Methods for Location Privacy: A comparative overview

Konstantinos Chatzikokolakis, Ehab Elsalamouny, Catuscia Palamidessi, Anna Pazii

Methods for Location Privacy: A comparative overview*

Kostantinos Chatzikokolakis, Catuscia Palamidessi and Anna Pazii

INRIA Saclay and LIX, École Polytechnique

December 22, 2016

Abstract

The growing popularity of location-based services, allowing to collect huge amounts of information regarding users' location, has started raising serious privacy concerns. In this report we analyze the various kind of privacy breaches that may arise in connection with the use of location-based services, and we survey and compare the metrics and the mechanisms that have been proposed in the literature.

Contents

*This work is supported by Renault R&D.

1 The problems of privacy in location-based services

The widespread use of Location-Based Services (LBS) in today's world has created new risks to user privacy that users are increasingly becoming aware of. In large part, the worries are caused by the shocking episodes of violations and leaks that keep appearing on the news. For instance, on April 20th, 2011 it was discovered that the iPhones were storing and collecting location data from their users, syncing them with iTunes and transmitting them to Apple, all without the users' knowledge. More recently, the Guardian has revealed, on the basis of the documents provided by Edward Snowden, that the NSA and the GCHQ have been using certain smartphone apps, such as the wildly popular Angry Birds game, to collect users' private information such as age, gender and location [Bal14], and again, without the user's knowledge.

The concerns about location privacy may seem exaggerated at first, but one can see that they are fully justified when thinking of the possible malicious uses of location information, such as robbing and stalking, or anyway hurting the user, one way or another. For instance, in Wisconsin there were episodes of men were tracking women with GPS or other location devices [Orl03]. In California, records from automatic toll booths on bridges were used in divorce proceedings to prove claims about suspicious deplacements of spouses [Sim07]. The application "Girls Around Me", combines social media and location information to find nearby women (who didn't necessarily agreed to be found), and, with one click the user can access the Facebook profiles of targeted girls [Bro12]. Particularly worrisome is the perspective of potential combination with the users' most sensitive information, such as sexual orientation.

To some extent, also the research and the experimentation on privacy contribute to raise the awareness about the practical risks. For instance, the "Please Rob Me" website [`http://pleaserobme.com/`] aggregates location check-ins

and presents them as "robbery opportunities", pointing out the fact that publicly announcing one's location effectively reveals to the world that they are not home.

1.1 Classification of threats

Following [FS14], we classify the concerns about the leakage of location information into three major kinds of threats:

Tracking Threat: An adversary collecting continuously the location updates of the user might be able to identify the users mobility patterns (frequently traveled routes) and predict his present and future location with high accuracy by leveraging the typical peoples mobility habits [JHP00, XZZ+13].

Identification Threat: The adversary can use the user's traces as quasi-identifiers to reveal the users identity from anonymous location traces. This may happen even if the adversary accesses the users locations only sporadically, since he might be able to infer his frequently visited locations, such as home and work. This is the most studied kind of threat in the literature, and we are going to expand on it in the next section.

Profiling Threat: The users mobility traces typically contain his points of interest, that the adversary can use to profile him. Examples include health clinics, religious places, areas which may revel his sexual inclinations, etc. [AS03]. The practice of location profiling is likely to increase in the future, as marketers are becoming more and more aware of its potential to gain visibility of consumer behavior in the real world, and to help targeting their marketing efforts. Indeed, location profiling seems to provide insights into offline activity at a level comparable to that of web or mobile app analytics for online activity. There are already various companies that provide this kind of services. For instance, Urban Airship (`https://www.urbanairship.com/`) offers tools that produce audience profiles by combining in-app behaviors, user preferences, and location. With location specifically, brands can segment mobile users based on where they are at currently or where they have been in the past.

1.2 Identification of the user from his traces

In this section we focus on the threat constituted by using location data for fingerprinting the user, namely for finding out the identity of the person who has originated the data. In short, the problem raises by the fact that mobility traces may be *unique* to an individual, and they can therefore allow identifying that individual like the ridges on his finger. Of course, as with traditional fingerprinting, some information about the person to be identified needs to have been previously recorded.

1.2.1 Uniqueness of human mobility traces.

Recently, de Montjoye et al. [dMHVB13] conducted a statistical study on the uniqueness of human mobility traces. They examined fifteen months of human mobility traces generated by 1.5 million of individuals, who were users of a mobile phone operator. These traces were collected by the phone company in a dataset in the following way: Each time a user interacts with the mobile phone operator network by initiating or receiving a call or a text message, the location of the connecting antenna is recorded in the dataset together with the time of the event, and linked to previous location-time points of the same user already in the dataset, via the user id, so to form a trace (one trace for each user). Successively, the traces were anonymized. The experiments showed that human mobility traces are highly unique: In fact, when the temporal granularity was fixed to an hour, and the spatial granularity equal to that given by the carrier's antennas, 4 spatio-temporal points, randomly drawn from a trace, were enough to uniquely identify the trace in 95% of the cases. They also observed that the uniqueness of mobility traces decays approximately as the $1/10$ power of the spatial and temporal resolution. Hence, they concluded that even coarse datasets provide little anonymity.

Song et al. [SDB14] conducted similar experiments on a dataset of location data generated by about a million users over a period of a week. Their results confirm that, even with a low resolution, location traces can be identified with only a few spatio-temporal points. In particular, they show that 2 points are enough to uniquely identify a trace in 60% of the cases.

Rossi et al. [RWM15] came to similar conclusions, but using location data generated using the GPS. The main difference with respect to previous works is that the points used for the re-identification did not need to be present in the training set. Thanks to the higher precision of the GPS, they were able to show that 2 spatio-temporal points were enough to identify almost 100% of the traces.

1.2.2 Reconstructing traces from location samples

Typically, there can be various users repeatedly updating and and sending their positions on the map to some LBS. Hence, collecting these locations may result in a mix-up of traces left by different individuals. Un-mixing the locations, i.e., reconstructing the individual traces, can be done easily when the data are associated to some invariant attribute, like, for instance, a pseudonym. Even when the data are completely anonymous, however, the traces can often still be reconstructed by linking the location samples. Clearly, the higher is the sample frequency compared to the user density in the area, the easier it is to recognize a trace. In fact, the next point in a trajectory will be at a distance determined by the speed of the user and the time in between the two updates. The reconstruction of a trace can also be facilitated by correlating location samples with likely routes on a map. Finally, the task can be enhanced by using a model of typical trajectories constructed on the basis of prior observations on

the population movements.

The first attempt to reconstruct the traces from completely anonymized mobility data (i.e., without any pseudonyms) was by Gruteser et al. [GH05]. They used a multi-tracking algorithm to identify individual mobility traces from a collection of anonymized location samples generated by multiple users. They tested their algorithm on a collection of GPS traces generated by the students of a university campus, and their experiments showed that often individuals used to travel along the same unique route and could therefore be re-identified. Their system however was prone to misclassification of crossing paths, as it was unable to determine whether the paths of two or more individuals actually crossed or just touched.

More recently, Tsoukaneri et al. [TTLM16] developed a mechanism called *Comber* which is able to disentangle the traces by using a generic, empirically derived histogram of user speeds. The authors evaluated Comber with two different datasets, MDC and GeoLife, which consist of GPS-based mobility traces collected in Lausanne and Beijing, respectively. Each of these datasets span more than a year and include location information of about 180 users. Their results show that Comber is able to infer the original traces of the users with more than 90% accuracy.

1.2.3 Linking traces to users' identity

There has been a lot of work showing that it is possible to infer user identities from anonymous traces, especially when the traces are pseudonymized (i.e. the real identity has been replaced by a pseudonym) rather than completely anonymized. Beresford and Stajano [BS03] already pointed out that the re-identification risks of LBS' users employing pseudonyms: they showed that almost all location traces of AT&T Labs Cambridge employees collected from the Active Bat system could be correctly identified by knowing the office positions of the workers and by keeping track of the frequency of visits of a given pseudonym to each office.

Many of the attacks on pseudonymized traces are, like the above, based on observing the frequent presence of the pseudonyms in specific locations that can be easily linked to a certain individual, like home or office. For instance, Krumm [Kru07] proposed various algorithms to infer users home address, and used a web search engine in order to reveal the real identities of the subjects. Notably, Golle and Partridge [GP09] showed that if the locations of an individuals home and workplace can both be known at the precision of a census block, then they would allow to uniquely identify most of the U.S. working population. Obfuscating these locations can provide some protection: by reducing the granularity to that of a census track, the median size of the anonymity set (i.e., the number of people sharing the same pair) went down to 21. However, the location data of people who lived and worked in different regions could still be re-identified much more easily.

Even when the location data are completely anonymized (i.e., no pseudonym is used) though, it is still possible to retrieve the user's identity if the attacker

disposes of side information about the user. Several works in the literature have investigated this problem, particularly in the case in which a database of users' profiles in the form of previously collected traces, called *the training set*, is available to the adversary. In general, the idea is that the adversary will use the training set to build a representation of the users' typical movements. Thus each user will be associated to a mathematical model of his past traces, playing the role of a signature. This model can be, for instance, a Markov chain, but other models have been investigated as well. Then the attacker will collect one or more of the victim's (sanitized) traces, *the testing set*, from which he will build a model as well. The latter is then compared to the models of the training set, according to some similarity criterium, and the user profile most likely to correspond to the target user is finally selected.

De Mulder et al. [DMDBP08] investigated this kind of attack on mobility traces generated by a GSM cellular network. They developed two methods based on different models and on the cosine similarity measure, and evaluated them on the Reality Mining dataset made available by the MIT Media Lab, which consists of the location traces of one hundred human subjects at MIT during the $2004-2005$ academic year, collected using one hundred instrumented Nokia 6600 smart phones. With the best of those methods, they were able to re-identify about 80% of the users. It is to be noted that a trace generated by a GSM network is formed by the sequence of all cells that the user has visited along his path, i.e., it is not possible to skip cells by "jumping" to a non-adjacent cell. This may affect the success rate when compared with the case in which the traces consist of locations generated dynamically with, say, a GPS.

Ma et al. [MYYR10] considered also two kinds of adversaries: the passive one, retrieving the testing set from a public source, and the active one that can deliberately participate or perturb the data collection phase in order to gain additional knowledge. The authors considered four different estimators to measure the similarity between mobility traces: the Maximum Likelihood Estimator, relying on the Euclidean distance, the Minimum Square Approach, computing the sum of the square of the difference between the traces, the Basic Approach, which assumes that the traces might been perturbed by uniform noise, and the Weighted Exponential Approach, which is similar to previous one except that no assumption is made on the type of noise generated. The authors tested their methods on two datasets: the Crawdad repository, recording the movements of San Francisco YellowCabs, and a collection of traces generated by the public buses in Shanghai city. They obtained a success rate of de-anonymization of 80% to 90%, even in the presence of noise.

Both [MYYR10] and [DMDBP08], however, took the samples to generate the testing set directly from the training set. Clearly such way of proceedings introduced a bias that may have resulted in an overly strong success rate in the re-identification results. In fact Gambs et al. [GKdPC14] showed that there is a substantial difference in the success rate when the training set and the testing set are separated. They used a model based on Mobility Markov Chains, namely Markov chains where the states are locations. They considered various similarity measures between such chains, and tested their methods on several

GPS datasets, including MDC and Geolife. For each individual, they split his mobility traces, chronologically ordered, into two disjoint parts of approximately the same size: the first half formed the training set, and the second half the testing set. Thus the training and the testing data were not only disjoint, but also separated in time. With such split, they were able to re-identify between 35% and 45% or the users. For comparison, they repeated the experiments also without splitting, i.e., using the same set of traces for training and for testing, and obtained, in this case, a success rate of almost 100%! Of course, this comparison is not completely fair because they used as testing set exactly the same as the training set, instead than a subset as in previous works. Nevertheless, such high success rate shows that (1) the training set and the testing set should be independent to avoid any bias, and (2) the Mobility Markov Chain obtained from the traces of a user is almost always unique to the user.

