Geo-indistinguishability: A Principled Approach to Location Privacy

Konstantinos Chatzikokolakis, Catuscia Palamidessi, Marco Stronati

Geo-indistinguishability: A Principled Approach to Location Privacy

Konstantinos Chatzikokolakis[1,2], Catuscia Palamidessi[2,3], and Marco Stronati[2]

[1] CNRS, France
[2] LIX, École Polytechnique, France
[3] INRIA, France

Abstract. In this paper we report on our ongoing project aimed at protecting the privacy of the user when dealing with location-based services. The starting point of our approach is the principle of geo-indistinguishability, a formal notion of privacy that protects the user's exact location, while allowing approximate information – typically needed to obtain a certain desired service – to be released. We then present two mechanisms for achieving geo-indistinguishability, one generic to sanitize locations in any setting with reasonable utility, the other custom-built for a limited set of locations but providing optimal utility. Finally we extend our mechanisms to the case of location traces, where the user releases his location repeatedly along the day and we provide a method to limit the degradation of the privacy guarantees due to the correlation between the points. All the mechanisms were tested on real datasets and compared both among themselves and with respect to the state of the art in the field.

1 Introduction

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. Just to mention a couple of them, 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 [1].

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 [2] aggregates location check-ins and presents them as "robbery opportunities", pointing out the fact that publically announcing one's location effectively reveals to the world that they are not home.

A survey among 180 smartphone users, described in [3], reported that 78% of the participants believe that apps accessing their location can pose privacy threats. Furthermore, 85% of them declared that they care about who accesses their location information. All these worries about location privacy may seem exaggerated at first, but one can see that they are fully justified when thinking to the possible malicious uses of location information, such as robbing and stalking. For instance, the application “Girls Around Me”, combines social media and location information to find nearby women (who hadn’t necessarily agreed to be found), and, with one click the user can access the Facebook profiles of targeted girls [4]. Particularly worrisome is the perspective of potential combination with the users’ most sensitive information, such as sexual orientation. Again, according to the Guardian [1], there have been cases of smartphone applications from which such information was collected without the user’s knowledge.

Furthermore, location information can be easily used to obtain a variety of other information that an individual usually wishes to protect: by collecting and processing accurate location data on a regular basis, it is possible to infer an individual’s home or work location, sexual preferences, political views, religious inclinations, etc.

There are numerous programs that collect location data from mobile devices. In this paper, we focus our attention to those applications which collect such data to provide an agreed-upon service, i.e., the LBSs. Obviously there exist methods for preventing the collection of location data entirely, however they would completely nullify the benefits of applications which provide location services. Our primary goal is to develop methods that hinder the undesired tracking capacities of LBSs, while preserving as much as possible the quality of the desired services.

Several notions of privacy for location-based systems have been proposed in the literature. In Section 2 we give an overview of such notions, and we discuss their shortcomings in relation to our motivating LBS applications. Aiming at addressing these shortcomings, we propose a formal privacy definition, called *geo-indistinguishability*, that allows a user to disclose *enough location information* to obtain the desired service, while satisfying the aforementioned privacy notion. Our proposal is based on a generalization of *differential privacy* [5] developed in [6]. Similarly to differential privacy, our notion and technique abstract from the side information of the adversary, such as any prior probabilistic knowledge about the user’s actual location.

To explain the principle of geo-indistinguishability, consider a user located in Paris who wishes to query an LBS provider for nearby restaurants in a private way. To achieve this the user employs *obfuscation*, i.e. he discloses some approximate location z instead of his exact one x. Interestingly, 52% of the surveyed individuals in [3] stated no problem in supplying apps with imprecise location information to protect their privacy; only 18% objected to providing imprecise location information. Note that, in contrast to various works in the literature, we assume that the user is interested in hiding his *location*, not his *identity*; in fact, the user might be authenticated to the service provider in order to obtain personalized recommendations.

