SURVEY

Privacy protection has attracted significant interest in many research areas. This survey focuses on *privacy preserving data dissemination* (*PPDD*) in scenarios involving location based data. A basic scenario is depicted in Figure 1. The data owner, Nikos, wants to share some of his data with Maria, without revealing sensitive information. In the case of Figure 1 Nikos wants to inform Maria about his location in an abstract manner, without revealing his exact position. In such scenarios there is a common interest to both the data owner and the data recipient to share information. For example, the data owner might want to receive a service from the data recipient or the data owner might be someone who publishes data to promote a common cause, e.g. research. Good surveys of the rapidly growing *PPDD* research literature can be found in [15; 24; 42].

The predominant paradigm in *PPDD* when dealing with location data is that of private data sharing in the invocation of a location based service (*LBS*). Content delivered by location based services is adjusted to the location of the user that invokes them. For example, a user might transmit his location and ask an *LBS* provider for the nearest gas station. To protect user privacy, anonymization methods will transform the user request in such a way that certain information will be hidden from the provider. At the same time, the information that remains in the request should be adequate for the provider to deliver his services and should not introduce significant overheads to any of the two parties. There are many scenarios where the privacy of a user who invokes an *LBS* is at risk. Scenarios are differentiated based on several factors: a) on the definition of privacy, i.e., what information the user wants to hide, b) on the communication architecture, e.g. an intermediate trusted server might exist between Nikos and Maria, c) on the attack model, i.e., the background knowledge of the attacker, and the inference capabilities she has and d) on the data transformation, i.e, in what way are the original data transformed in order to provide the desired privacy guaranty. In Sections 4-6 I present several works that deal with *real-time PPDD* in the invocation of *LBS* .

Another important scenario of *PPDD* that involves spatial data is that of the off-line publishing of data collections [64; 39; 53; 22; 70; 5; 4; 52]. The difference from the previous scenario is that the data publisher does not want to share a single location or limited information describing his request, but a large collection of data that describe movement of users over significant time periods. In such scenarios, the anonymization procedure takes into account the whole dataset and can perform more complicated transformations, but at the same time it has to face attackers who can have significant background knowledge about user movement. Since the focus of these works are in the off-line publications, they are out of the scope of the paper and they are not presented here.

There is significant work on privacy protection that focuses on the data access control. In this context, policy models that enable users to easily define who and when can access their data, and models for the propagation of access rights are investigated. Several characteristic approaches for designing and enforcing access policies specific to location based data appear in [1; 36; 37; 54; 57]. Such methods are again outside the scope of our work and are not presented in the paper. We only present some methods based on cryptography since they do transform the data and in some cases they are employed for ensuring anonymity in the communication with *LBS* providers.

3. REAL-TIME PRIVACY PROTECTION

As location based services become more and more popular, users are allowing possibly unknown service providers to collect data about their movement. When a user invokes an *LBS* she usually sends her exact position to the provider, so that she can receive a response that is adjusted to her

location. In this usage scenario a malicious provider can collect accurate movement data about users. Following the collection of the data, the provider can process them and extract potentially sensitive information, e.g. the user's home address, work address, sexual and religious preferences etc. This is a technically feasible scenario that can have significant negative consequences to the user. To address such threats research in *PPDD* has proposed several anonymization methods, which guarantee that the *LBS* provider or any other recipient of the user's location-based request receives an anonymized message, where sensitive information is hidden.

This section briefly describes some of the basic concepts that appear in the anonymization methods for the real-time communication with *LBS* providers. I present the communication architectures that are most frequently adopted, the basic transformations that are used to anonymize the data and finally the attack models and the privacy guaranties that are offered by the different anonymization methods.

3.1 Background and Terminology

As in the case of relational data, anonymization techniques for location data assume that each record is partitioned to *quasi-identifiers* and to *sensitive values*. Quasi-identifiers are the possible background knowledge of the adversary. Adversaries can associate the quasi-identifiers with the true identity of a person, thus they can use them to identify the person's record in the published dataset. The association can be done through external public or private catalogs where the quasi-identifiers appear together with direct identifiers, e.g., voters' catalogs. Quasi-identifiers are not considered usually as harmful to the person they describe, and they can be safely disclosed. On the other hand, sensitive values are unknown to the attacker and they can cause harm to an individual if they are associated with her or him. In most attack scenarios there is a clear separation between quasi-identifiers and sensitive values, i.e., a value is either a quasi-identifier or sensitive, it can not be both. Still, there are several variations where sensitive values act as quasi-identifiers.

Privacy guaranties that protect against identity disclose, like k-anonymity [63], make a record indistinguishable from a group of records with respect to their quasi-identifiers. For example, they replace the quasi-identifiers of all records in each group with a common value, e.g, all different salaries are replaced with the average salary of the group. We term these groups with indistinguishable quasi-identifiers as *equivalence classes*. Creating equivalence classes is also the basic idea in privacy guaranties against attribute disclosure, like l-diversity [48]. The difference from the case of protection against identity is that the anonymization procedure imposes restrictions on the statistical distribution of sensitive values inside each equivalence class. For example, it limits the percentage of records that might contain a sensitive value.

Finally, in the case of location data the space of which users move is termed as *map* throughout the paper.

4. LBS ARCHITECTURES

The *PPDD* literature adopts several different communication architectures for the provision of *LBS* . The three basic ones are depicted in Figure 2: a)The users send their requests to a trusted server, termed *anonymizer*, who anonymizes them and then forwards them to the *LBS* provider, b) the users communicate directly with the untrusted *LBS* provider and the anonymization happens in the side of the client and c) the users communicate not only with an untrusted *LBS* provider but also with other users. The basic assumptions in each scenario are briefly presented in the following.

4.1 Anonymization by a trusted server

In the *trusted server* scenario which is depicted in the left part of Figure 2, the users do not send their requests directly to the *LBS* provider, but they communicate through a trusted server, who acts as an *anonymizer*. The anonymizer receives the original service request and anonymizes it, before forwarding it to the *LBS* provider. The anonymization procedure involves removing direct identifiers from the request, and then applying some data transformation to the quasi-identifiers that are contained in the message. A benefit of this architecture is that the truster server does not have to anonymize requests independently, but it can instead perform a bulk anonymization on several requests. The bulk anonymization of requests allows the anonymizer to take into account the information of multiple users, thus it can group the requests into *equivalence classes* which have some collective common property. For example, a user request can be grouped with other requests that share the same or similar quasi-identifiers in order to provide k-anonymity. The creation of equivalence classes is very difficult without the existence of an anonymizer, thus protection against identity is rarely provided without it. An exception to this rule is *Prive* which employs a *peer-to-peer* architecture for creating equivalence classes [30]. Protection against location and attribute disclose can be provided in the absence of an anonymizer, but without using equivalence classes. The communication through an anonymizer is the most commonly adopted architecture in literature [69; 31; 51; 19; 59; 46; 34; 8]. A slightly alternative architecture appears in [21; 27] where the trusted server does not communicate itself with the *LBS* provider, but anonymizes the request and returns it to the client who handles all communication with the *LBS* provider. The authors assume that given the increase of the computational power in handheld devices it might be possible for the anonymization to fully take part in the client side.