1.3 The users' point of view

The users' concern about location privacy, and privacy in general, vary a lot from individual to individual, and depend on factors such as age, education, cultural background, etc. They also tend to evolve in time, and cases of privacy breaches that hit the news, like that of "Birds and 'leaky' phone apps" [Bal14], can have a huge impact on the attitude of the population.

There have been several studies to assess people perceptions and attitude towards privacy. We mention in particular the empirical research conducted at CMU by Acquisti and his team, which provides a systematic analysis of several aspects of human behavior in relation to privacy. See [ABL15] for a summary of their findings.

Concerning the specific case of location privacy, the concerns seem in general less strong than for other kinds of sensitive data (such as medical records, financial data, bank information etc.), and the studies give mixed results. For instance, the authors of [FS14] interviewed 180 smartphone users, recruited through social network announcements and through Amazon Mechanical Turk. They chose Mechanical Turk workers who had achieved "master qualification," i.e., those who had shown high competency of performing tasks. They obtained the following statistics: 78% of the participants believed that apps accessing their location can pose privacy threats. Also, 85% of them reported that they care about who accesses their location information (in line with the 87% reported by a similar Microsoft survey [MTC11] two years before). Furthermore, 77% of the users were interested in installing a privacy protection mechanism. Finally, on the specific method based on the addition of random noise, 52% of the surveyed individuals stated no problem in supplying apps with imprecise location information to protect their privacy. Only 18% of the surveyed people objected to supplying apps with imprecise location information.

On the other hand, in contrast with the other kinds of sensitive data mentioned above (medical record etc.) there seem to be more willingness to renounce to location privacy in exchange of compensation. For instance, Danezis et al. [DLA05] conducted a study on 74 undergraduates to find how much money they

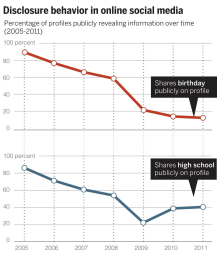


Figure 1: Percentage of members in the CMU Facebook network who chose to publicly reveal personal information (source: [ABL15])

would require in order to share a months worth of their location data. The median price was £10 if the data were to be used for research purposes, and £20 if the data were to be used commercially. In [Kru09] the author says that he could we easily convince over 250 people from his institution to give him two weeks of GPS data recorded in their car in return for a 1% chance of winning a US$ 200 MP3 player. He asked 97 of them if he could share their location data outside our institution, and only 20% said no. In contrast, in an experiment conducted by Acquisti et al. [AJL13] on the privacy attitude towards payments, where people were offered the choice between a traceable gift card of 12 US$ or an anonymous gift card of 10 US$, about half of the people chose the second option.

In conclusion, location data seems to be less critical in the mind of many people than data like financial or medical ones, but this may be due to the lack of knowledge about the negative consequences of a location leak. In particular, about the fact that the location can help profiling the user with respect to more sensitive data. Furthermore, the attitude of people concerning the protection of location information may change during time, along with the general increase of privacy concerns. For example, Figure 1, taken from [ABL15] shows shows that, over time, the percentage of members in the Carnegie Mellon University Facebook network who chose to publicly reveal personal information had decreased dramatically. The increase between 2009 and 2010 is probably due to the fact that Facebook changed the default visibility settings for various fields on its profiles, including high school (bottom), but not birthday (top).

2 Deterministic methods for location privacy

In this section we review the most popular deterministic computational techniques to protect location privacy.

In general, all computational methods for privacy protection are based on degrading the precision of information. In the particular case of location data, this is obtained essentially in two ways: spatial cloaking and spatial obfuscation. Spatial cloaking, first proposed in [GG03], is based on the idea of concealing

the user's exact coordinates by reporting a cloaked area, so to meet certain anonymity constraints. Often, the cloaking is not only spatial, but also temporal, so to conceal also the time in which the user was in that position. The anonymity constraints that have been mostly considered in the case of location privacy are: k-anonymity [Swe02a], l-diversity [MKGV07], t-closeness [LLV07], and p-sensitivity [SSDF08]. In addition, in order to reduce the linkability between identity and trajectories, [BS03] proposed the so-called mix-zones. This idea assumes that people will only report their location in certain regions, called application zones, where a location-based service is offered, e.g. an airport, bank, or coffee shop. In the mix zones, outside the application zones, users will receive new, unused pseudonyms. This helps prevent an attacker from linking pseudonyms, because the new pseudonym could have been assigned to anyone else in the mix zone.

Spatial obfuscation approaches try to preserve privacy by reducing the precision of the position sent from the user to the server. A classic spatial obfuscation approach is the one presented by Ardagna et al. [ACD$^+$07], where a user sends a circular area instead of the precise position. The main advantage of spatial obfuscation approaches is that they do not need a trusted third party, since the user himself can define the obfuscation area and apply the obfuscation mechanism [DSR11].

In the following sections, we review k-anonymity and l-diversity, which can be considered among the most popular deterministic methods for anonymity.

2.1 k-anonymity

Every day, when we execute most of our common daily activities information about us is collected. This large amount of information could be used by adversary because this information easily accessible. For example, in some countries, it is today possible to get access to registers that include the identities of individuals such as voter lists, city directories, and information from motor vehicle agencies, etc.

Typical data like names, birth dates, addresses, telephone numbers, gender contained in databases that could be public can be used for linking identities and re-identified information. In that case, information could be divided into two types: sensitive (such as individuals' names, addresses) and non-sensitive (such as birth dates, gender). Non-sensitive attributes of database that, in combination, can be linked with external information to re-identify the person to whom information refers are called *quasi-identifiers*.

To solve the problem of re-identifying we can use the approach of k-anonymity that was originally proposed by Samarati and Sweeney in the field of database privacy [SS98, Swe02b, Sam01]. By this definition, a database provides k-anonymity if sensitive identifiers are removed from the database and non-sensitive (quasi-identifiers) of each individual in the database are indistinguishable from $k-1$ other individuals. To provide k-anonymity we use an approach based on generalization (obfuscation). Intuitively, attribute values stored in the private table can be substituted, upon release, with generalized values. More simply,

User's id	Location	Query	Time
person 1	49.413521, 21.316322	Grocery store	2016-09-21 15:05:37
person 2	49.417653, 21.316890	Hotel	2016-09-21 15:07:00
person 3	49.413123, 21.316876	Night club	2016-09-21 15:08:11
person 4	49.413098, 21.316485	Gas station	2016-09-21 15:14:52

Table 1: Queries

Cloaked Location	Query	Cloaked Time
49.412 - 49.414, 21.314 - 21.329	Grocery store	2016-09-21 14:40 - 15:10
49.415 - 49.419, 21.315 - 21.331	Hotel	2016-09-21 14:50 - 15:20
49.411 - 49.414, 21.315 - 21.321	Night club	2016-09-21 15:00 - 15:30
49.412 - 49.415, 21.314 - 21.319	Gas station	2016-09-21 15:10 - 15:40

Table 2: Anonymized Queries

it means that we use values from a more general domain, for example postal addresses can be generalized to the street, then to the city, to the county and so on. It then measures the provided privacy with a single parameter k. Privacy protection increases directly with the privacy parameter k.

We now consider the extension of k-anonymity to the field of location privacy. The goal of the adversary is to identify the request (query) that a specific user has issued to a LBS. The purpose of the k-anonymity location obfuscation technique is to make impossible for the adversary to link the identity of a user to her query, and to make it impossible for the adversary to learn the location l of a user i at time t (query anonymity and location privacy).

As mentioned above, the approach of k-anonymity originated from the field of database privacy. In the case of location privacy, we could consider that the entries in the database are the requests send from the users to the LBS. Table 1 illustrates an example of such a database: each row contains the user's identity, the accurate position, the query time and the query.

In order to protect a user's location privacy via k-anonymity, we need to ensure that the user's query be indistinguishable from those of at least $k-1$ other users. To achieve this goal, the identities of these k users are removed from the queries, and their location-time pair is obfuscated to be the same location-area and time-windows, large enough to contain the users' actual locations (Table 2).

A k-anonymity model consisting of mobile users, an LBS, and an anonymity server depicted in Figure 2. The anonymity server is an entity trusted by the

users that mediates the queries between the users and the untrusted LBS. The users send their queries $\langle i, q, l, t \rangle$ to the anonymity server, where i is the id of the user, q is the query, l is the location, and t is the time at which the query is generated. To cloak a query's location the anonymity server removes the identity of the user. Furthermore, it obfuscates the location l and the time t at which the queries were generated. To achieve this aim, the server constructs a cloaking region $R = ([x_1, x_2], [y_1, y_2], [t_1, t_2])$ such that there are at least k users in R whose location $l = (x, y)$ at time t satisfies $x_1 \leq x \leq x_2$, $y_1 \leq y \leq y_2$, and $t_1 \leq t \leq t_2$. $[x_1, x_2]$ and $[y_1, y_2]$ represent a two dimensional area where the subject is located, while $[t_1, t_2]$ represents the time period during which the subject was at this area.The server then sends the anonymized queries to the LBS, and the latter sends back the query responses to the server, which will forward them to the corresponding users.

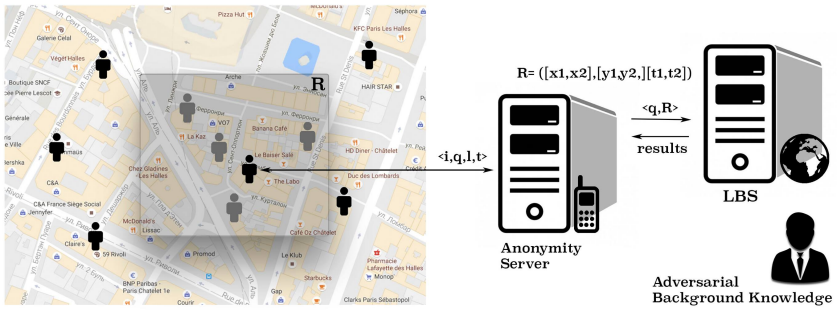


Figure 2: k-anonymity model

Sometimes it may happen that in the area of the user to be protected there are no enough users to form a set of k. In this case, we can use fake users and dummy queries. This technique involves generating $k - 1$ suitably selected dummy queries, and sending these queries to the service provider. "Suitably selected" means that the dummy requests must look likely to be real queries from the point of view of the attacker. Any side information that allows to rule out any of those queries as having low probability of being real, would fail the purpose [ABCP12].

2.1.1 Adaptive-Interval Cloaking Algorithms

Adaptive-interval cloaking algorithm was proposed by Gruteser [GG03]. The idea of this algorithm is that anonimity can be reached by decreasing the accuracy of the revealed spatial data independently from population density. This is achieved by selecting a large area which contains enough amount of subjects inside to satisfy the anonymity constraint.

1.	Initialize the quadrants q and q_{prev} as the total area covered by the anonymizer
2.	Initialize a traffic vector with the current positions of all known vehicles
3.	Initialize p as the position of requestor vehicle
4.	If the number of vehicles in traffic vector $< k_{min}$, then return the previous quadrant q_{prev}
5.	Divide q into quadrants of equal size
6.	Set q_{prev} to q
7.	Set q to the quadrant that includes p
8.	Remove all vehicles outside q from the traffic vector
9.	Repeat from Step 2

Table 3: Adaptive-interval cloaking algorithm. The algorithm computes an area (quadrant) that includes the actual requester and enough potential requesters to satisfy the anonymity constraint k_{min}[GG03].

As input algorithm takes: current position of the requester, the coordinates of the area covered by the anonymity server, and the current positions of all other subjects in the area. Parameter k_{min} is desired degree of anonymity. Algorithm described more detail in Table 3.

But cloaking is not only spatial there is other approach which called temporal cloaking. The idea of this method is to delay the request until k_{min} vehicles have visited the area chosen for the requestor. To achieve this aim spatial cloaking algorithm takes additional parameter - spatial resolution. It then determines the monitoring area by dividing the space until the specified resolution is reached. Time interval [t_1,t_2] is computed, when monitoring area contains k_{min} vehicles, as: the current time is assigned to t_2, and t_1 calculates as the time of request minus a random cloaking factor. The area and the time interval are then returned.