We say that the user enjoys *ℓ-privacy within r* if, any two locations at distance at most r produce observations with “similar” distributions, where the “level of similarity” depends on ℓ. The idea is that ℓ represents the user’s *level* of privacy for that radius: the smaller ℓ is, the higher is the privacy.

The definition of geo-indistinguishability abstracts from r by requiring that the (inverse of the) level of privacy ℓ depend on the radius r. Formally: A mechanism satisfies geo-indistinguishability iff for any radius $r > 0$, the user enjoys ϵr-privacy within r.

This definition implies that the user is protected within any radius r, but with a level $\ell = \epsilon r$ that increases with the distance. Within a short radius, for instance $r = 1$ km, ℓ is small, guaranteeing that the provider cannot infer the user's location within, say, the 7th arrondissement of Paris. Farther away from the user, for instance for $r = 1000$ km, ℓ becomes large, allowing the LBS provider to infer that with high probability the user is located in Paris instead of, say, London.

We propose a mechanism that achieves geo-indistinguishability by perturbating the user's location x. The inspiration for our mechanism comes from one of the most popular approaches for differential privacy, namely the Laplace noise. We adopt a specific planar version of the Laplace distribution, allowing to draw points in a *geo-indistinguishable* way; moreover, we are able to do so efficiently, by using polar coordinates. Another advantage of the resulting mechanism is that it is independent from the particular user or the area it is used in, the only parameter is the desired level of privacy or conversely the desired level of accuracy of the service.

Clearly, the perturbation of the information sent to the LBS provider leads to a degradation of the quality of service, and consequently there is a trade-off between the level of privacy that the user wishes to guarantee and the service quality loss (QL) that he has to accept. The study of this trade-off, and the design of mechanisms which optimize it, is an important research direction started with the seminal paper of Shokri et al. [7]. In [8] we have compared our mechanism with other ones in the literature, using the privacy metric proposed in [9]. It turns our that our mechanism offers the best privacy guarantees, for the same utility, among those which do not depend on the user.

The advantages of the independence from the user are obvious: first, the mechanism is designed once and for all, we do not need different mechanisms for different users. Second, even the same user may have different behaviors, for instance during different parts of the day, and it would not be practical to change the mechanism all the time. Finally, computing the prior of the user can be an expensive operation, and in some cases even unfeasible.

However, if we are interested in protecting a particular user, then in general there are mechanisms, specific for that user, that do better than the generic Laplace mechanism. Thus, we are also interested in defining specialized mechanisms that optimize the trade-off between geo-indistinguishability and quality of service for a particular user. More precisely, given a certain threshold on the degree of geo-indistinguishability, and a prior, we aim at obtaining the mechanism K which minimizes the QL. Based on the fact that the geo-indistinguishability threshold can be expressed by linear constraints, we can reduce the problem of producing such an optimal K to a linear optimization problem, which can then be solved by using standard techniques of linear programming.

The two mechanisms discussed above correspond to a *sporadic* use of the service in which a single location needs to be sanitized. In practice, however, a user might performs *repeated* location-based queries from several locations, forming a *location trace* that he wishes to protect. For each query, a new obfuscated location needs to be reported to the service provider, which can be easily obtained by independently adding noise at the moment when each query is executed. We refer to independently applying noise to each location as the *independent mechanism*.

However, it is easy to see that privacy is degraded as the number of queries increases, due to the *correlation* between the locations. Intuitively, in the extreme case when the user never moves (i.e. there is perfect correlation), the reported locations are centered around the real one, thus revealing it more and more precisely as the number of queries increases. Technically, the independent mechanism applying ϵ-geo-indistinguishable noise (where ϵ is a privacy parameter) to n locations can be shown to satisfy $n\epsilon$-geo-indistinguishability. This is typical in the area of differential privacy, in which ϵ is thought as a privacy *budget*, consumed by each query; this linear increase makes the mechanism applicable only when the number of queries remains small. In order to deal with multiple queries we propose a *trace obfuscation* mechanism with a smaller *budget consumption rate* than applying independent noise [10]. The main idea is to actually use the correlation from previous locations to try to *predict* a point close to the user's actual location. Predicted points are safe to report directly and thus have a smaller footprint on the privacy budget.