4.2 Untrusted server

Another common communication scenario where the user does not trust any server, is depicted in the middle part of Figure 2. In this case, the user is responsible for obfuscating the message he sends to the server in order to protect his privacy. The lack of an anonymizer imposes certain constraints on the type of the anonymization that be offered; the anonymization procedure cannot use information from other users to make the user indistinguishable from other users. Privacy guaranties like k-anonymity or l-diversity (with respect to the sensitive values issued by other similar users) cannot be provided.

Most works that adopt this communication architectures adopt secure protocols that rely on *Private Information Retrieval* [17; 43] for the communication between the user and the *LBS* provider. The first proposal that used PIR to remove the anonymizer from the communication architecture was [29; 41], and was quickly followed by several other works relying again on PIR [56; 40] and on other cryptographic

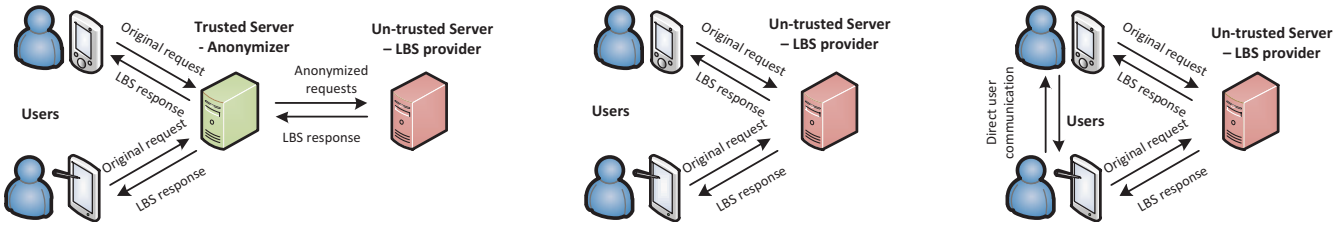

Figure 2: Different architectures for the communication with *LBS* providers.

protocols [55; 58]. In these works, the content of the query is completely hidden from the *LBS* provider though cryptographic means, providing stricter guaranties than most other approaches. On the other hand, the architecture does not allow for hiding the identity of the user or the fact that he has sent a request.

Cryptographic methods are not the only way to to protect the privacy of users in a communication that does not include an anonymizer. In [71] the user sends a fake location to the *LBS* provider, which lies close to his actual location, in order to get his nearest neighbors. In [49; 62] the user replaces in the request his real location with a region that contains it. In all cases the protocols guarantee that the *exact* user location is hidden from the *LBS* provider.

4.3 Untrusted server and user-to-user communication

A special case of the untrusted server scenario, is the scenario where each user wants to also communicate with other users. There are two versions of this scenario in the literature: a) one where the user does not trust the *LBS* provider, nor the other users and b) and one where users trust each other but do not trust the *LBS* provider. The most common application case that adopts the former version of this scenario is that of *proximity based services* where a service depends on nearby users [49]. For example, a proximity based service might inform a user that some of his friends are at nearby locations. Users in this setting want to communicate with their friends or be alerted when they are in their neighborhood, without revealing their exact locations. In [55] the authors propose a cryptographic protocol for performing the proximity test and in [62; 49] the users send a obfuscated version of their location, so they can get approximate answers about to the proximity test.

In [30] the user trusts all other users and communicates with them in a structured peer-to-peer network. The adversary in this scenario is the *LBS* provider and the users collaborate in order to protect themselves against identity disclosure. Users need a trust certification to participate in the network, but they are all considered trustworthy. They are organized in clusters, and each cluster has a leader. Leaders are organized recursively to other clusters with new leaders. The network architecture allows each user to acquire information from nerby users in order to anonymize his request before sending it to the *LBS* provider. It is one of the few example where users can achieve protection against identity and creation of equivalence classes without the need of an anonymizer.

A drastically different architecture for real time communication between users appears in [26]. The paper focuses on privacy preservation in Mobile Ad-hoc Networks (MANETs).

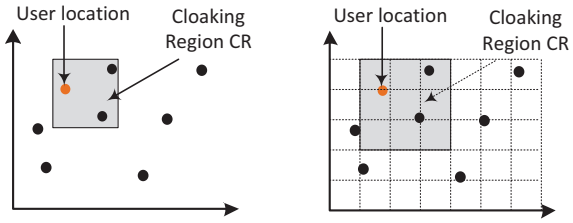

Figure 3: Spatial cloak with arbitrary and grid based cloaking regions

MANETs are self-organized networks of mobile users who communicate in order to exchange information. In this setting there is no *LBS* provider or anonymizer and queries are answered by other members of the network. Every user is considered untrusted and each user is responsible for anonymizing his own messages. In this architecture it is very hard to create equivalence classes and provide protection against identity disclosure.

5. DATA TRANSFORMATIONS AND PROTECTION METHODS

The core idea in *PPDD* is to provide data where certain information has been obfuscated. Depending on how the data have been transformed certain properties can be hidden in order to protect the user privacy and at the same time other data properties are preserved in order to keep data useful. This section describes the most common data transformations that are employed in the various anonymization methods.

5.1 Spatial cloaking

One of the most popular and intuitive data transformations is *spatial cloaking* [35], where the exact location of a user is replaced by a broader region termed *cloaking region (CR)* which almost always contains it. For example, an *LBS* user walking in the center of Athens, might replace her exact location as reported by the GPS or the wireless positioning functionality of her handheld device, with the region that covers her nearby building blocks and streets or even with a predefined region, e.g., "Athens center". The cloaking region $CR(p)$ of a point p is created in such a way that it validates a certain privacy predicate PP, i.e., $PP(CR) = true$. For example, PP might require that the number of users who exist in CR is over k, thus guaranteeing k anonymity. The various PPs are discussed in Section 6.

Spatial cloaking is an adjusted form of the generalization technique used for relational [44], transactional [65] and

other types of data. In privacy preserving techniques for publishing relational data it is common to generalize numerical data to numeric ranges, e.g., 8 to $[5-10]$ or categorical data according to some predefined hierarchy, e.g. "skimmed milk" to "milk". In a similar way, the exact location of users, expressed usually by two coordinates (or three if they are timestamped), is transformed to an arbitrary region that meets the privacy requirements of the data publisher or to predefined regions with known characteristics.

One approach in creating the CR is to use a predefined grid for the map where the users move. The actual location of the user is replaced by a grid cell or by many cells in order to validate the selected PP. Several approaches use a hierarchical organization of grids cells [51; 35]. Partitioning the map to predefined cells is a popular method since it is computationally less expensive than creating arbitrary regions and many works adopt it [49; 62; 21; 33; 35; 19; 51; 46; 8].

Creating arbitrary regions can provide better utility to the anonymized data, since the algorithm can enlarge CR only as much as needed to validate PP and it is not constrained by the grid granularity. The downside is that it is computationally more expensive and it is more prone to minimality attacks [67] as shown in [31; 38]. It is adopted by numerous anonymization methods that address both identity and location disclosure [25; 10; 18; 31; 34; 38; 27; 69; 59; 26]. Note that the CR might be a set of disjoint regions and it might even not contain the original location [23].