2.2 *l*-diversity

Sometimes it may happen the k users of a k-anonymity group all have the same value for a sensitive attribute. In this case, being indistinguishable from the other members of the group is of no use, because the entire group leaks the sensitive information. To cope with this problem, a stronger notion of privacy has been proposed, called l-diversity. It is a rather subtle concept, and, in order to explain it properly, we need to introduce some technical notions.

Usually an attacker has some background knowledge that can help him to discover sensitive information about the user. In general, we can distinguish the knowledge of the adversary into two kinds [MKGV07]: prior belief and posterior belief. The prior belief is what the adversary knows before exploiting any observation on the system or on the database, and this is what we call adversary's background knowledge. More formally, let S be a sensitive attribute in a tuple t of the original database, Q be a quasi-identifier of the tuple, and P_f the probability derived from the frequency distribution. The prior belief for a value s of S, given a value q of Q, is:

$$\alpha_{(q,s)} = P_f\left(t[S] = s | t[Q] = q\right),$$

where the notation $P(a|b)$ represents the conditional probability of a given b.

Assume now that the database table T is sanitized into a table T^*, which is then published. The posterior belief of the adversary after observing the table T^* is defined as:

$$\beta_{(q,s,T^*)} = P_f\left(t[S] = s | t[Q] = q \wedge \exists t^* \in T^* \wedge t \rightarrow^* t^*\right)$$

where $t \rightarrow^* t^*$ means that the tuple t^* is generated by the sanitization of t.

This posterior-belief can be derived in the way expressed by the following theorem:

Theorem 2.1 ([MKGV07]) *Let q be a value of the non-sensitive attribute Q in the original database T; let q^* be the generalized value of q in the published (sanitized) database T^*; let s be a possible value of the sensitive attribute S; let $n_{(q^*,s')}$ be the number of data tuples $t^* \in T^*$ where $t^*[Q] = q^*$ and $t^*[S] = s'$ and let $f_{(s'|q^*)}$ be the conditional probability of the sensitive value conditioned on the fact that the non-sensitive attribute Q can be generalized to q^*. Then the following relationship holds:*

$$\beta_{(q,s,T^*)} = \frac{n_{(q^*,s)} \frac{f(s|q)}{f(s|q^*)}}{\sum_{s' \in S} n_{(q^*,s')} \frac{f(s'|q)}{f(s'|q^*)}}$$

When talking about privacy there are two different possibilities of revealing sensitive information : positive disclosure and negative disclosure [MKGV07]. Positive disclosure denotes that an adversary can correctly identify the individual's sensitive value with very high probability. The process of correctly eliminating possible sensitive values with very high probability is called negative disclosure.

Definition 2.1 (Positive disclosure)) *Publishing the table T^* that was derived from T results in a positive disclosure if the adversary can correctly identify the value of a sensitive attribute with high probability; i.e., given a $\delta > 0$, there is a positive disclosure if $\beta_{(q,s,T^*)} > 1 - \delta$ and there exists $t \in T$ such that $t[Q] = q$ and $t[S] = s$.*

Definition 2.2 (Negative disclosure) *Publishing the table T^* that was derived from T results in a negative disclosure if the adversary can correctly eliminate some possible values of the sensitive attribute (with high probability); i.e., given a $\varrho > 0$, there is a negative disclosure if $\beta_{(q,s,T^*)} < \varrho$ and there exists a $t \in T$ such that $t[Q] = q$ but $t[S] \neq s$.*

The ideal definition of privacy is that after observing the published table prior and posterior- belief of one adversary should not have big difference from each other. Let us now examine the l-diversity principle that based on theorem 3.1 that allows to calculate attacker's observed belief. The number of occurrences of the sensitive value s, in order to infer positive disclosure,must be much higher than the counts of all other sensitive values. The theorem 3.1 can be rebuilded as:

$$\exists s, \forall s' \neq s, n_{(q^*,s')} \frac{f\,(s'|q)}{f\,(s'|q^*)} \ll n_{(q^*,s)} \frac{f\,(s|q)}{f\,(s|q^*)}$$

This implies that this condition takes place in two cases: lack of diversity and very good background knowledge.

When there is about one sensitive value s in block lack of diversity happens.To make well-represented q^*- block it's important to ensure diversity according to require that a q^*-block has at least $l \geq 2$ different sensitive values and l most frequent values should have roughly the same frequency.

If an adversary knows the frequency distribution $f_{(s'|q)}$ of sensitive values s' in a certain population Ω he will eliminate possible sensitive value of one individual with very high probability - strong background knowledge. For this reasons every q^*-block should have at least l different sensitive attributes. Thus, the following principle is used to define l-Diversity in [MKGV07]:

***l*-Diversity Principle** A q -block is l-diverse if contains at least l 'well-represented' values for the sensitive attribute S. A table is l-diverse if every q^*-block is l-diverse.

To define the 'well-representation' of sensitive attributes there are two instantiations: Entropy and Recursive diversity.

Entropy l-Diversity uses to quantify the uncertainty of possible sensitive values. **Entropy *l*-Diversity** [MKGV07]:

A table is Entropy l-diverse if for every q^*-block

$$H_l = -\sum_{s \in\ S} p_{(q^*,s)log\left(p_{(q^*,s)}\right)} \geq log\,(l)$$

where $p_{(q^*,s)} = \frac{n_{(q^*,s)}}{\sum_{s' \epsilon S} n_{(q^*,s')}}$ is thr fraction of tuples in the q^*-block with sensitive attribute equal to s.

From this notion follows that every q^* - block has at least l different values for sensitive attribute, also as a consequence the higher is value of H_l, the more pieces of information are needed to infer positive disclosure. The minimal value of entropy $H_l = 0$ is achieved when $p_{(q^*,s)} = 1$ and $p_{(q^*,s')} = 0$, $\forall s' \in S$, $s' \neq s$ and means that there is no information needed to determine the possible

sensitive value because there is only one sensitive value in the q^*-block. The maximal value of entropy $H_l = log\,(l)$ is only achieved if $p_{(q^*,s')}$is equal for at least l existing sensitive values s' in the block, the entropy of the whole table must be greater or equal *log(l).*

Obviously Entropy l-Diversity is hard to achieve. For example, if 85% of the sensitive value is common Entropy l-Diversity cannot be satisfied by such a table because the probability of sensitive value is too high compared to all other sensitive values. For this reason, there is another definition which used to solve this problem, which is called Recursive *(c,l)*-Diversity.

Let $s_1, ..., s_m$ be the possible values of the sensitive attribute S in a q^*-block. Let $n_{(q^*,s_i)}$ be their frequencies which are put into the set of the overall frequencies $r_1, ..., r_m$ sorted in descending order.

The idea is that the adversary needs to eliminate l-1 different sensitive values S to gain positive disclosure. So, in order to prevent it, the most frequent sensitive value should not exist too often in a table. This is satisfied, if the following definition (introduced in [MKGV07]) holds:

Definition 4.2 (Recursive *(c,l)*-Diversity) In a given q^*- block, let r_i denote the number of times the i^{th} most frequent sensitive value appears in that q^*-block. Given a constant c, the q^*-block satisfies recursive *(c,l)*-diversity if $r_1 < c(r_l + r_{l+1} + \ + r_m)$. A table T^* satisfies recursive *(c,l)*-diversity if every q^*-block satisfies recursive l-diversity. We say that 1-diversity is always satisfied.

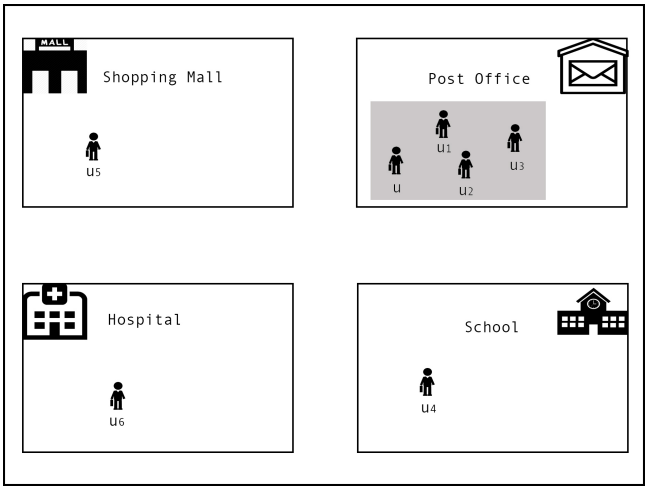
It means, that if any sensitive value s' is eliminated by an adversary within a *(c,l)*-diverse q^*-block, then the remaining block (without s' containing tuples) has to be at least *(c,l-1)*-diverse. Here c is a constant which shows how often the most frequent sensitive value may occur in relation to the total amount of the other sensitive attribute values. This constant defined manually by the user.

2.2.1 Location *l*-diversity

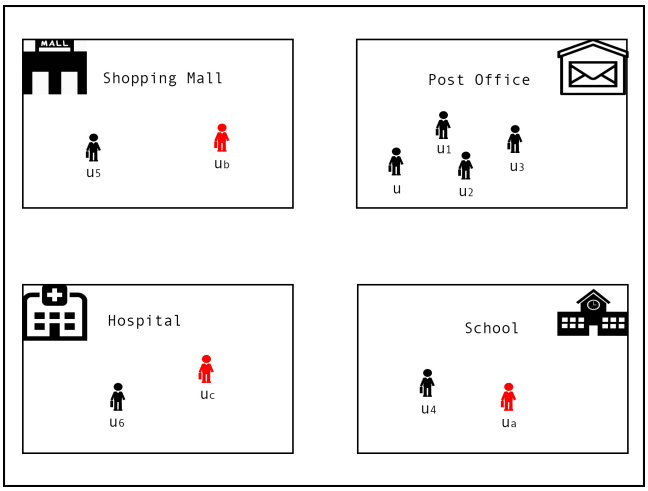
In this section we consider l-diversity in the context of Location Privacy Protection. It's important to understand that location is related with semantic information such as shop, school, hospital, restaurant, etc, and typically this region contains users with similar contextual information, such as age, gender, hobby, etc. It means, that for each location query message, in addition to user level k- anonymity (k different user identities at the same location), there are at least l different still location objects associated with each of the k users. For example:

Example 1. Couriers that work at large post office with delivery service plan their way to clients use LBS (BestMaps, Google maps, etc.). To avoid identification queries are issued through anonymity server which performs query anonymization and sends the anonymized queries to the LBS (k-anonymity). Figure 2.a shows a map of current users, where u is querying user. Cloaking region contain 4 *users* { *u, u_1, u_2, u_3*} *(k = 4), but these* 4 *users are Post Office worker's and adversary knows location of user.*

Therefore was proposed *Location Diversity* [XKP09] which guarantees that



(a) k-anonymity



(b) Location diversity

Figure 3: Approaches comparison

Location	Query
Location 1	Grocery store
Location 1	School
Location 2	School
Location 2	Shopping mall
Location 3	Grocery store
Location 3	Shopping mall

Table 4: l-diverse queries

each query can be associated with at least l semantically different locations, it shown at Table 4. In Figure 2.b shown 4-l-diverse query { u, u_a, u_b, u_c} , the semantic location of these users are Post Office, school, hospital and shopping mall. So, the adversary can identify where query comes from with probability $1/l$.

Each semantic location (school, shopping mall, hospital, its.) represents a large spatial region. When user sends query the anonymity server selects fake users from other regions u_a, u_b, u_c. After all these locations, real and fake, { u, u_a, u_b, u_c} sent to the LBS. From the LBS's point of view it's four different queries which executes and returns all results to the anonymity server. Next, anonymity server filters out the false positives and returns the actual result to u.

2.3 Mix zones

To provide unlinkability between identity and trajectories [BS03] proposed mix-zones. The idea is in frequently changing pseudonyms to unused one, it means, that each time, when user enter mix-zone new pseudonym assigned. The anonymity is measured in terms of the unlinkability between the old and new pseudonyms. Also, when users are inside mix-zone they do not send any location information to LBS, so identities are "mixed". Because of it, users going into mix-zone are indistinguishable from those who coming out of it or who is in the mix zone at same time. Figure 3 shows users u_1, u_2, u_3 that enter the mix-zone with pseudonyms x,y,z and exit the mix-zone with new pseudonyms s,q,h. Thus, adversary can not make link between their entry and exit order. A sat of users U in a mix-zone Z is said to be k-anonymized iff [PL11]

1. The set U contains at least k users.
2. All users in U are in Z at a point in time, i.e., all users in U must enter Z before any user in U exits.