We experimentally compare the above mechanisms on two large real-life data sets, Geolife and Tdrive. The results show the utility improvements of the optimal constructed mechanism wrt the Laplace one, as well as the improvements of the predictive mechanism wrt the independently applied noise.

This paper presents a systematic overview of the approach to location privacy developed by our INRIA team Comète. Some of the results presented here have appeared in previous papers of ours specialized in particular aspects of the project [8,10,11].

Road Map. In Section 2 we discuss notions of location privacy from the literature and point out their weaknesses and strengths. In Section 3 we formalize the notion of geo-indistinguishability in three equivalent ways. We then proceed to describe two mechanisms that provide geo-indistinguishability in Section 4: one general, the other with optimal utility. In Section 5 we propose a predictive mechanism that exploits correlations on the input by means of a prediction function to improve the privacy guarantee. In Section 6 we give an overview of the experimental analysis and comparison of the mechanisms and Section 7 concludes.

2 Existing Notions of Privacy

In this section, we examine various notions of location privacy from the literature, as well as techniques to achieve them. We consider the motivating example from the introduction, of a user in Paris wishing to find nearby restaurants with good reviews. To achieve this goal, he uses a handheld device (e.g.. a smartphone) to query a public LBS provider. However, the user expects his location to be kept private: informally speaking, the information sent to the provider should not allow him to accurately infer the user's location. Our goal is to provide a *formal* notion of privacy that adequately captures the user's expected privacy. From the point of view of the employed mechanism, we require a technique that can be performed in real-time by a handheld device, without the need of any trusted anonymization party.

Expected Adversary Error. The expected error of an optimal Bayesian adversary [7,9,12] is a natural way to quantify the privacy offered by a location-obfuscation mechanism.

Intuitively, it reflects the degree of accuracy by which an adversary can guess the real location of the user by observing the obfuscated location, and using any side-information available to him.

There are several works relying on this notion. In [12], a perturbation mechanism is used to confuse the attacker by crossing paths of individual users, rendering the task of tracking individual paths challenging. In [9], an optimal location-obfuscation mechanism (i.e., achieving maximum level of privacy for the user) is obtained by solving a linear program in which the constraints are determined by the quality of service and by the user's profile. In [13] bandwidth constraints are also taken into account, while [14] considers the case of repeated location reporting, as opposed to a sporadic use of the mechanism. Furthermore, [15] analyzes the case where the attacker can also exploit co-location information, such as geo-located pictures, shared on a social network, in which several friends are tagged together.

It is worth noting that this privacy notion and the obfuscation mechanisms based on it are explicitly defined in terms of the adversary's side information. In contrast, our notion of geo-indistinguishability abstracts from the attacker's prior knowledge, and is therefore suitable for scenarios where the prior is unknown, or the same mechanism must be used for multiple users.

k-anonymity. The notion of k-anonymity is the most widely used definition of privacy for location-based systems in the literature. Many systems in this category [16,17,18] aim at protecting the user's *identity*, requiring that the attacker cannot infer which user is executing the query, among a set of k different users. Such systems are outside the scope of our problem, since we are interested in protecting the user's *location*.

On the other hand, k-anonymity has also been used to protect the user's location (sometimes called l-diversity in this context), requiring that it is indistinguishable among a set of k points (often required to share some semantic property). One way to achieve this is through the use of *dummy locations* [19,20]. This technique involves generating $k - 1$ properly selected dummy points, and performing k queries to the service provider, using the real and dummy locations. Another method for achieving k-anonymity is through *cloaking* [21,22,23]. This involves creating a cloaking region that includes k points sharing some property of interest, and then querying the service provider for this cloaking region.