5.2 Transformation approaches

A number of approaches instead of cloaking the user location, transform all the data space to a new space, usually with cryptographic methods. The *LBS* provider works on this transformed space, but is not able to interpret user data. Most of the methods in this area rely on Private Information Retrieval (PIR) protocols [17; 43], which allow the client to retrieve data from a database without the database server learning what information was retrieved. The data in the *LBS* provider are encrypted and both sides exchange encrypted messages. At the end of the communication the client is able to decrypt the message of the server which contains the requested information. The idea was first introduced in [29] and independently in [41] and was followed by several other works [56; 40]. In another cryptographic approach [62], the authors propose a protocol that allows private proximity testing between users of a mobile social network. The proposed solution is based on sharing of cryptographic keys between friends. Another cryptographic solution that allows encrypted communication between *LBS* users is proposed in [58].

The basic advantage of the cryptography based methods is the strong privacy guaranties they provide. Unfortunately, this comes at a significant computational cost that does not make them ideal solutions in the case of *LBS* with a very large number of users.

Finally, in order to combine the merits of the strong privacy guaranties provided by PIR based methods with the low computational cost of the protocols that send an obfuscated version of the users' location to the *LBS* server, the authors of [28] propose a hybrid method that combines both PIR and spatial cloaking. In this approach, a broad CR is created for obfuscating the user location and it is sent to the *LBS* provider. After receiving the CR, the user and the *LBS* provider engage into a cryptographic protocol that allows retrieving refined results, without the server being able to learn the user location with more accuracy than that of the CR.

5.3 Reporting of dummy locations

A simple and intuitive approach that significantly differs both from cloaking and transformation based techniques was recently proposed in [71]. In the setting of [71] the user communicates directly with the *LBS* server and sends a request reporting an *anchor*, i.e., a location that is different from his actual location, but lies in a close distance. The basic idea is that the *LBS* will provide a set of answers that will fit the position of the anchor. The correct answers for the actual location of the user, will be a subset of the solution provided for the anchor, thus the user will be able to filter them in the client side.

A dummy based approach is also followed in [16]. The difference here is that the user does not send one location to the *LBS* provider, but instead he sends l different locations, including his real one. The rest of the locations are dummy locations.

5.4 Transformation of timestamps

In most application scenarios in *PPDD* the user location is coupled with a timestamp. Obfuscating time, might be an option for the off-line scenarios, but it is not an easy option for real time anonymization methods since in the latter case the timestamp can be inferred by the time the adversary received the request. In approaches that protect only against location disclose and do not require creating equivalence classes, as discussed in Section 4.2, time could be reported accurately. When the creation of equivalence classes is necessary the only way to create them is to *delay* some requests, until the anonymize has accumulated enough requests to create an equivalence class that satisfies the PP. This approach is followed by numerous works in this area [27; 34; 8]. Usually there is a limit on how long a request can be delayed and if it is not anonymized in time, the request is rejected. In [18] where the anonymizer and the adversary monitor users continuously, users are allowed to issue queries only if they are already grouped in an anonymous equivalence class.

6. ATTACK MODELS AND GUARANTIES

Privacy, even when restricted to the *LBS* case, has many different meanings and it can be breached in various scenarios. Research literature has proposed a variety of privacy guaranties, adjusted to different attack scenarios. The attack scenarios vary depending on the background knowledge of the attacker, on the role of the attacker in the communication architecture, on the goals of the attacker and finally on what the user perceives as important sensitive information. We can divide the various attack modes and privacy guaranties in two main classes: a) attacks against the identity, i.e, the attacker is interested in learning which user issued a request and b) attacks against content, i.e., the attacker is interested in associating a request (from a user he already knows) to a certain property, e.g., exact location. Attacks against content focus mainly on the current location of the user. There are several anonymization methods that provide protection both against identity and content disclosure. Finally, an important class of attack models are those where the attacker can monitor user movement and

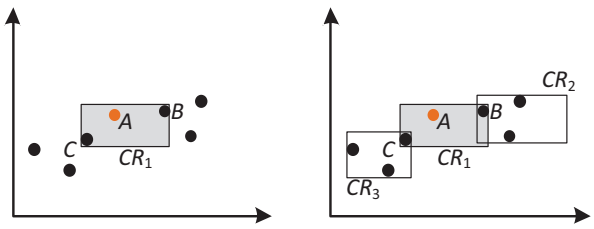

Figure 4: Naive approach to k-anonymity

perform a continuous attack. In the following, some of the most characteristic attack scenarios and privacy guaranties are presented.

6.1 Attacks against identity disclosure

Scenarios of attacks against user identity usually assume the architecture on Scenario A from Figure 2, where an intermediate trusted server handles the anonymization. In these scenarios the adversary is usually the *LBS* provider. Protection against identity disclosure is provided by anonymization methods that guarantee k-anonymity for users that send requests to an *LBS* server. In the basic scenario for k-anonymity in relational data, an adversary knows a) that a user exists in a dataset and b) the quasi-identifiers that are related with each user, e.g., age, zip code etc.. In the context of *LBS* almost all anonymization methods consider the location of the user as a quasi-identifier. The task of the anonymizer is to make each user indistinguishable to the *LBS* provider from at least other $k-1$ users, whose locations are usually close (depending on the information loss metric). Because k-anonymity requires knowing the quasi-identifiers of at least k-users, it is very hard to provide it in scenarios where the trusted server does not exist. A good survey for k-anonymization techniques in *LBS* appears in [32].

In the simplest scenario, the adversary wants to know the identity of a user who issued a service request. His background knowledge (at worst case) consists of a) the knowledge that a user issued a request and b) the location of all users. The data that are available to the adversary are only 1 request per user that contains her location. An intuitive but naive solution based on spatial cloaking is depicted in the left part of Figure 4. Assume that user A issues a request and the anonymizer wants to make her indistinguishable from other $k-1$ users. The anonymizer can issue a k-nearest neighbor (kNN) query to retrieve the $k-1$ closest users to A and then replace their precise location with the MBR of all their locations (CR_1 in Figure 4). This technique is followed by *Center Cloak* [38].

Casper [19; 51] is grid based approach which provides k-anonymity in the same attack scenario and follows a similar technique. *Casper* creates a pyramid, quad-tree like structure that partitions the entire map in disjoint cells. The cells of each layer are grouped to larger cells in higher levels of the hierarchy. When a user issues a request, the anonymizer first checks the smallest cell that contains the user location and sets the CR to the cell area. If there are k total users in the cell, it replaces the user location with CR and forwards the request to the anonymizer. Else, it starts augmenting the CR by adding neighbor cells until the k-user constraint is satisfied.

A similar approach is followed by the *bottom-up* algorithm of *PrivacyGrid* [8]. The bottom-up algorithm starts again by setting the CR as the smallest cell that contains the user location and expands it until the privacy criteria are met. In the expansion phase the neighbor grid cells that contain most other users are added first to the CR. *PrivacyGrid* proposes also a *top-down* algorithm that starts from the maximum CR, which is defined based on user preferences about service utility. If the maximum CR satisfies the privacy criteria then the algorithm proceeds by removing a row or a column of cells from CR until it reaches a CR which is not privacy preserving any more. At each step the row or column with the smaller object count is removed. *PrivacyGrid* provides both k-anonymity and l-diversity. In another approach *Interval Cloak* [35] creates again a hierarchical grid on the user space and generates the CR not by unifying neighbor cells, but instead it climbs up in grid hierarchy and replaces the user location with higher level cells.