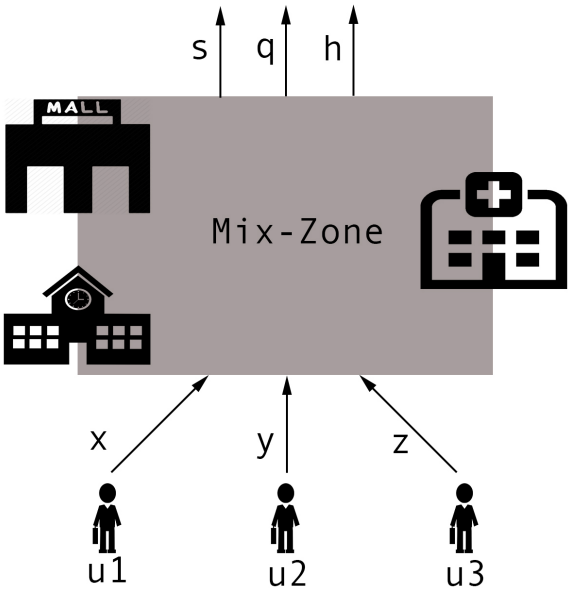


Figure 4: A mix-zone with three users

3. Each user in U spends a completely random duration of time inside Z.

4. The probability of every user in U entering through an entry point is equally likely to exit in any of the exit points.

In a road network mix-zones match with a road intersection. But vehicle movements are depend on many factors, such as travel speed, road conditions, traffic conditions, the number of road segments at the intersection and physical roads. This implies that users cannot be inside a mix- zone for a random amount of time and adversary can get background information by linking entering events and exiting events. For example, Figure 4.a provides a plan view of a mix zone that is placed on four roads segments $Seg1$,$Seg2$,$Seg3$ and $Seg4$. An adversary knows that a vehicle with pseudonym c enters the mix-zone from either $Seg1_{in}$ or $Seg2_{in}$ or $Seg4_{in}$. Zone $Seg3_{out}$ is closer to $Seg2_{in}$ than $Seg1_{in}$ or $Seg4_{in}$, so the adversary would use this information to link events at road segments ($Seg3_{out}$ with $Seg1_{in}$ or $Seg2_{in}$ or$Seg4_{in}$).

More effective way to construct mix-zones is shown in Figure 4.b, the idea is to construct non-rectangular, adaptive mix-zones that start from the centre of the crossroad on the outgoing road segments [PL11]. The length of the mix-zone along each outgoing segment is depends on the average speed of the road segment, the minimum pairwise entropy threshold and the size of the chosen time window. The pair-wise entropy is computed for every pair of users a and

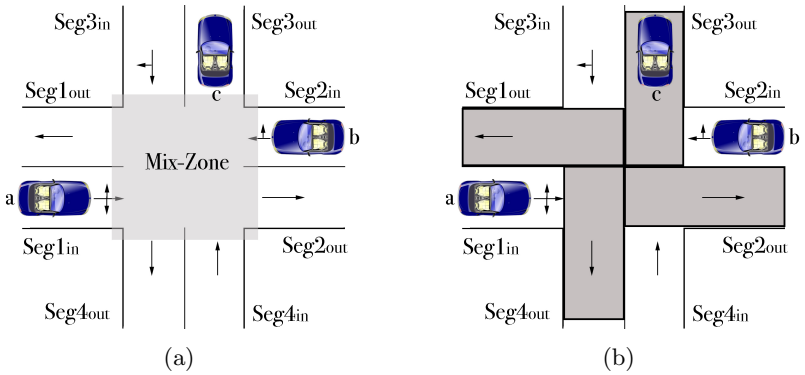


Figure 5: A vehicular mix-zone(a) and non-rectangular, adaptive vehicular mix-zones (b)

b in an anonymity set U by considering a and b to be the only members in U and determining the linkability between their old and new pseudonyms [CM11].

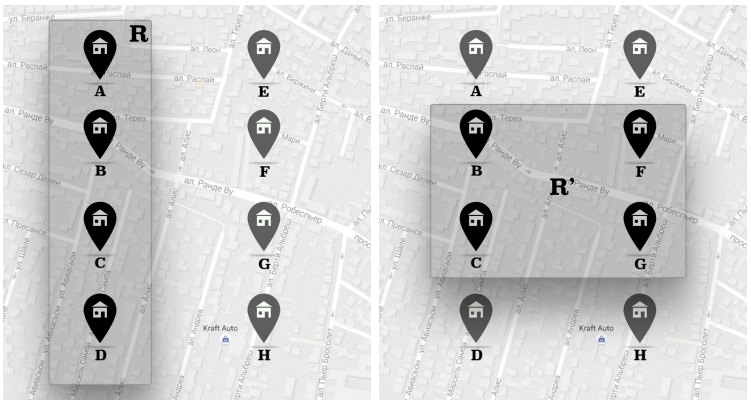
2.4 Criticism of the cloaking method

We can define a cloaking method as a generated by system region for a given user's request that satisfy k-anonymity. As known, in k-anonymity, the generated cloaking region contains a user who must be indistinguishable from k-1 other users. But in the case if the cloaking region is generated for a single building, the behavior of the user can be disclosed easily to adversaries, for example, when users are situated near each other in a not so big place (e.g., a caffe). Or the opposite situation, when cloaking region is large and encompasses the same number of users. It is obvious that these users are better protected because the adversary has more uncertainty about their exact locations, consequently location privacy is high. It's clearly, that the number of users who are situated in the obfuscated region is not a consistent metric for location privacy. The independence of the value of k and the accuracy of a user's location estimation by the adversary, implies that k is irrelevant to their actual location privacy [STD+10].

We consider a security analysis from the adversary's point of view in the case when he has background information.

First, when adversary, for example, using multiple directional antennas could eavesdrop on the communications between users and anonymity server he gets user's location. Of course, he cannot know whose this query is, because a query contains information which looks like $\langle q, R \rangle$. Thus, the query is k-anonymous, but the adversary knows the users' location so in this case they have no location privacy.

Second, when adversary knows statistical information about users? Mobility, such as their places of work and homes it is easy to understand that users will be at home at night, and at their work places during day. All resources of adversary



(a) (b)

Figure 6: Statistical background information: efficiency (a) and additional information (b)

is user's query $\langle q, R\rangle$ and his background knowledge. Thus, computing cloaking regions ineffectually.

Example 1 [STD$^+$10]: Let us consider a neighborhood as the one shown in Fig. 5(a), and assume that the adversary knows that with a high probability all users are at home (for instance, late in the evening). When user A sends a query $\langle q, R\rangle$ to the LBS, it is unaware of the current location of users, and can only use the available statistical information to infer who/where is the sender of q. Thus, user A is 4-anonymous independently of whether or not B, C, and D are currently using the system, or even present at their home locations.

Conclusion : it's not necessary for anonymity server to make difficult calculations to select region R. Using available to the adversary background information regions R can be pre-computed and after be selected by users uniquely based on their own location.

Example 2 [STD$^+$10]: in Fig. 5(b) where only users A, B, E, and F are active (i.e., using the system), and assume that the adversary has the same information as in the previous case. When user A sends a query, the anonymity server forwards $\langle q, R'\rangle$ to the LBS. Upon receiving this information, the adversary learns that A, B, E, and F are currently in their home locations, and that C and D are either inactive or absent. This is because had C and D been active in the system, the minimal region sent to the LBS would have been R (as in Fig. 2(a)). Thus, the only configuration that results in R' is that A, B, E, and F areactive and at home. In this case, although the query q is still 4-anonymous, A, B, E, and F have no location privacy, due to the information revealed to the adversary by the cloaking region itself.

Conclusion : when we want to use minimization of the region R it allows the adversary to make inferences about their current position .

Third, even when adversary knows nothing except $\langle q, R\rangle$ constructing cloak-

ing regions based on the k-anonymity technique does not provide any additional protection. Furthermore, reducing the performance of the system in terms of accuracy and computational load happens.

3 Randomized methods for location privacy

In this section we give an overview of the randomized methods for location privacy.

Like in the case of the deterministic methods, they can classified into two groups: those that protect the identity of the users, and those which aim at obfuscating the position of the user. In both these cases, the most investigated random mechanisms are those based on the popular notion of differential privacy [Dwo06]. The reasons of the success of of this notion are essentially the fact that it is much more robust to composition attacks than other notions (especially the deterministic ones), and that it does not depend on the side knowledge of the adversary. These advantages are preserved also in the case of its application in the field of location privacy.

3.1 Differential Privacy

Differential Privacy [Dwo06] is a notion of privacy from the area of statistical databases. Its goal is to protect an individual's data while publishing aggregate information about the database. Differential privacy requires that modifying a single user's data should have a negligible effect on the query outcome. More precisely, it requires that the probability that a query returns a value v when applied to a database D, compared to the probability to report the same value when applied to an *adjacent* database D' – meaning that D, D' differ in the value of a single individual – should be within a bound of e^ϵ (where ϵ is a parameter quantifying the level of privacy).

A typical way to achieve this notion is to add controlled random noise to the query output, for example drawn from a Laplace distribution.

As already mentioned, the advantage of this notion is that it does not depend on the prior (attacher's side information), hence a differentially private mechanism can be designed without making any assumption about the knowledge of the adversary.

Even more important, by using the Bayes theorem the above definition can be shown equivalent to a definition which binds the ration of the prior and posterior information from above by e^ϵ, which means that when combining n ϵ-differentially private mechanisms, the level of privacy decreases linearly with n, which is a very good performance with respect to all other known methods.

3.2 Protection of identity

Most of the works that have used differential privacy in the context of location privacy have considered a scenario where *aggregate* information about several users is published. In such situation, differential privacy can be applied just like in the case of databases. For instance, Machanavajjhala et al. [MKA$^+$08]

used a synthetic data generation technique to publish statistical information about commuting patterns in a differentially private way. Ho et al. [HR11] used a spatial decomposition technique to ensure differential privacy in a database with location pattern mining capabilities. Chen et al. used variable-length n-grams to disclose sequential data, such as mobility traces, in a differentially private way.

3.3 Protection of location

We consider now the case in which we want to obfuscate the location of a single user.

Differential privacy as it is is not too suitable for this scenario, because it would require that any change in that location should have negligible effect on the published output, making it impossible to communicate any useful information to the service provider.

To overcome this issue, Dewri [2012] proposes a mix of differential privacy and k-anonymity, by fixing an anonymity set of k locations and requiring that the probability to report the same obfuscated location z from any of these k locations should be similar (up to e^ϵ). This property is achieved by adding Laplace noise to each Cartesian coordinate independently. There are however two problems with this definition: first, the choice of the anonymity set crucially affects the resulting privacy; outside this set no privacy is guaranteed at all. Second, the property itself is rather weak; reporting the geometric median (or any deterministic function) of the k locations would satisfy the same definition, although the privacy guarantee would be substantially lower than using Laplace noise.

Nevertheless, Dewri's intuition of using Laplace noisefor location privacy is valid, and [Dew12] provides extensive experimental analysis supporting this claim.

3.3.1 Geo-indistinguishability

In this section we formalize the notion of geo-indistinguishability. The main idea behind this notion is that, for any radius $r > 0$, the user enjoys ϵr-privacy within r, i.e. the level of privacy is proportional to the radius. Note that the parameter ϵ corresponds to the level of privacy at one unit of distance. For the user, a simple way to specify his privacy requirements is by a tuple (ℓ, r), where r is the radius he is mostly concerned with and ℓ is the privacy level he wishes *for that radius*. In this case, it is sufficient to require ϵ-geo-indistinguishability for $\epsilon = \ell/r$; this will ensure a level of privacy ℓ within r, and a proportionally selected level for all other radii.