Even when side knowledge does not explicitly appear in the definition of k-anonymity, a system cannot be proven to satisfy this notion unless assumptions are made about the attacker's side information. For example, dummy locations are only useful if they look equally likely to be the real location from the point of view of the attacker. Any side information that allows to rule out any of those points, as having low probability of being the real location, would immediately violate the definition.

Counter-measures are often employed to avoid this issue: for instance, [19] takes into account concepts such as ubiquity, congestion and uniformity for generating dummy points, in an effort to make them look realistic. Similarly, [23] takes into account the user's side information to construct a cloaking region. Such counter-measures have their own drawbacks: first, they complicate the employed techniques, also requiring additional data to be taken into account (for instance, precise information about the environment or the location of nearby users), making their application in real-time by a handheld device challenging. Moreover, the attacker's actual side information might simply be inconsistent with the assumptions being made.

As a result, notions that abstract from the attacker's side information, such as differential privacy, have been growing in popularity in recent years, compared to k-anonymity-based approaches.

Differential Privacy. Differential Privacy [5] 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. A typical way to achieve this notion is to add controlled random noise to the query output, for example drawn from a Laplace distribution. An advantage of this notion is that a mechanism can be shown to be differentially private independently from any side information that the attacker might possess.

Differential privacy has also been used in the context of location privacy. In [24], it is shown that a synthetic data generation technique can be used to publish statistical information about commuting patterns in a differentially private way. In [25], a quadtree spatial decomposition technique is used to ensure differential privacy in a database with location pattern mining capabilities, while [26] uses variable-length n-grams to disclose sequential data, such as mobility traces, in a differentially private way.

As shown in the aforementioned works, differential privacy can be successfully applied in cases where *aggregate* information about several users is published. On the other hand, the nature of this notion makes it poorly suitable for applications in which only a single individual is involved, such as our motivating scenario. The secret in this case is the location of a single user. Thus, differential privacy 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 [27] 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 noise[4] for location privacy is valid, and [27] provides extensive experimental analysis supporting this claim. Our notion

[4] The planar Laplace distribution that we use in our work, however, is different from the distribution obtained by adding Laplace noise to each Cartesian coordinate, and has better differential privacy properties (c.f. Section 4.1).

of geo-indistinguishability provides the formal background for justifying the use of Laplace noise, while avoiding the need to fix an anonymity set by using the generalized variant of differential privacy from [6].

Other location-privacy metrics. [28] proposes a location cloaking mechanism, and focuses on the evaluation of Location-based Range Queries. The degree of privacy is measured by the size of the cloak (also called *uncertainty region*), and by the coverage of sensitive regions, which is the ratio between the area of the cloak and the area of the regions inside the cloak that the user considers to be sensitive. In order to deal with the side-information that the attacker may have, ad-hoc solutions are proposed, like patching cloaks to enlarge the uncertainty region or delaying requests. Both solutions may cause a degradation in the quality of service.

In [29], the real location of the user is assumed to have some level of inaccuracy, due to the specific sensing technology or to the environmental conditions. Different obfuscation techniques are then used to increase this inaccuracy in order to achieve a certain level of privacy. This level of privacy is defined as the ratio between the accuracy before and after the application of the obfuscation techniques.

Similar to the case of k-anonymity, both privacy metrics mentioned above make implicit assumptions about the adversary's side information. This may imply a violation of the privacy definition in a scenario where the adversary has some knowledge about the user's real location.

Transformation-based approaches. A number of approaches for location privacy are radically different from the ones mentioned so far. Instead of cloaking the user's location, they aim at making it completely invisible to the service provider. This is achieved by transforming all data to a different space, usually employing cryptographic techniques, so that they can be mapped back to spatial information only by the user [30,31]. The data stored in the provider, as well as the location send by the user are encrypted. Then, using techniques from *private information retrieval*, the provider can return information about the encrypted location, without ever discovering which actual location it corresponds to.

A drawback of these techniques is that they are computationally demanding, making it difficult to implement them in a handheld device. Moreover, they require the provider's data to be encrypted, making it impossible to use existing providers, such as Google Maps, which have access to the real data.