A significant problem of the aforementioned approaches is that they are susceptible to minimality attacks [67]. The cause of the problem is that they start with the user location as the center of CR and then they expand it in a deterministic way, trying to minimize information loss. As a result, a different characteristic CR can be created for each request. Consider the naive anonymization shown in the left part of Figure 4. The anonymization algorithm provides 3-anonymity by replacing the user location with the bounding MBR of its 2 nearest neighbors B and C, creating CR_1. In the right part of Figure 4, the CRs that are created to protect users B and C (CR_2 and CR_3 respectively) are depicted. The CR in each case is different. This fact allows the adversary to infer that the request with CR_1 comes from user A. This problem is studied extensively for the first time in [30; 38] and independently in [18]. The solution proposed in [30; 38] and extended in [31] is to add the requirement for *reciprocity* in the creation of a CR for k-anonymity. A CR that contains k users satisfies reciprocity only if the same CR is generated for everyone of the k users. If the same CR is generated for every user of the same equivalence class, then the attacker cannot infer which user is the source of a request with probability over than $1/k$. The *Hilbert Cloak*, proposed in [38], uses a Hilbert curve to transform the two dimensional space of user locations to a one dimensional space. The Hilbert Cloak transformation preserves the spatial locality; points that are close in the two dimensional space are usually close in the one dimensional space. Following the transformation of the 2-D space, the algorithm places the users to buckets of size k. Users are placed to buckets according to their Hilbert value, i.e., the k users with the smallest values are placed to the first buckets, the next k users to the second etc. The CR of any user is the MBR that encloses all users in his bucket. In [31] an improvement in terms of efficiency of *Hilbert Cloak* is proposed, the *Greedy Hilbert Partitioning (GH)*. The GH indexes the data with an R-tree and then limits the Hilbert transformation to the subtree that contains the user whose request needs to be anonymized and at least $k-1$ other users. The algorithm that offers the best quality CR for reciprocal spatial k-anonymity comes again from [31] and it is based on the partitioning algorithm of the R*-tree [9]. The difference is that instead of using the classic partitioning heuristic of the R*-tree, the authors use a heuristic that

takes into account the size of the CRs that are going to be created in each partitioning.

Prive [30] provides k-anonymity with the reciprocity requirement, but works under a very different communication architecture. In *Prive* there is no anonymizer and users communicate directly with the *LBS* provider. Each user is responsible for protecting her own identity by using spatial cloaking. A user creates the CR for her position by taking into account the locations of other nearby users. A CR always contains at least k users. The CR for every user of the same equivalence class is the same, and this is achieved by a similar approach with the one of [38]. Every user is assigned a Hilbert value, which is stored in a distributed B-tree like index among the users. When a user needs to cloak her location, she issues a k-request query to the index and receives the CR that satisfies k-anonymity and reciprocity.

6.1.1 Continuous attacks against user identity

The previous methods provide k-anonymity under the assumption that the adversary knows only the current location of the user. Still, in various application scenarios the user must communicate with the *LBS* provider multiple times, under the same id. Even if the id is a pseudo-identifier inserted by the anonymizer, the adversary gains significant additional knowledge: the previous anonymized locations of the user. An adversary who knows the CRs of previous user requests, can intersect the users that lie inside each CR and limit the candidates for a user id to less than k. To counter such attacks each user has to be grouped in all his requests with the *same* $k-1$ other users. In [18] an anonymization method that addresses the problem of adversaries who continuously monitor the movement of a user is addressed. The work of [34] applies to a similar setting, but here the adversary is also able to extract frequent movement patterns from historical data.

In [10; 68] the authors also propose anonymization methods that provide k-anonymity in application scenarios with continuous queries, but there is a significant difference from the works of [18; 34]: the adversaries here cannot intersect the CRs of all requests to find the common users. Adversaries in this setting can only be aware of frequent patterns in their historical information. The proposed anonymization methods obfuscate the position of a user, by creating a CR that takes into account previously traveled trajectories.

6.2 Attacks on location

The location of a user can act as quasi-identifier, as it usually assumed in the attacks against identity, but it can also be a sensitive value. Unlike the approaches in relational privacy preservation, where most methods avoid transforming the sensitive value, in the *LBS* setting transforming the location is the most common method for protecting user privacy. Again most anonymization methods rely on spatial cloaking. The location of users is often protected by privacy guaranties that are based on l-diversity. l-diversity in the spatial context is achieved in a different way than in the relational context. Instead of creating equivalence classes of users with l different well represented sensitive values, the sensitive value, i.e., the user location, is generalized to a CR that contains l well presented values. For example if a user has visited a hospital and the hospital is a sensitive value, instead of grouping him with other users, the anonymization method creates a CR which contains the hospital and several other locations.

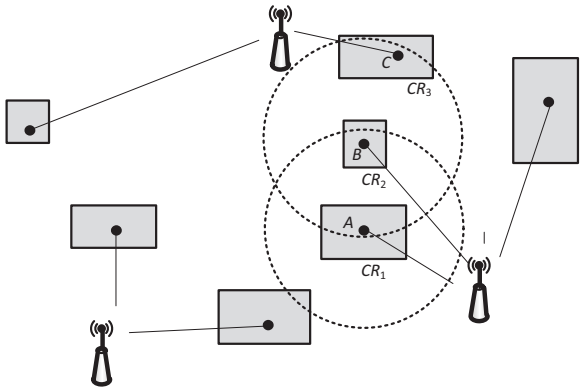

Figure 5: MANET forwarding with spatially cloaked users.

A beneficial consequence of avoiding to create equivalence classes is that the anonymization procedure for protection against location disclosure is cheaper than providing protection against identity disclosure. Instead of requiring tracking and processing other dynamic information (e.g., the locations of other users) the anonymization algorithm only needs to know the map and its properties.

6.2.1 Protection against exact location disclosure

The simplest case of protection against location disclosure is to guarantee that the user location cannot be traced with a granularity less than a threshold, i.e., to put a direct constraint on the size of CR. This is the approach of *Casper* [51] for providing location privacy (complementary to the k-anonymity). The same type of location protection is proposed by *SpaceTwist* [71]. The approach of *SpaceTwist* is suitable for privacy preservation in *LBS* services that provide answers to kNNqueries. The user instead of sending a request with his own position, sends the request with the location of a predefined point on the map termed *anchor* that lies close to him. The *LBS* provider starts sending incrementally the nearest neighbors of the *anchor* to the user. The user filters these results and when he has the k answers he needs, he signals to the *LBS* to stop the transmission of results. An approach based on the creation of dummy locations is proposed in [16]. In this scenario the user sends a request directly to the *LBS* provider and inserts in his request l locations, where one is his real location and the rest $l-1$ are dummies. The dummy locations are created is a way that reflects real movement. Dummy generation algorithm remembers the previous locations of dummies and creates the new one in their proximity. Moreover, the density of users in each neighborhood is taken into account.