In the remaining of this section we give a formal definition of ℓ-privacy, as well as two characterizations which clarify the privacy guarantees provided by geo-indistinguishability.

Probabilistic model. We first introduce a simple model used in the rest of the paper. We start with a set $\mathcal{X}$ of *points of interest*, typically the user's possible locations. Moreover, let $\mathcal{Z}$ be a set of possible *reported values*, which in general can be arbitrary, allowing to report obfuscated locations, cloaking regions, sets of locations, etc. However, to simplify the discussion, we sometimes consider $\mathcal{Z}$ to also contain spatial points, assuming an operational scenario of a user located at $x \in \mathcal{X}$ and communicating to the attacker a randomly selected location $z \in \mathcal{Z}$ (e.g. an obfuscated point).

Probabilities come into place in two ways. First, the attacker might have side information about the user's location, knowing, for example, that he is likely to be visiting the Eiffel Tower, while unlikely to be swimming in the Seine river. The attacker's side information can be modeled by a *prior* distribution π on $\mathcal{X}$, where $\pi(x)$ is the probability assigned to the location x.

Second, the selection of a reported value in $\mathcal{Z}$ is itself probabilistic; for instance, z can be obtained by adding random noise to the actual location x. A *mechanism* K is a probabilistic function for selecting a reported value; i.e. K is a function assigning to each location $x \in \mathcal{X}$ a probability distribution on $\mathcal{Z}$, where $K(x)(Z)$ is the probability that the reported point belongs to the set $Z \subseteq \mathcal{Z}$, when the user's location is x.[1] Starting from π and using Bayes' rule, each observation $Z \subseteq \mathcal{Z}$ of a mechanism K induces a *posterior* distribution $\sigma = \mathbf{Bayes}(\pi, K, Z)$ on $\mathcal{X}$, defined as $\sigma(x) = \frac{K(x)(Z)\pi(x)}{\sum_{x'} K(x')(Z)\pi(x')}$.

We define the *multiplicative distance* between two distributions σ_1, σ_2 on some set $\mathcal{S}$ as $d_{\mathcal{P}}(\sigma_1, \sigma_2) = \sup_{S \subseteq \mathcal{S}} |\ln \frac{\sigma_1(S)}{\sigma_2(S)}|$, with the convention that $|\ln \frac{\sigma_1(S)}{\sigma_2(S)}| = 0$ if both $\sigma_1(S), \sigma_2(S)$ are zero and ∞ if only one of them is zero.

3.3.2 Definition

We are now ready to state our definition of geo-indistinguishability. Intuitively, a privacy requirement is a constraint on the distributions $K(x), K(x')$ produced by two different points x, x'. Let $d_2(\cdot, \cdot)$ denote the Euclidean metric. Enjoying ℓ-privacy within r means that for any x, x' s.t. $d_2(x, x') \leq r$, the distance $d_{\mathcal{P}}(K(x), K(x'))$ between the corresponding distributions should be at most ℓ. Then, requiring ϵr-privacy for all radii r, forces the two distributions to be similar for locations close to each other, while relaxing the constraint for those far away from each other, allowing a service provider to distinguish points in Paris from those in London.

Definition 3.1 (geo-indistinguishability) *A mechanism K satisfies ϵ-geo-indistinguishability iff for all x, x':*

$$d_{\mathcal{P}}(K(x), K(x')) \leq \epsilon d_2(x, x')$$

[1] For simplicity we assume distributions on $\mathcal{X}$ to be discrete, but allow those on $\mathcal{Z}$ to be continuous (c.f. Section 3.3.4). All sets to which probability is assigned are implicitly assumed to be measurable.

Equivalently, the definition can be formulated as $K(x)(Z) \leq e^{\epsilon d_2(x,x')} K(x')(Z)$ for all $x, x' \in \mathcal{X}, Z \subseteq \mathcal{Z}$. Note that for all points x' within a radius r from x, the definition forces the corresponding distributions to be at most ϵr distant.

The quantity $\epsilon d_2(x, x')$ can be viewed as the *distinguishability level* between the secrets x and x'. The use of the Euclidean metric d_2 is natural for location privacy: the *closer* (geographically) two points are, the *less distinguishable* we would like them to be. Note, however, that other metrics could be used instead of d_2, such as the Manhattan metric or driving distance, depending on the application. The definition that we obtain by using an arbitrary distinguishability metric $d_{\mathcal{X}}$, i.e. requiring that $d_{\mathcal{P}}(K(x), K(x')) \leq d_{\mathcal{X}}(x, x')$, is referred to as $d_{\mathcal{X}}$-privacy[2], and is studied on its own right in [CABP13]. Some of the results of this paper do not depend on the actual metric, so they are given in the general framework of $d_{\mathcal{X}}$-privacy.

Note also that standard differential privacy simply corresponds to $\epsilon d_h(x, x')$-privacy, where d_h is the Hamming distance between databases x, x', i.e. the number of individuals in which they differ. However, in our scenario, using the Hamming metric of standard differential privacy – which aims at completely protecting the value of an individual – would be too strong, since the only information is the location of a single individual. Nevertheless, we are not interested in completely hiding the user's location, since some approximate information needs to be revealed in order to obtain the required service. Hence, using a privacy level that depends on the Euclidean distance between locations is a natural choice.

Protecting location traces. So far, we have assumed a *sporadic* use of an LBS, meaning that the service is used infrequently enough that we can assume no correlation between different uses and treat each one of them independently. In this case, the user's secret is a single location. In the case of *repeated* use, however, the user forms a *location trace* which should be protected; the provider is allowed to obtain only approximate information about the locations, their exact value should be kept private.

In this case, the secret is the trace, i.e. a tuple of points denoted by $\mathbf{x} = [x_1, \ldots, x_n]$, while $\mathbf{x}[i]$ denotes the i-th element of the trace. The notion of ϵ-geo-indistinguishability extends naturally by defining the distance between two tuples $\mathbf{x}, \mathbf{x}'$ as:

$$d_\infty(\mathbf{x}, \mathbf{x}') = \max_i d_2(\mathbf{x}[i], \mathbf{x}'[i])$$

and using ϵd_∞-privacy as our privacy definition. Following the idea of reasoning within a radius r, this definition requires that two traces at most r away from each other (i.e. such that $\mathbf{x}[i], \mathbf{x}'[i]$ are all within distance r from each other) should produce distributions at most ϵr apart.

[2]Note that we can generally consider the scaling factor ϵ to be part of the metric, although sometimes we emphasize it by talking of $\epsilon d_{\mathcal{X}}$-privacy

3.3.3 Characterizations

In this section we state two characterizations of geo-indistinguishability, obtained from the corresponding results of [CABP13] (for general metrics), which provide intuitive interpretations of the privacy guarantees offered by this notion.

Adversary's conclusions under hiding. The first characterization uses the concept of a *hiding function* $\phi : \mathcal{X} \to \mathcal{X}$. The idea is that ϕ can be applied to the user's actual location before the mechanism K, so that the latter has only access to a hidden version $\phi(x)$, instead of the real location x. A mechanism K with hiding applied is simply the composition $K \circ \phi$. Intuitively, a location remains private if, regardless of his side knowledge (captured by his prior distribution), an adversary draws the same conclusions (captured by his posterior distribution), regardless of whether hiding has been applied or not. However, if ϕ replaces locations in Paris with those in London, then clearly the adversary's conclusions will be greatly affected. Hence, we require that the effect on the conclusions depends on the maximum distance $d_2(\phi) = \sup_{x \in \mathcal{X}} d_2(x, \phi(x))$ between the real and hidden location.

Theorem 3.1 *A mechanism K satisfies ϵ-geo-indistinguishability iff for all $\phi : \mathcal{X} \to \mathcal{X}$, all priors π on $\mathcal{X}$, and all $Z \subseteq \mathcal{Z}$:*

$$d_{\mathcal{P}}(\sigma_1, \sigma_2) \leq 2\epsilon d_2(\phi) \qquad \textit{where} \qquad \sigma_1 = \mathbf{Bayes}(\pi, K, Z) \quad \sigma_2 = \mathbf{Bayes}(\pi, K \circ \phi, Z)$$

Note that this is a natural adaptation of a well-known interpretation of standard differential privacy, stating that the attacker's conclusions are similar, regardless of his side knowledge, and regardless of whether an individual's real value has been used in the query or not. This corresponds to a hiding function ϕ removing the value of an individual.

Note also that the above characterization compares two *posterior* distributions. Both σ_1, σ_2 can be substantially different than the initial knowledge π, which means that an adversary does learn some information about the user's location.

Knowledge of an informed attacker. A different approach is to measure how much the adversary learns about the user's location, by comparing his prior and posterior distributions. However, since some information is allowed to be revealed by design, these distributions can be far apart. Still, we can consider an *informed* adversary who already knows that the user is located within a set $N \subseteq \mathcal{X}$. Let $d_2(N) = \sup_{x,x' \in N} d_2(x, x')$ be the maximum distance between points in x. Intuitively, the user's location remains private if, regardless of his prior knowledge within N, the knowledge obtained by such an informed adversary should be limited by a factor depending on $d_2(N)$. This means that if $d_2(N)$ is small, i.e. the adversary already knows the location with some accuracy, then the information that he obtains is also small, meaning that he cannot improve

his accuracy. Denoting by $\pi_{|N}$ the distribution obtained from π by restricting to N (i.e. $\pi_{|N}(x) = \pi(x|N)$), we obtain the following characterization:

Theorem 3.2 *A mechanism K satisfies ϵ-geo-indistinguishability iff for all $N \subseteq \mathcal{X}$, all priors π on $\mathcal{X}$, and all $Z \subseteq \mathcal{Z}$:*

$$d_{\mathcal{P}}(\pi_{|N}, \sigma_{|N}) \leq \epsilon d_2(N) \qquad \textit{where} \qquad \sigma = \mathbf{Bayes}(\pi, K, Z)$$

Note that this is a natural adaptation of a well-known interpretation of standard differential privacy, stating that an informed adversary who already knows all values except individual's i, gains no extra knowledge from the reported answer, regardless of side knowledge about i's value [DMNS06].

Abstracting from side information. A major difference of geo-indistinguishability, compared to similar approaches from the literature, is that it abstracts from the side information available to the adversary, i.e. from the prior distribution. This is a subtle issue, and often a source of confusion, thus we would like to clarify what "abstracting from the prior" means. The goal of a privacy definition is to restrict the information *leakage* caused by the observation. Note that the lack of leakage does not mean that the user's location cannot be inferred (it could be inferred by the prior alone), but instead that the adversary's knowledge does not increase *due to the observation*.

However, in the context of LBSs, no privacy definition can ensure a small leakage under any prior, and at the same time allow reasonable utility. Consider, for instance, an attacker who knows that the user is located at some airport, but not which one. The attacker's prior knowledge is very limited, still any useful LBS query should reveal at least the user's city, from which the exact location (i.e. the city's airport) can be inferred. Clearly, due to the side information, the leakage caused by the observation is high.

So, since we cannot eliminate leakage under any prior, how can we give a reasonable privacy definition without restricting to a particular one? First, we give a formulation (Definition 3.1) which does not involve the prior at all, allowing to verify it without knowing the prior. At the same time, we give two characterizations which explicitly quantify over all priors, shedding light on how the prior affects the privacy guarantees.

3.3.4 Mechanisms for the sporadic case

In this section we present two mechanisms for applying noise to a single location while satisfying geo-indistinguishability. The first one, the *planar Laplace mechanism*, is a simple and efficient mechanism that scales to any number of possible locations while being generic and independent from the user's behaviour. The second is adapted to a specific user and guarantees *optimal utility* (or minimum quality loss) for that user, however it is only applicable when the number of possible locations is limited.

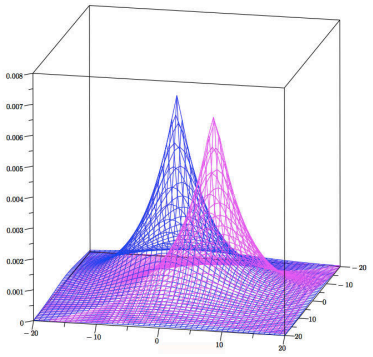


Figure 7: The pdf of two planar Laplace distributions, centered at $(-2, -4)$ and at $(5, 3)$ respectively, with $\epsilon = 1/5$.