Protection against exact location disclosure, but in a very different communication scenario, is studied in [26]. The paper considers privacy issues in Mobile Ad Hoc Networks, where there is no anonymizer or *LBS* provider. In this scenario, depicted in Figure 5, the network is comprised only by users. The requests and the information pass from one user to another. A significant class of routing protocols in MANETs forwards the requests according to the location of each user. For example, user A in Figure 5 who wants to send a message to user C, will send first his message to B, who will then forward it to C. In location-aided routing

protocols a user must be aware of the location of other users, in order to forward a request to the most suitable one. The work of [26] assumes that each user might be a potential adversary, and the sensitive information is location itself. The authors propose adjusting the location-aided routing protocol to spatially cloaked user locations. This way each user can protect her exact location and still allow for the routing algorithms to work. The privacy guaranties that are supported by the proposed routing protocol are guaranties against location disclosure; depending on how the CR is created, it can provide protection against exact location disclosure or a bound to the probability of associating a sensitive feature with a user.

The most effective privacy preserving solutions, i.e., those that provide the stronger protection, are based on cryptographic methods, like PIR [17; 43]. For example, [56] guaranties *strong* location privacy in the evaluation of kNNqueries; the adversary does not learn anything about the location of a user. The method proposed in [29] provides the same strong location privacy but only for single kNNqueries. In [41; 40] another method for evaluating kNNqueries is proposed. A kNNquery is answered through multiple requests. Each single request does not reveal anything about the user location but the adversary could exploit the cardinality of the requests per kNNquery to reveal location information. An additional good property of PIR based methods is that they protect against correlation attacks. Adversaries who can continuously monitor a user, cannot intersect different CRs to find the common users, since no location information is revealed to them at any time.

6.2.2 l-diversity based protection

Putting a minimum size constraint to the CR is an easy and practical privacy protection measure but it is not necessarily meaningful in every case. For example a CR at the city center might contain many different points of interest, thus effectively obscuring the location and the possible activity of a user. On the other hand, the same CR in a large university campus might not really hide anything; the adversary does not know where in the campus the user is, but he knows she is at the university. To address these types of problems many methods perform some type of semantic preprocessing to identify *spatial features* on the map where the users move. A feature is an area or a point that represents an entity with spatial extent, e.g., a hospital. Using this preprocessing they can define variations of l-diversity that are based on the number (or spatial extent) of features that appear in the CR.

In [23] the movement space is modeled as a connected graph where nodes stand for spatial features (e.g., hospitals, schools etc) and edges denote that one location is a direct neighbor of another. The CR in this case is not a continuous area in space, but a set of locations that do not necessarily constitute a single connected component. As a result, the provided privacy guaranty is a type of l-diversity where the actual position of a user might be in any of the semantic locations in CR. In *PrivacyGrid* [8] the authors provide l-diversity combined with k-anonymity. The l-diversity is provided in terms of features of the map as in [23], and the CR is created so that it contains at least l different features. The same guaranty appears as *Weak Location Privacy* in [69]. The CR here is the MBR of the l locations chosen to obfuscate the position of a user.

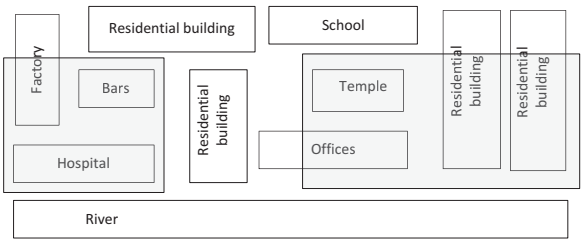

Figure 6: CRs created in the Probe framework

6.2.3 Adversaries with statistical background knowledge

In [69] another form of location l-diversity, the *Strong Location Privacy (SLP)*, is presented, that protects against adversaries who have as background knowledge the association between a query request and a spatial feature. Such adversaries know the *probability distribution function* (*pdf*) of any query over all map features. For example, an adversary might know that queries for nearby gas stations come more frequently from users that are in the street than from users who are in a hospital. The SLP of [69] guarantees that even if the adversary knows the *pdf* of a query, he will not be able to associate a query request with any feature with over a probability threshold. In [21] the authors adopt a similar privacy guaranty. The spatial features are categorized as sensitive and non sensitive. The privacy of a user is endangered only if she is in a sensitive location. In this case her location is cloaked with a CR, in such a way that the attacker cannot associate her with any sensitive feature that lies inside the CR with probability over $1/l$. A novelty of the approach is that the CRs are created off-line and *only for sensitive regions*. Figure 6 depicts two CRs created in the PROBE frameworks to mask the locations of users who are in the two sensitive features of the map, the hospital and the temple. The CRs are depicted as gray boxes in the map. If a user issues a request from any location inside a CR, then her location is replaced with the CR in the anonymization. Even if the user is not in the hospital or the temple, but she is simply inside the CR, her location will be obfuscated. This technique protects the user location from reverse engineering attacks. The attack model assumes that the adversary has the *pdf* of users in any point of the map. For example, the adversary might know that the probability of a request coming from the river in Figure 6 is 0 and that the respective probability for the hospital is 0.05. The probability of a user whose location has been obfuscated to be at a specific feature is calculated based on the *pdf*, the *area* of the feature and the total area of the CR.

6.2.4 Location protection in proximity services

A special case of location privacy arises in the usage of proximity services. In the delivery of proximity services users not only communicate with the *LBS* provider but they also communicate with each other, as depicted in the last scenario of Figure 2. In these scenarios the users want to hide their exact location both from the *LBS* provider and from other users, without the help of an anonymizer. At the same time they want to know if they are in the *proximity* of some other users. A user A is in the proximity of user B if their distance is smaller than a threshold d. In the solution proposed in

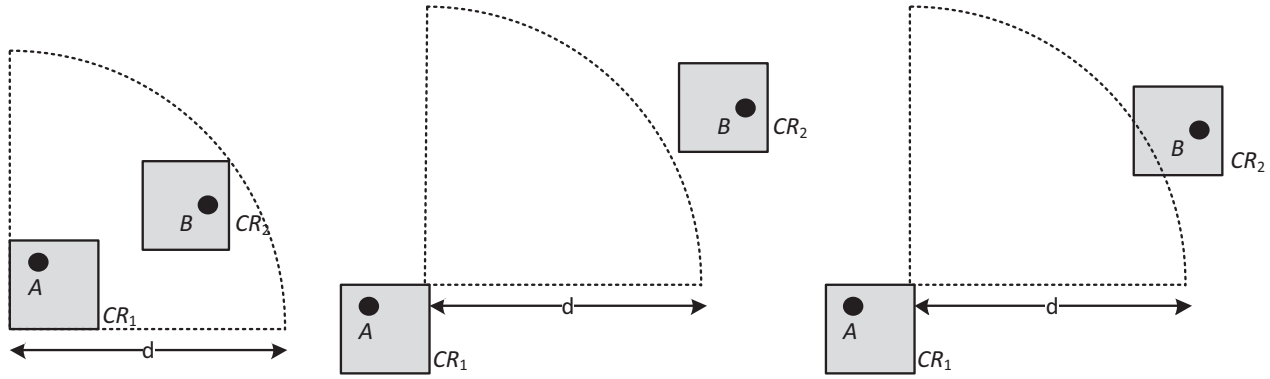

Figure 7: Proximity test on user CRs.

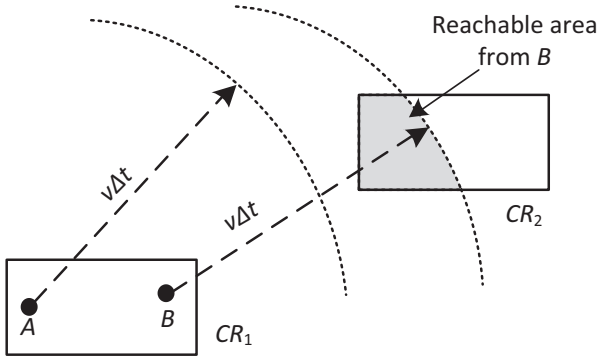

Figure 8: Reachable areas in successive CRs with maximum velocity v

[49; 50] the users create a CR based only on their preferences for their location and they send it to the *LBS* server. The *LBS* server can calculate the minimum and maximum distance between the users as shown in Figure 7. If the proximity threshold is greater than the maximum distance then the two users are certainly in proximity so they can get a definite answer. They get also a negative answer if the minimum distance is greater than the proximity threshold. In the in-between cases, the users engage direct communication through an secure-two-party computation protocol that allows them to verify whether they are in proximity or not, without revealing their actual locations. In a different approach, *FriendLocator* [66] and *VincinityLocator* [62] allow users to perform proximity tests through an encrypted protocol. The map is partitioned in grid cells and the user creates a CR as a collection of neighboring cells. The cell ids are then encrypted and sent to the *LBS* provider, who can decide on the proximity based on the encrypted values. An approach based on a distance-preserving mapping is proposed in [60]. A weakness of this solution is that it is possible for the attacker to easily guess the mapping function [47].

6.2.5 Protection in continuous queries

A realistic but significantly more powerful adversary is considered in [27]. As already mentioned in the methods for protecting against identity disclosure in Section 6.1, there are several practical scenarios where the user has to communicate multiple times with the adversary under the same pseudonym. This allows the adversary to intersect the equivalence classes that correspond to the different CRs and narrow down the potential real users that might have issued a request. The authors of [27] observe that in real scenarios the adversary can easily have additional background knowledge about the speed limits of different transportation means. For example, if the adversary infers that a user is on a car he can be certain that he will not be moving with more that 200km/hour in a city environment. The knowledge about the maximum velocity of a user can be exploited by an adversary that monitors continuously the requests of a user. If the adversary knows the cloaking regions CR_1 and CR_2 of the positions where the same user issues two requests, he can calculate the maximum distance a user might have traveled from every point of CR_1 and prune parts of CR_2 using the maximum velocity, as shown in Figure 8. The adversary can also do the reverse pruning: he can prune the parts of CR_1 that are so far away from any point of CR_2, that the user could not have been there if he issued a request from CR_2 in a later time. The paper proposes an anonymization method that creates CRs in such a way that an adversary, who can continuously monitor user requests and has the maximum user velocities as background information, cannot pinpoint the exact user location with accuracy greater than the intended CR. The paper provides a more powerful guaranty against attackers who additionally have semantic information about the map and know the sensitive spatial features on it, e.g., hospitals. In this case the anonymization algorithm guaranties that the adversary will not be able to associate a user with a sensitive feature with probability more than a threshold. The probability that associates a user with a sensitive feature is calculated based on the area of the feature that lies in a CR to the total area of the CR.

6.3 Attacks on other query content

Apart from the identity and the location of the user, it is also the rest of the query content that might be sensitive. An anonymization method that protects the query content from adversaries that might use location as a quasi-identifier is presented in [59]. In this setting the attacker can keep the history of previously issued queries, and based on this information he can create the *pdf* of each query (different service invocation) to each user. In other words he has the proba-

bility that associates each different service request with each user. This background knowledge can be used to decide that not all users in an equivalence class are associated with different requests with equal probability. Equivalence classes are created again by spatial cloaking. The proposed method guarantees k-anonymity and t-closeness [45]. A CR has always more than k users to provide k-anonymity but also the distribution of requests from users in CR does not differ by the global *pdf* more than a certain threshold. For example assume a threshold $t = 5\%$ and a service s with a probability $p = 30\%$. According to the t-closeness guaranty of [59], the probability p' with which the adversary is able to associate any user from a single CR with a request of service s lies inside the bounds $p - t \leq p' \leq p + t$, i.e, $25\% \leq p' \leq 30\%$.
Finally, in [46] the authors consider l-diversity for the query content, and location only acts as a quasi-identifier. The creation of equivalence classes is performed by spatial cloaking. The authors define *query entropy* qe as $qe = -\sum p_i \log p_i$, where the p_i is the probability that a query q is issued. The l diversity is achieved by bounding the qe of each query in an equivalence class with $\log l$.

7. PRIVACY PRESERVATION IN LOCATION BASED SOCIAL NETWORKS

Location Based Social Networks (*LBSN*) become increasingly popular and a growing number of users participates by sharing their location information. Users seem to ignore privacy related dangers and simply post their location to wide audiences with little regulation. In *LBSN*s, like *Foursquare* (`http://4sq.com`), users alert their friends when they are in certain location by "checking-in" and they even make this information available to wider audiences by having their alerts being forwarded to *Twitter* (`http://twitter.com`). The potential dangers are high; for example, *PleaseRobMe* (`http://pleaserobme.com/`) demonstrated that it is possible to use this information to infer when users are out of their apartments. There have also been a series of studies which show that users do not put great value on their location privacy [20; 3; 7; 14].
On the other hand, privacy is not an obvious quality and people are mostly aware of it when it is missed. For example, when [20] examined the value that users put to privacy by nationality, it noticed that Greek citizens valued their privacy many times more that the rest of nationalities participating in the survey. The paper attributes this deviation to the fact the privacy related issues have been widely discussed in Greece in the period before the survey, following an eavesdropping scandal. Moreover, [13] supports that the participation of young adult users in Facebook by making their their data available, does not signify that they do not care for their privacy, since they actively adjust their privacy settings. In [6] the authors support that the users' desire to protect their privacy is not always consistent with their behavior. Often they are not well aware of the public nature of the network and of their audience [61; 12]. [11] argues that electronic publication has disrupted the boundaries between public and private and that users' control over data has been undermined. The paper documents four basic characteristics of the electronic publication of personal data, which disrupt the users' efforts to control their privacy in social networks:

- *Persistence.* Whatever is published electronically can stay available for a very long period of time. This is in contrast with real life discussions, where content is ephemeral and easily forgotten.

- *Replicability.* Electronic documents can be very easily replicated with absolute accuracy and reproduced in different audiences. Moreover, alterations in the reproduction of documents can be performed in such a way that it is not easy for the reader to understand which is the original and which is the alteration.

- *Scalability.* Often users are not aware or cannot predict the audience of the information they publish. Social networks allow information to reach very large audiences, whereas the the publisher might have intended to share information with a specific group.

- *Searchability.* Search on the Internet has greatly increased the ability of users to access information. Personal data can be traced with an effectiveness that is unimaginable in the non-digital world.