3.3.5 The planar Laplace mechanism

We start by defining a mechanism for geo-indistinguishability on the continuous plane. The idea is that whenever the actual location is $x \in \mathbb{R}^2$, we report, instead, a point $z \in \mathbb{R}^2$ generated randomly according to a distribution with probability density function:

$$D_\epsilon(z) = \frac{\epsilon^2}{2\,\pi}\, e^{-\epsilon\, d_2(x,z)} \tag{1}$$

This function is called the *planar Laplace centered at* x and is is illustrated in Figure 7. The resulting mechanism can be shown to satisfy ϵ-geo-indistinguishability [ABCP13].

Note that this definition of the two-dimensional Laplace distribution follows [LS93] and is different than generating the two coordinates independently from a standard (one dimensional) Laplace distribution. Such an approach would not, in fact, satisfy geo-indistinguishability.

Drawing a random point. We illustrate now how to draw a random point from the pdf defined in (1). First of all, we note that the pdf of the planar Laplace distribution depends only on the distance from x. It will be convenient, therefore, to switch to a system of polar coordinates with origin x. A point z will be represented as a point (r, θ), where r is the distance of z from x, and θ is the angle that the line $x\,z$ forms with respect to the horizontal axis of the Cartesian system. After the transformation, the pdf of the *polar Laplace* centered at the origin x is:

$$D_\epsilon(r, \theta) = \frac{\epsilon^2}{2\,\pi}\, r\, e^{-\epsilon\, r} \tag{2}$$

Let R, Θ be the random variables representing the radius and the angle; the property that allows to efficiently draw from the polar Laplace is that the two

variables are *independent*, that is $D_\epsilon(r,\theta)$ is the product of the two marginals:

$$D_{\epsilon,R}(r) \;=\; \int_0^{2\pi} D_\epsilon(r,\theta)\,d\theta \;=\; \epsilon^2\, r\, e^{-\epsilon\, r}$$

$$D_{\epsilon,\Theta}(\theta) \;=\; \int_0^{\infty} D_\epsilon(r,\theta)\,dr \;=\; \tfrac{1}{2\,\pi}$$

Note that $D_{\epsilon,R}(r)$ corresponds to the *gamma distribution* with shape 2 and scale $1/\epsilon$.

Hence, in order to draw a point (r,θ) it is sufficient to draw separately r and θ from $D_{\epsilon,R}(r)$ and $D_{\epsilon,\Theta}(\theta)$ respectively. Since $D_{\epsilon,\Theta}(\theta)$ is constant, θ can be drawn from a uniform distribution on the interval $[0, 2\pi)$.

We now show how to draw r. Following standard lines, we consider the cumulative distribution function (cdf) $C_\epsilon(r)$:

$$C_\epsilon(r) \;=\; \int_0^r D_{\epsilon,R}(\rho)d\rho \;=\; 1 - (1+\epsilon\, r)\, e^{-\epsilon\, r}$$

Intuitively, $C_\epsilon(r)$ represents the probability that the radius of the random point falls between 0 and r. Finally, we generate a random number p with uniform probability in the interval $[0, 1)$, and we set $r = C_\epsilon^{-1}(p)$. Note that

$$C_\epsilon^{-1}(p) = -\tfrac{1}{\epsilon}\big(W_{-1}(\tfrac{p-1}{e}) + 1\big)$$

where W_{-1} is the Lambert W function (the -1 branch), which can be computed efficiently and is implemented in several numerical libraries.

Note that in practice only a discretized version of the continuous mechanism can be implemented; the discretized variant can be shown to also satisfy geo-indistinguishability, for a slightly bigger ϵ, although the difference is negligible on a double precision machine. A detailed discussion of discretization issues can be found in [ABCP13].

The planar Laplace mechanism has two main advantages: first, it is simple and efficient to compute without restricting the number of possible locations. Second, it can be applied to a generic user without prior information on his behaviour. The usefulness of the mechanism for generic applications is showcased in *Location Guard* [`https://github.com/chatziko/location-guard`], a browser extension for Chrome and Firefox, which provides location privacy for websites accessing the user's location through the HTML5 geolocation API, by adding noise to the reported location using the planar Laplace mechanism.

On the other hand, being generic, the planar Laplace mechanism offers no optimality guarantees for the quality loss of the reported location. In the following section, we show how to improve utility by construct mechanisms adapted to the behaviour of a particular user.

3.3.6 Geo-indistinguishable mechanisms of optimal utility

The goal of a privacy mechanism is not to hide completely the secret but to disclose enough information to be useful for some service while hiding the rest

to protect the user's privacy. Typically these two requirements go in opposite directions: a stronger privacy level requires more noise which results in a lower utility.

From the user's point of view, we want to quantify the service *quality loss (QL)* produced by the mechanism K. Given a *quality metric* d_Q on locations, such that $d_Q(x, z)$ measures how much the quality decreases by reporting z when the real location is x (the Euclidean metric d_2 being a typical choice), we can naturally define the quality loss as the expected distance between the real and the reported location, that is

$$\mathrm{QL}(K, \pi, d_Q) = \sum_{x,z} \pi(x)K(x)(z)d_Q(x, z)$$

where π is a prior on $\mathcal{X}$ modeling the user's behaviour.

Despite the generality of the planar Laplace mechanism, in some cases we want to be able to build a mechanism that optimizes the trade-off between privacy (in terms of geo-indistinguishability) and quality loss (in terms of QL) for a specific *user*. Our main goal is, given a set of locations $\mathcal{X}$ with a privacy metric $d_\mathcal{X}$, a privacy level ϵ, a user profile π and a quality metric d_Q, to find an $\epsilon d_\mathcal{X}$-private mechanism such that its QL is as small as possible. We start by describing a set of linear constraints that enforce $\epsilon d_\mathcal{X}$-privacy, which allows to obtain an optimal mechanism as a linear optimization problem. However, the number of constraints can be large, making the approach computationally demanding as the number of locations increases. As a consequence, we then propose an approximate solution that replaces $d_\mathcal{X}$ with the metric induced by a spanning graph.

Constructing an optimal mechanism. The constructed mechanism is assumed to have as both input and output a predetermined finite set of locations $\mathcal{X}$. For instance, $\mathcal{X}$ can be constructed by dividing the map in a finite number of regions (of arbitrary size and shape), and selecting in $\mathcal{X}$ a representative location for each region. We also assume a prior π over $\mathcal{X}$, representing the probability of the user being at each location at any given time. Since $\mathcal{X}$ is finite, a mechanism K can be represented by a stochastic matrix, where k_{xz} is the probability to report z from location x.

Given a privacy metric $d_\mathcal{X}$ and a privacy parameter ϵ, the goal is to construct a $\epsilon d_\mathcal{X}$-private mechanism K such that the *service quality loss* with respect to a quality metric d_Q is minimum. This property is formally defined below:

Definition 3.2 *Given a prior π, a privacy metric $d_\mathcal{X}$, a privacy parameter ϵ and a quality metric d_Q, a mechanism K is $\epsilon d_\mathcal{X}$-*OptQL(π, d_Q) *iff:*

1. *K is $\epsilon d_\mathcal{X}$-private, and*
2. *for all mechanisms K', if K' is $\epsilon d_\mathcal{X}$-private then* $\mathrm{QL}(K, \pi, d_Q) \leq \mathrm{QL}(K', \pi, d_Q)$

In order for K to be $\epsilon d_\mathcal{X}$-private it should satisfy the following constraints:

$$k_{xz} \leq e^{\epsilon d_\mathcal{X}(x,x')} k_{x'z} \qquad x, x', z \in \mathcal{X}$$

Hence, we can construct an optimal mechanism by solving a linear optimization problem, minimizing $\mathrm{QL}(K, \pi, d_Q)$ while satisfying $\epsilon d_{\mathcal{X}}$-privacy:

$$\begin{aligned}
\textbf{Minimize:} \quad & \sum_{x,z \in \mathcal{X}} \pi_x k_{xz} d_Q(x,z) & \\
\textbf{Subject to:} \quad & k_{xz} \leq e^{\epsilon d_{\mathcal{X}}(x,x')} k_{x'z} & x, x', z \in \mathcal{X} \\
& \sum_{z \in \mathcal{X}} k_{xz} = 1 & x \in \mathcal{X} \\
& k_{xz} \geq 0 & x, z \in \mathcal{X}
\end{aligned}$$

It is easy to see that the mechanism K generated by the previous optimization problem is $\epsilon d_{\mathcal{X}}$-$\textsc{OptQL}(\pi, d_Q)$.

A more efficient method using spanners. In the optimization problem of the previous section, the $\epsilon d_{\mathcal{X}}$-privacy definition introduces $|\mathcal{X}|^3$ constraints in the linear program. However, in order to be able to manage a large number of locations, we would like to reduce this amount to a number in the order of $O(|\mathcal{X}|^2)$.

So far we are not making any assumption about $d_{\mathcal{X}}$, and therefore we need to specify $|\mathcal{X}|$ constraints for each pair of locations x and x'. However, it is worth noting that if the distance $d_{\mathcal{X}}$ is induced by a weighted graph (i.e. the distance between each pair of locations is the weight of a minimum path in a graph), then we only need to consider $|\mathcal{X}|$ constraints for each pair of locations that are *adjacent in the graph*.

It might be the case, though, that the metric $d_{\mathcal{X}}$ is not induced by any graph (other than the complete graph), and consequently the amount of constraints remains the same. In fact, this is generally the case for the Euclidean metric. Therefore, we consider the case in which $d_{\mathcal{X}}$ can be *approximated* by some graph-induced metric.

If G is an undirected weighted graph, we denote with d_G the distance function induced by G, i.e. $d_G(x, x')$ denotes the weight of a minimum path between the nodes x and x' in G. Then, if the set of nodes of G is $\mathcal{X}$ and the weight of its edges is given by the metric $d_{\mathcal{X}}$, we can approximate $d_{\mathcal{X}}$ with d_G. In this case, we say that G is a spanning graph, or a spanner [NS07, SU99], of $\mathcal{X}$.

Definition 3.3 (Spanner) *A weighted graph $G = (\mathcal{X}, E)$, with $E \subseteq \mathcal{X} \times \mathcal{X}$ and weight function $w : E \to \mathbb{R}$ is a* spanner *of $\mathcal{X}$ if*

$$w(x, x') = d_{\mathcal{X}}(x, x') \quad \forall (x, x') \in E$$

Note that if G is a spanner of $\mathcal{X}$, then

$$d_G(x, x') \geq d_{\mathcal{X}}(x, x') \quad \forall x, x' \in \mathcal{X}$$

A main concept in the theory of spanners is that of dilation, also known as stretch factor:

Definition 3.4 (Dilation) *Let $G = (\mathcal{X}, E)$ be a spanner of $\mathcal{X}$. The* dilation *of G is calculated as:*

$$\delta = \max_{x \neq x' \in \mathcal{X}} \frac{d_G(x, x')}{d_{\mathcal{X}}(x, x')}$$

A spanner of $\mathcal{X}$ with dilation δ is called a δ-spanner *of $\mathcal{X}$.*

Informally, a δ-spanner of $\mathcal{X}$ can be considered an approximation of the metric $d_{\mathcal{X}}$ in which distances between nodes are "stretched" by a factor of at most δ.

If G is a δ-spanner of $\mathcal{X}$, then it holds that

$$d_G(x, x') \leq \delta d_{\mathcal{X}}(x, x') \quad \forall x, x' \in \mathcal{X}$$

which leads to the following proposition:

Proposition 3.3 *propprivacyimpl Let $\mathcal{X}$ be a set of locations with metric $d_{\mathcal{X}}$, and let G be a δ-spanner of $\mathcal{X}$. If a mechanism K for $\mathcal{X}$ is $\frac{\epsilon}{\delta} d_G$-private, then K is $\epsilon d_{\mathcal{X}}$-private.*

We can then propose a new optimization problem to obtain a $\epsilon d_{\mathcal{X}}$-private mechanism. If $G = (\mathcal{X}, E)$ is a δ-spanner of $\mathcal{X}$, we require not the constraints corresponding to $\epsilon d_{\mathcal{X}}$-privacy, but those corresponding to $\frac{\epsilon}{\delta} d_G$-privacy instead, that is, $|\mathcal{X}|$ constraints for each edge of G:

$$\begin{aligned}
\textbf{Minimize:} \quad & \sum_{x,z \in \mathcal{X}} \pi_x k_{xz} d_Q(x, z) & \\
\textbf{Subject to:} \quad & k_{xz} \leq e^{\frac{\epsilon}{\delta} d_G(x,x')} k_{x'z} & z \in \mathcal{X}, (x, x') \in E \\
& \sum_{x \in \mathcal{X}} k_{xz} = 1 & x \in \mathcal{X} \\
& k_{xz} \geq 0 & x, z \in \mathcal{X}
\end{aligned}$$

Since the resulting mechanism is $\frac{\epsilon}{\delta} d_G$-private, by Proposition 3.3 it must also be $\epsilon d_{\mathcal{X}}$-private. However, the number of constraints induced by $\frac{\epsilon}{\delta} d_G$-privacy is now $|E||\mathcal{X}|$. Moreover, as discussed in the next section, for any $\delta > 1$ there is an algorithm that generates a δ-spanner with $O(\frac{|\mathcal{X}|}{\delta - 1})$ edges, which means that, fixing δ, the total number of constraints of the linear program is $O(|\mathcal{X}|^2)$.

It is worth noting that although $\epsilon d_{\mathcal{X}}$-privacy is guaranteed, optimality is lost: the obtained mechanism is $\frac{\epsilon}{\delta} d_G$-OptQL$(\pi, d_Q)$ but not necessarily $\epsilon d_{\mathcal{X}}$-OptQL(π, d_Q), since the set of $\frac{\epsilon}{\delta} d_G$-private mechanisms is a subset of the set of $\epsilon d_{\mathcal{X}}$-private mechanisms. The QL of the obtained mechanism will now depend on the dilation δ of the spanner: the smaller δ is, the closer the QL of the mechanism will be from the optimal one. In consequence, there is a trade-off between the accuracy of the approximation and the number of constraints in linear program.

3.3.7 Mechanisms for the repeated case

In the previous section we considered a sporadic use of a service, in which case only a single location needs to be obfuscated. We now turn our attention to the repeated case, in which the user's location *trace* (sometimes called *trajectory* in the literature) needs to be protected. We denote by $\mathbf{x} = [x_1, \ldots, x_n]$ a trace, by $\mathbf{x}[i]$ the i-th element of $\mathbf{x}$, by $[\,]$ the empty trace and by $x :: \mathbf{x}$ the trace obtained by adding x to the head of $\mathbf{x}$. We also define $\texttt{tail}(x :: \mathbf{x}) = \mathbf{x}$. As already discussed in Section 3.3.2, geo-indistinguishability can be naturally extended to the case of location traces by using d_∞ as the underlying distinguishability metric.

3.3.8 Independent Mechanism

mechanism IM($\mathbf{x}$)
  $\mathbf{z} := [\,]$
  for $i := 1$ to $|\mathbf{x}|$
    $z := N(\epsilon_N)(\mathbf{x}[i])$
    $\mathbf{z} := z :: \mathbf{z}$
  return $\mathbf{z}$

Figure 8: Independent Mechanism

In order to sanitize $\mathbf{x}$ we can simply apply a *noise mechanism* independently to each secret x_i. We assume that a family of noise mechanisms $N(\epsilon_N) : \mathcal{X} \to \mathcal{P}(\mathcal{Z})$ are available, parametrized by ϵ_N, where each mechanism $N(\epsilon_N)$ satisfies ϵ_N-privacy. Both mechanisms of Section 3.3.4 can be used for this purpose. The resulting mechanism, called the *independent mechanism* IM : $\mathcal{X}^n \to \mathcal{P}(\mathcal{Z}^n)$, is shown in Figure 8. As explained in the introduction, the main issue with IM is that it is $n\epsilon d_\infty$-private, i.e. the budget consumed increases linearly with n.

3.3.9 A predictive $d_\mathcal{X}$-private mechanism

We introduce now our prediction-based approach. The fundamental intuition is that the correlation of the points in the trace can be exploited to the advantage of the mechanism. A simple way of doing this is to try to predict new points from past information; if the point can be predicted with enough accuracy it is called *easy*; in this case the prediction can be reported without adding new noise. One the other hand, *hard* points, that is those that cannot be predicted, are sanitized with new noise. However testing if a point is easy or hard reveals some information about the real location and violates $d_\mathcal{X}$-privacy as for different locations we might have different answers. In order to respect the definition we will need to make the test $d_\mathcal{X}$-private itself, reducing its precision and adding a new cost to our global budget. We will show that with enough correlation in the input the gain in predicted points is worth the cost of the test.

Let $\mathcal{B} = \{0, 1\}$. A boolean $b \in \mathcal{B}$ denotes whether a point is easy (0) or hard (1). A sequence $\mathbf{r} = [z_1, b_1, \ldots, z_n, b_n]$ of reported values and booleans is called a *run*; the set of all runs is denoted by $\mathcal{R} = (\mathcal{Z} \times \mathcal{B})^*$. A run will be the output of our predictive mechanism; note that the booleans b_i are considered public and will be reported by the mechanism.

Main components. The predictive mechanism has three main components: first, the *prediction* is a deterministic function $\Omega : \mathcal{R} \to \mathcal{Z}$, taking as input the run reported up to this moment and trying to predict the next *reported point*, which should be at an acceptable distance from the actual one. The output of the prediction function is denoted by $\tilde{z} = \Omega(\mathbf{r})$. Note that the possibility of a successful prediction should not be viewed as a privacy violation because Ω predicts the reported location, not the actual one.

Second, a *test* is a family of mechanisms $\Theta(\epsilon_\theta, l, \tilde{z}) : \mathcal{X} \to \mathcal{P}(\mathcal{B})$, parametrized by $\epsilon_\theta, l, \tilde{z}$. The test takes as input the point x and reports whether the prediction $\tilde{z}$ is acceptable or not for this point. If the test is successful then the prediction will be used instead of generating new noise. The purpose of the test is to guarantee a certain level of utility: predictions that are farther than the threshold l should be rejected. Since the test is accessing the actual location, it should be private itself, where ϵ_θ is the allowed budget for testing.

The test mechanism that will be used throughout the paper is the one below, which is based on adding Laplace noise to the threshold l:

$$\Theta(\epsilon_\theta, l, \tilde{z})(x) = \begin{cases} 0 & \text{if } d_{\mathcal{X}}(x, \tilde{z}) \leq l + Lap(\epsilon_\theta) \\ 1 & \text{ow.} \end{cases} \tag{3}$$

The test is defined for all $\epsilon_\theta > 0, l \in [0, +\infty), \tilde{z} \in \mathcal{Z}$, and can be used for any metric $d_{\mathcal{X}}$, as long as the domain of reported locations is the same as the one of the actual locations, so that $d_{\mathcal{X}}(x, \tilde{z})$ is well defined.

Finally, a *noise mechanism* is a family of mechanisms $N(\epsilon_N) : \mathcal{X} \to \mathcal{P}(\mathcal{Z})$, parametrized by the available budget ϵ_N. The noise mechanism is used for hard secrets that cannot be predicted and can be any of the sporadic mechanisms presented in Section 3.3.4, although in the following we will assume the use of the planar Laplace for simplicity.

Budget management. The parameters of the mechanism's components need to be configured at each step. This can be done in a dynamic way using the concept of a *budget manager*. A budget manager β is a function that takes as input the run produced so far and returns the budget and the threshold to be used for the test at this step as well as the budget for the noise mechanism: $\beta(\mathbf{r}) = (\epsilon_\theta, \epsilon_N, l)$.

Of course the amount of budget used for the test should always be less than the amount devoted to the noise, otherwise it would be more convenient to just use the independent noise mechanism. Still, there is great flexibility in configuring the various parameters and several strategies can be implemented in terms of a budget manager.

The mechanism. We are now ready to fully describe our mechanism. A single step of the predictive mechanism, displayed in Figure 9b, is a family of mechanisms $\text{Step}(\mathbf{r}) : \mathcal{X} \to \mathcal{P}(\mathcal{Z} \times \mathcal{B})$, parametrized by the run $\mathbf{r}$ reported up to this point. The mechanism takes a location x and returns a reported location

```
mechanism PM(x)
  r := []
  for i := 1 to |x|
    (z, b) := Step(r)(x[i])
    r := (z, b) :: r
  return r
```

(a) Predictive Mechanism

```
mechanism Step(r)(x)
  (ε_θ, ε_N, l) := β(r)
  z̃ := Ω(r)
  b := Θ(ε_θ, l, z̃)(x)
  if b == 0 then z := z̃
  else z := N(ε_N)(x)
  return (z, b)
```

(b) Single step of the Predictive Mechanism

Figure 9: The mechanisms

z, as well as a boolean b denoting whether the secret was easy or hard. First, the mechanism obtains the various configuration parameters from the budget manager as well as a prediction $\tilde{z}$. Then the prediction is tested using the test mechanism. If the test is successful the prediction is returned, otherwise a new reported location is generated using the noise mechanism.

Finally, the predictive mechanism, displayed in Figure 9a, is a mechanism $\text{PM} : \mathcal{X}^n \to \mathcal{P}(\mathcal{R})$. It takes as input a trace $\mathbf{x}$, and applies $\text{Step}(\mathbf{r})$ to each point, while extending at each step the run $\mathbf{r}$ with the new reported values (z, b).

Note that an important advantage of the mechanism is that it is *online*, that is the sanitization of each location does not depend on future ones. This means that the user can query at any time during the life of the system, as opposed to *offline* mechanisms were all the requests need to be generated before the sanitization.

The main innovation of this mechanism if the use of the prediction function, which allows to decouple the privacy mechanism from the correlation analysis, creating a family of modular mechanisms where by *plugging* in different predictions we are able to work in new domains.

3.3.10 Privacy

It can be shown that the predictive mechanism, given a family of test functions and noise functions respectively ϵ_θ and ϵ_N $d_\mathcal{X}$-private, is itself $d_\mathcal{X}$-private. The global budget $\epsilon_\beta(\mathbf{r})$ is actually dependent on the budget manager and on the specific run, which is incompatible with $d_\mathcal{X}$-privacy that is always independent from the prior. The reason is that a hard step is more expensive than an easy step because of the cost of the noise mechanism. Therefore there is a difference between the budget spent on a "good" run, where the input has a considerable correlation, the prediction performs well and the majority of steps are easy, and a run with uncorrelated secrets, where any prediction is useless and all the steps are hard. In the latter case it is clear that our mechanism wastes part of its budget on tests that always fail, performing worse than an independent mechanism.

However we can still enforce the definition with the use of a ϵ-bounded budget manager. Such a budget manager provides a fixed privacy guarantee by sacrificing utility: in the case of a bad run it either needs to lower the budget spend per secret, leading to more noise, or to stop early, handling a smaller number of requests. In this case the budget manager moves the impact of the runs away from the privacy budget and to utility. Two such managers were developed, both with fixed global privacy, one improving QL for a fixed number of requests, the other increasing the number of requests for a certain fixed QL.

References

[ABCP12] Miguel E. Andrés, Nicolás Emilio Bordenabe, Konstantinos Chatzikokolakis, and Catuscia Palamidessi. Geo-indistinguishability: Differential privacy for location-based systems. *CoRR*, abs/1212.1984, 2012.

[ABCP13] Miguel E. Andrés, Nicolás E. Bordenabe, Konstantinos Chatzikokolakis, and Catuscia Palamidessi. Geo-indistinguishability: differential privacy for location-based systems. In *Proceedings of the 20th ACM Conference on Computer and Communications Security (CCS 2013)*, pages 901–914. ACM, 2013.

[ABL15] Alessandro Acquisti, Laura Brandimarte, and George Loewenstein. Privacy and human behavior in the age of information. *Science*, 347(6221):509–514, 2015.