These properties make the communication and the dissemination of personal information in the Internet drastically different from the non-digital and especially the oral communication. Publishing information in such an environment has implications on users' privacy that are not easily foreseen by a large part of *LBSN* users. There are several initiatives to help users enforce more effective control on their data. For example, following a regulatory approach, the French government wants to give to users the "right to be forgotten" [2]. Practically, this is translated to the right of having personal information removed from the WWW after some time, in order to limit the persistence of data. Still, it is not easy to see how such a measure can be enforced from a technical point of view.
An important challenge for *PPDD* is to provide techniques that will help users wield better control over their data. Until now privacy in *LBSN* has only been protected by allowing users to set an access policy for their data. The policy based paradigm is a poor tool for controlling information dissemination. Data can be either accessible or inaccessible by a certain user or group. On the other hand, in real life the same information is disseminated with varying degrees of accuracy depending on the audience. In a similar way, *PPDD* can help to enrich privacy policies in *LBSN*, by offering varying degrees of access. Following techniques like spatial cloaking, information can be accessed in different degrees of accuracy depending on the audience. Moreover, regulatory approaches could specify more refined rules using privacy guaranties from *PPDD* about how information about minors or other user groups should appear in social networks.

8. CONCLUSIONS AND FUTURE DIRECTIONS

The focus of this survey is on the privacy preserving data dissemination techniques that have been developed especially for spatial data. These techniques preserve users' privacy by transforming the data that describe them. The transformation procedure, termed *anonymization*, is performed to prohibit malicious adversaries from using background information, to identify people in the published data. The survey

presents a broad range of techniques focusing on the models that were adopted at each case.

The majority of works in *PPDD* for location data studies the problem of the real time communication between users and *LBS* providers. In Sections 4-6, a classification of the methods was presented based on the communication architectures, on the data transformation that is used by the anonymization algorithm and on the attack and privacy model that is adopted. Basic properties of the different privacy preservation methods were highlighted. We saw that architectures without an anonymizer cannot easily support anonymization methods that require the creation of equivalence classes and that spatial cloaking achieves location based protection not by grouping records, but by grouping features on a map. The case of most powerful attackers, who can monitor continuously the requests of a user was explored and key ideas for addressing such attacks were presented.

There are several technical challenges that arise in *PPDD* for location data. The case of continuous attacks, which is a realistic scenario for many applications, has only been partially studied. Moreover, most works focus on devising strong privacy guaranties and the data utility is studied only through a few artificial metrics. Investigating in more depth how to efficiently preserve data utility remains an important issue in *PPDD* . Apart from the technical challenges, there is a important question about how the existing research answers the problems of privacy protection in the *location based social networks*, which are becoming increasingly popular. Users publish more and more information about their location and movement, without often being aware of how exposed they are. There are already several studies in social sciences that investigate user behavior and privacy dangers in the social networks [20; 3; 7; 14; 61]. A challenge for the *PPDD* research is to bridge the gap between the low level anonymization techniques, which provide strict but limited guaranties (e.g., identity protection through k-anonymity, protection of sensitive information using l-diversity etc.) and the higher level user requirements on privacy like audience regulation, limitations to data persistence etc.

9. REFERENCES

[1] Geopriv (http://datatracker.ietf.org/wg/geopriv/charter/).

[2] http://www.gouvernement.fr/gouvernement/charte-du-droit-a-l-oubli-numerique-mieux-proteger-les-donnees-personnelles-des-interna.

[3] Boombox report on location-based social networks, Septempber 2010.

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

[5] O. Abul, F. Bonchi, and M. Nanni. Anonymization of moving objects databases by clustering and perturbation. *Inf. Syst*, 35(8):884–910, 2010.

[6] A. Acquisti and R. Gross. Imagined Communities: Awareness, Information Sharing, and Privacy on the Facebook. In *Privacy Enhancing Technologies*, chapter 3, pages 36–58. 2006.

[7] S. Ahern, D. Eckles, N. Good, S. King, M. Naaman, and R. Nair. Over-exposed?: privacy patterns and considerations in online and mobile photo sharing. In *CHI*, pages 357–366. ACM, 2007.

[8] B. Bamba, L. Liu, P. Pesti, and T. Wang. Supporting anonymous location queries in mobile environments with privacygrid. In *WWW*, pages 237–246. ACM, 2008.

[9] N. Beckmann, H.-P. Kriegel, R. Schneider, and B. Seeger. The R*-tree: an efficient and robust access method for points and rectangles. In *ACM SIGMOD*, pages 322–331, 1990.

[10] C. Bettini, X. S. Wang, and S. Jajodia. Protecting privacy against location-based personal identification. In *Secure Data Management*, pages 185–199, 2005.

[11] D. Boyd. Social network sites: Public, private, or what? *Knowledge Tree*, (13), 2007.

[12] D. Boyd and N. Ellison. Social network sites: Definition, history, and scholarship. *Journal of Computer-Mediated Communication*, 1(13), 2007.

[13] D. Boyd and E. Hargittai. Facebook privacy settings: Who cares? *First Monday*, 15(8), 2010.

[14] A. J. B. Brush, J. Krumm, and J. Scott. Exploring end user preferences for location obfuscation, location-based services, and the value of location. In *UbiComp*, pages 95–104. ACM, 2010.

[15] B.-C. Chen, D. Kifer, K. LeFevre, and A. Machanavajjhala. Privacy-preserving data publishing. *Foundations and Trends in Databases*, 2(1-2):1–167, 2009.

[16] E.-A. Cho, C.-J. Moon, H.-S. Im, and D.-K. Baik. An anonymous communication model for privacy-enhanced location based service using an echo agent. In *ICUIMC*, pages 290–297, 2009.

[17] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan. Private information retrieval. *FOCS*, 0:41, 1995.

[18] C.-Y. Chow and M. F. Mokbel. Enabling private continuous queries for revealed user locations. In *SSTD*, pages 258–275, 2007.

[19] C.-Y. Chow, M. F. Mokbel, and W. G. Aref. Casper*: Query processing for location services without compromising privacy. *ACM Transactions on Database Systems*, 34(4):1–24, 2009.

[20] D. Cvrcek, M. Kumpost, V. Matyas, and G. Danezis. A study on the value of location privacy. In *WPES*, pages 109–118. ACM, 2006.

[21] M. L. Damiani, E. Bertino, and C. Silvestri. The PROBE framework for the personalized cloaking of private locations. *Transactions on Data Privacy*, 3(2):123–148, 2010.

[22] J. Domingo-Ferrer, M. Sramka, and R. Trujillo-Rasua. Privacy-preserving publication of trajectories using microaggregation. In *SPRINGL*, pages 26–33, 2010.

[23] M. Duckham and L. Kulik. A formal model of obfuscation and negotiation for location privacy. In *Pervasive*, pages 152–170, 2005.

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

[25] B. Gedik and L. Liu. Location privacy in mobile systems: A personalized anonymization model. In *ICDCS*, pages 620–629, 2005.

[26] G. Ghinita, M. Azarmi, and E. Bertino. Privacy-aware location-aided routing in mobile ad hoc networks. In *MDM*, pages 65–74, 2010.

[27] G. Ghinita, M. L. Damiani, C. Silvestri, and E. Bertino. Preventing velocity-based linkage attacks in location-aware applications. In *ACM GIS*, pages 246–255, 2009.

[28] G. Ghinita, P. Kalnis, M. Kantarcioglu, and E. Bertino. A hybrid technique for private location-based queries with database protection. In *SSTD*, pages 98–116, 2009.

[29] G. Ghinita, P. Kalnis, A. Khoshgozaran, C. Shahabi, and K.-L. Tan. Private queries in location based services: anonymizers are not necessary. In *ACM SIGMOD*, pages 121–132. ACM, 2008.