[ACD+07] Claudio Agostino Ardagna, Marco Cremonini, Ernesto Damiani, Sabrina De Capitani di Vimercati, and Pierangela Samarati. Location privacy protection through obfuscation-based techniques. In Steve Barker and Gail-Joon Ahn, editors, *Proc. of the 21st Annual IFIP WG 11.3 Working Conference on Data and Applications Security (DAS)*, volume 4602 of *Lecture Notes in Computer Science*, pages 47–60. Springer, 2007.

[AJL13] Alessandro Acquisti, Leslie K. John, and George Loewenstein. What is privacy worth? *The Journal of Legal Studies*, 42(2):249 – 274, 2013.

[AS03] Daniel Ashbrook and Thad Starner. Using gps to learn significant locations and predict movement across multiple users. *Personal and Ubiquitous Computing*, 7(5):275–286, 2003.

[Bal14] James Ball. Angry birds and 'leaky' phone apps targeted by NSA and GCHQ for user data. The Guardian, January 27, 2014. `http://www.theguardian.com/world/2014/jan/27/nsa-gchq-smartphone-app-angry-birds-personal-data`.

[Bro12] J. Brownlee. This Creepy App Isn't Just Stalking Women Without Their Knowledge, It's A Wake-Up Call About Facebook Privacy (Update), March 2012. `http://www.cultofmac.com/157641/`.

[BS03] Alastair R. Beresford and Frank Stajano. Location privacy in pervasive computing. *IEEE Pervasive Computing*, 2(1):46–55, 2003.

[CABP13] Konstantinos Chatzikokolakis, Miguel E. Andrés, Nicolás E. Bordenabe, and Catuscia Palamidessi. Broadening the scope of Differential Privacy using metrics. In Emiliano De Cristofaro and Matthew Wright, editors, *Proceedings of the 13th International Symposium on Privacy Enhancing Technologies (PETS 2013)*, volume 7981 of *Lecture Notes in Computer Science*, pages 82–102. Springer, 2013.

[CAC12] Rui Chen, Gergely Ács, and Claude Castelluccia. Differentially private sequential data publication via variable-length n-grams. In Ting Yu, George Danezis, and Virgil D. Gligor, editors, *Proceedings of the 19th ACM Conference on Computer and Communications Security (CCS 2012)*, pages 638–649. ACM, 2012.

[CM11] Chi-Yin Chow and Mohamed F. Mokbel. Trajectory privacy in location-based services and data publication. *SIGKDD Explorations*, 13(1):19–29, 2011.

[Dew12] Rinku Dewri. Local differential perturbations: Location privacy under approximate knowledge attackers. *IEEE Transactions on Mobile Computing*, 99(PrePrints):1, 2012.

[DLA05] George Danezis, Stephen Lewis, and Ross J. Anderson. How much is location privacy worth? In *Proceedings of the 4th Annual Workshop on the Economics of Information Security, (WEIS 2005)*, 2005.

[DMDBP08] Yoni De Mulder, George Danezis, Lejla Batina, and Bart Preneel. Identification via location-profiling in gsm networks. In *Proceedings of the 7th ACM Workshop on Privacy in the Electronic Society (WPES 2008)*, pages 23–32. ACM, 2008.

[dMHVB13] Yves-Alexandre de Montjoye, César A. Hidalgo, Michel Verleysen, and Vincent D. Blondel. Unique in the crowd: The privacy bounds of human mobility. *Nature Scientific Reports*, 3(1376), 03 2013.

[DMNS06] Cynthia Dwork, Frank Mcsherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In Shai Halevi and Tal Rabin, editors, *In Proceedings of the Third Theory of Cryptography Conference (TCC)*, volume 3876 of *Lecture Notes in Computer Science*, pages 265–284. Springer, 2006.

[DSR11] Frank Durr, Pavel Skvortsov, and Kurt Rothermel. Position sharing for location privacy in non-trusted systems. In *Proceedings of the Ninth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom 2011)*, pages 189–196, Seattle, WA, USA, March 2011. IEEE Computer Society.

[Dwo06] Cynthia Dwork. Differential privacy. In Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener, editors, *33rd International Colloquium on Automata, Languages and Programming (ICALP 2006)*, volume 4052 of *Lecture Notes in Computer Science*, pages 1–12. Springer, 2006.

[FS14] Kassem Fawaz and Kang G. Shin. Location privacy protection for smartphone users. In *Proceedings of the 21st ACM Conference on Computer and Communications Security (CCS 2014)*, pages 239–250. ACM Press, 2014.

[GG03] Marco Gruteser and Dirk Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. In *Proc. of the First International Conference on Mobile Systems, Applications, and Services (MobiSys)*. USENIX, 2003.

[GH05] Marco Gruteser and Baik Hoh. On the anonymity of periodic location samples. In Dieter Hutter and Markus Ullmann, editors, *Proceedings of the Second International Conference on Security in Pervasive Computing (SPC 2005)*, volume 3450 of *Lecture Notes in Computer Science*, pages 179–192. Springer, 2005.

[GKdPC14] Sébastien Gambs, Marc-Olivier Killijian, and Miguel Núñez del Prado Cortez. De-anonymization attack on geolocated data. *J. Comput. Syst. Sci.*, 80(8):1597–1614, 2014.

[GP09] Philippe Golle and Kurt Partridge. On the anonymity of home/-work location pairs. In Hideyuki Tokuda, Michael Beigl, Adrian Friday, A. J. Bernheim Brush, and Yoshito Tobe, editors, *Proceedings of the 7th International Conference on Pervasive Computing (Pervasive 2009)*, volume 5538 of *Lecture Notes in Computer Science*, pages 390–397. Springer-Verlag, Nara, Japan, May 2009.

[HR11] Shen-Shyang Ho and Shuhua Ruan. Differential privacy for location pattern mining. In *Proceedings of the 4th ACM SIGSPATIAL International Workshop on Security and Privacy in GIS and LBS (SPRINGL)*, pages 17–24. ACM, 2011.

[JHP00] Oliver Jan, Alan Horowitz, and Zhong-Ren Peng. Using global positioning system data to understand variations in path choice. *Transportation Research Record: Journal of the Transportation Research Board*, (1725):37–44, 2000.

[Kru07] John Krumm. Inference attacks on location tracks. In Anthony LaMarca, Marc Langheinrich, and Khai N. Truong, editors, *Proceedings of the 5th International Conference on Pervasive Computing (Pervasive 2007)*, volume 4480 of *Lecture Notes in Computer Science*, pages 127–143. Springer, 2007.

[Kru09] John Krumm. A survey of computational location privacy. *Personal and Ubiquitous Computing*, 13(6):391–399, 2009.

[LLV07] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *ICDE*, volume 7, pages 106–115, 2007.

[LS93] K. Lange and J. S. Sinsheimer. Normal/independent distributions and their applications in robust regression. *Journal of Computational and Graphical Statistics*, 2(2):175–198, 1993.

[MKA+08] Ashwin Machanavajjhala, Daniel Kifer, John M. Abowd, Johannes Gehrke, and Lars Vilhuber. Privacy: Theory meets practice on the map. In Gustavo Alonso, José A. Blakeley, and Arbee L. P. Chen, editors, *Proceedings of the 24th International Conference on Data Engineering, ICDE 2008, April 7-12, 2008, Cancún, México*, pages 277–286. IEEE, 2008.

[MKGV07] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):3, 2007.

[MTC11] Microsoft trustworthy computing. location based services and privacy, January 2011. `http://www.microsoft.com/enus/download/confirmation.aspx?id=3250`.

[MYYR10] Chris Y. T. Ma, David K. Y. Yau, Nung Kwan Yip, and Nageswara S. V. Rao. Privacy vulnerability of published anonymous mobility traces. In *Proceedings of the 16th Annual International Conference on Mobile Computing and Networking, (MOBICOM 2010)*, pages 185–196, 2010.

[NS07] Giri Narasimhan and Michiel Smid. *Geometric spanner networks*. Cambridge University Press, 2007.

[Orl03] Kevin Orland. Stalker Victims Should Check For GPS. The Associated Press, February 2003. `http://www.cbsnews.com/news/stalker-victims-should-check-for-gps/`.

[PL11] Balaji Palanisamy and Ling Liu. Mobimix: Protecting location privacy with mix-zones over road networks. In Serge Abiteboul, Klemens Böhm, Christoph Koch 0001, and Kian-Lee Tan, editors, *Proceedings of the 27th International Conference on Data*

Engineering, ICDE 2011, April 11-16, 2011, Hannover, Germany, pages 494–505. IEEE Computer Society, 2011.

[RWM15] Luca Rossi, James Walker, and Mirco Musolesi. Spatio-temporal techniques for user identification by means of GPS mobility data. *EPJ Data Science*, 4(11), 2015.

[Sam01] Pierangela Samarati. Protecting respondents' identities in microdata release. *IEEE Trans. Knowl. Data Eng*, 13(6):1010–1027, 2001.

[SDB14] Yi Song, Daniel Dahlmeier, and Stéphane Bressan. Not so unique in the crowd: a simple and effective algorithm for anonymizing location data. In Luo Si and Hui Yang, editors, *Proceeding of the 1st International Workshop on Privacy-Preserving IR: When Information Retrieval Meets Privacy and Security*, volume 1225 of *CEUR Workshop Proceedings*, pages 19–24. CEUR-WS.org, 2014.

[Sim07] John Simerman. FasTrak to courthouse. East Bay Times, 2007. `http://www.eastbaytimes.com/2007/06/05/fastrak-to-courthouse/`.

[SS98] Pierangela Samarati and Latanya Sweeney. Generalizing data to provide anonymity when disclosing information (abstract). In ACM, editor, *PODS '98. Proceedings of the ACM SIGACT–SIGMOD–SIGART Symposium on Principles of Database Systems, June 1–3, 1998, Seattle, Washington*, pages 188–188, pub-ACM:adr, 1998. ACM Press.

[SSDF08] Agusti Solanas, Francesc Sebé, and Josep Domingo-Ferrer. Micro-aggregation-based heuristics for p-sensitive k-anonymity: one step beyond. In Farshad Fotouhi, Li Xiong 0001, and Traian Marius Truta, editors, *Proceedings of the 2008 International Workshop on Privacy and Anonymity in Information Society (PAIS 2008)*, ACM International Conference Proceeding Series, pages 61–69. ACM, 2008.

[STD+10] Reza Shokri, Carmela Troncoso, Claudia Díaz, Julien Freudiger, and Jean-Pierre Hubaux. Unraveling an old cloak: k-anonymity for location privacy. In Ehab Al-Shaer and Keith B. Frikken, editors, *Proceedings of the 2010 ACM Workshop on Privacy in the Electronic Society, WPES 2010, Chicago, Illinois, USA, October 4, 2010*, pages 115–118. ACM, 2010.

[SU99] J.R. Sack and J. Urrutia. *Handbook of Computational Geometry*. Elsevier Science, 1999.

[Swe02a] Latanya Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(5):571–588, 2002.

[Swe02b] Latanya Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(5):557–570, 2002.

[TTLM16] Galini Tsoukaneri, George Theodorakopoulos, Hugh Leather, and Mahesh K. Marina. On the inference of user paths from anonymized mobility data. In *Proceedings of the IEEE European Symposium on Security and Privacy (EuroS&P 2016)*, pages 199–213. IEEE, 2016.

[XKP09] Mingqiang Xue, Panos Kalnis, and Hung Keng Pung. Location diversity: Enhanced privacy protection in location based services. In Tanzeem Choudhury, Aaron J. Quigley, Thomas Strang, and Koji Suginuma, editors, *Location and Context Awareness, 4th International Symposium, LoCA 2009, Tokyo, Japan, May 7-8, 2009, Proceedings*, volume 5561 of *Lecture Notes in Computer Science*, pages 70–87. Springer, 2009.

[XZZ+13] Andy Yuan Xue, Rui Zhang, Yu Zheng, Xing Xie, Jin Huang, and Zhenghua Xu. Destination prediction by sub-trajectory synthesis and privacy protection against such prediction. In *29th IEEE International Conference on Data Engineering (ICDE)*, pages 254–265. IEEE, 2013.