[30] G. Ghinita, P. Kalnis, and S. Skiadopoulos. Prive: anonymous location-based queries in distributed mobile systems. In *WWW*, pages 371–380, 2007.

[31] G. Ghinita, K. Zhao, D. Papadias, and P. Kalnis. A reciprocal framework for spatial K-anonymity. *Inf. Syst*, 35(3):299–314, 2010.

[32] A. Gkoulalas-Divanis, P. Kalnis, and V. S. Verykios. Providing K-anonymity in location based services. *SIGKDD Explorations*, 12(1):3–10, 2010.

[33] A. Gkoulalas-Divanis and V. S. Verykios. A free terrain model for trajectory k-anonymity. In *DEXA*, pages 49–56, 2008.

[34] A. Gkoulalas-Divanis, V. S. Verykios, and M. F. Mokbel. Identifying unsafe routes for network-based trajectory privacy. In *SDM*, pages 942–953. SIAM, 2009.

[35] M. Gruteser and D. Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. In *MobiSys*, pages 31–42, 2003.

[36] U. Hengartner and P. Steenkiste. Implementing access control to people location information. In *SACMAT*, pages 11–20, New York, NY, USA, 2004. ACM.

[37] U. Hengartner and P. Steenkiste. Access control to people location information. *ACM Trans. Inf. Syst. Secur.*, 8:424–456, November 2005.

[38] P. Kalnis, G. Ghinita, K. Mouratidis, and D. Papadias. Preventing location-based identity inference in anonymous spatial queries. *IEEE Trans. Knowl. Data Eng.*, 19(12):1719–1733, 2007.

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

[40] A. Khoshgozaran, C. Shahabi, and H. Shirani-Mehr. Location privacy: going beyond K-anonymity, cloaking and anonymizers. *Knowl. Inf. Syst*, 26(3):435–465, 2011.

[41] A. Khoshgozaran, H. Shirani-Mehr, and C. Shahabi. Spiral: A scalable private information retrieval approach to location privacy. In *MDMW*, pages 55–62, 2008.

[42] J. Krumm. A survey of computational location privacy. *Personal and Ubiquitous Computing*, 13(6):391–399, 2009.

[43] E. Kushilevitz and R. Ostrovsky. Replication is not needed: single database, computationally-private information retrieval. In *FOCS*, pages 364–, Washington, DC, USA, 1997. IEEE Computer Society.

[44] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan. Incognito: Efficient Full-domain k-Anonymity. In *SIGMOD*, pages 49–60, 2005.

[45] N. Li, T. Li, and S. Venkatasubramanian. t-Closeness: Privacy Beyond k-Anonymity and l-Diversity. In *ICDE*, pages 106–115, 2007.

[46] F. Liu, K. A. Hua, and Y. Cai. Query l-diversity in location-based services. In *Mobile Data Management*, pages 436–442. IEEE Computer Society, 2009.

[47] K. Liu, C. Giannella, and H. Kargupta. An attacker's view of distance preserving maps for privacy preserving data mining. In *PKDD'06*, pages 297–308, 2006.

[48] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. l-Diversity: Privacy Beyond k-Anonymity. In *ICDE*, 2006.

[49] S. Mascetti, C. Bettini, D. Freni, X. S. Wang, and S. Jajodia. Privacy-aware proximity based services. In *MDM*, pages 31–40, 2009.

[50] S. Mascetti, D. Freni, C. Bettini, X. Wang, and S. Jajodia. Privacy in geo-social networks: proximity notification with untrusted service providers and curious buddies. *The VLDB Journal*, pages 1–26, 2010. 10.1007/s00778-010-0213-7.

[51] M. F. Mokbel, C.-Y. Chow, and W. G. Aref. The new casper: A privacy-aware location-based database server. In *ICDE*, pages 1499–1500. IEEE, 2007.

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

[53] A. Monreale, R. Trasarti, C. Renso, D. Pedreschi, and V. Bogorny. Preserving privacy in semantic-rich trajectories of human mobility. In *SPRINGL*, pages 47–54. ACM, 2010.

[54] G. Myles, A. Friday, and N. Davies. Preserving privacy in environments with location-based applications. *IEEE Pervasive Computing*, 2:56–64, January 2003.

[55] A. Narayanan, N. Thiagarajan, M. Lakhani, M. Hamburg, and D. Boneh. Location privacy via private proximity testing. In *NDSS*, 2011. To appear.

[56] S. Papadopoulos, S. Bakiras, and D. Papadias. Nearest neighbor search with strong location privacy. *PVLDB*, 3(1):619–629, 2010.

[57] N. Poolsappasit and I. Ray. Towards achieving personalized privacy for location-based services. *Trans. Data Privacy*, 2:77–99, April 2009.

[58] K. P. N. Puttaswamy and B. Y. Zhao. Preserving privacy in location-based mobile social applications. In *HotMobile*, pages 1–6, 2010.

[59] D. Riboni, L. Pareschi, C. Bettini, and S. Jajodia. Preserving anonymity of recurrent location-based queries. In *TIME*, pages 62–69, 2009.

[60] P. Ruppel, G. Treu, A. Kupper, and C. Linnhoff-Popien. Anonymous user tracking for location-based community services. In *LoCA*, page 116. Springer-Verlag New York Inc, 2006.

[61] N. M. Sadeh, J. I. Hong, L. F. Cranor, I. Fette, P. G. Kelley, M. K. Prabaker, and J. Rao. Understanding and capturing people's privacy policies in a mobile social networking application. *Personal and Ubiquitous Computing*, 13(6):401–412, 2009.

[62] L. Siksnys, J. R. Thomsen, S. Saltenis, and M. L. Yiu. Private and flexible proximity detection in mobile social networks. In *MDM*, pages 75–84, 2010.

[63] L. Sweeney. k-Anonymity: A Model for Protecting Privacy. *IJUFKS*, 10(5), 2002.

[64] M. Terrovitis and N. Mamoulis. Privacy Preservation in the Publication of Trajectories. In *MDM*, 2008.

[65] M. Terrovitis, N. Mamoulis, and P. Kalnis. Privacy-preserving Anonymization of Set-valued Data. *PVLDB*, 1(1), 2008.

[66] L. Šikšnys, J. R. Thomsen, S. Šaltenis, M. L. Yiu, and O. Andersen. A location privacy aware friend locator. In *SSTD*, pages 405–410, Berlin, Heidelberg, 2009. Springer-Verlag.

[67] R. C.-W. Wong, A. W.-C. Fu, K. Wang, and J. Pei. Minimality attack in privacy preserving data publishing. In *VLDB*, pages 543–554. VLDB Endowment, 2007.

[68] T. Xu and Y. Cai. Exploring historical location data for anonymity preservation in location-based services. In *INFOCOM*, pages 547–555, 2008.

[69] M. Xue, P. Kalnis, and H. K. Pung. Location diversity: Enhanced privacy protection in location based services. In *LoCA*, 2009.

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

[71] M. L. Yiu, C. S. Jensen, X. Huang, and H. Lu. Spacetwist: Managing the trade-offs among location privacy, query performance, and query accuracy in mobile services. In *ICDE*, pages 366–375. IEEE, 2008.