Effectiveness of attacks. An indirect way of assessing the privacy guarantees of a mechanism is to measure the effectiveness of various location inference attacks. Several works present attacks and practical challenges for location privacy. In [32] the authors develop and test a toolkit for inference attacks on the reported locations of users to discover points of interests, future locations and co-location of two individuals. The same technique was employed in [33] focusing on de-anonymization attacks with the goal of evaluating the effectiveness of sanitization mechanisms. In [34] the authors tested the resilience of Geo-Indistinguishability to identification of Points of Interests of users over two real GPS traces datasets, with varying level of privacy (and therefore noise).

3 Geo-Indistinguishability

In this section we formalize our notion of geo-indistinguishability. As already discussed in the introduction, 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.

So far we kept the discussion on an informal level by avoiding to explicitly define what ℓ-privacy within r means. In the remaining of this section we give a formal definition, 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 technique used in Section 4). 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.[5] 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.

[5] For simplicity we assume distributions on $\mathcal{X}$ to be discrete, but allow those on $\mathcal{Z}$ to be continuous (c.f. Section 4). All sets to which probability is assigned are implicitly assumed to be measurable.

3.1 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 1 (geo-indistinguishability). *A mechanism K satisfies ϵ-geo-indistinguishability iff for all x, x':*

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

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[6], and is studied on its own right in [6]. 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.

[6] 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

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.

3.2 Characterizations

In this section we state two characterizations of geo-indistinguishability, obtained from the corresponding results of [6] (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 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 $Z \subseteq \mathcal{Z}$:*

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

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 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 [35].

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 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.

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

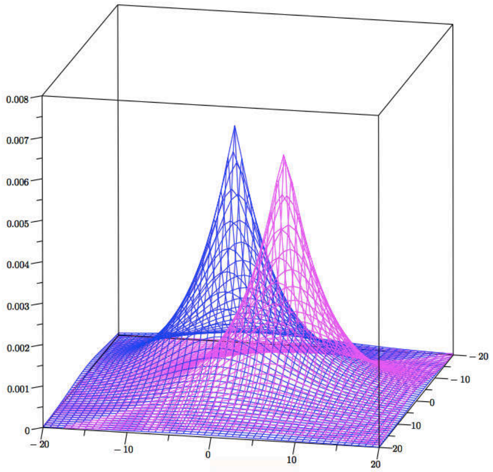

Fig. 1. The pdf of two planar Laplace distributions, centered at $(-2,-4)$ and at $(5,3)$ respectively, with $\epsilon = 1/5$.

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.

4.1 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 1. The resulting mechanism can be shown to satisfy ϵ-geo-indistinguishability [8].

Note that this definition of the two-dimensional Laplace distribution follows [36] 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:

$$\begin{aligned} D_{\epsilon,R}(r) &= \textstyle\int_0^{2\pi} D_\epsilon(r, \theta)\, d\theta \;=\; \epsilon^2\, r\, e^{-\epsilon\, r} \\ D_{\epsilon,\Theta}(\theta) &= \textstyle\int_0^{\infty} D_\epsilon(r, \theta)\, dr \;=\; \frac{1}{2\,\pi} \end{aligned}$$

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 [8].

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* [37], 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.

4.2 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

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

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 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 $\epsilon d_{\mathcal{X}}$-*$\text{O}\textsc{pt}\text{QL}(\pi, d_Q)$ *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:

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

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

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 [38,39], of $\mathcal{X}$.

Definition 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 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 1. *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:

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

Since the resulting mechanism is $\frac{\epsilon}{\delta}d_G$-private, by Proposition 1 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 $\frac{\epsilon}{\delta}d_G$-$\textsc{Opt}QL(\pi, d_Q)$ but not necessarily $\epsilon d_\mathcal{X}$-$\textsc{Opt}$QL(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.

5 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 3.1, geo-indistinguishability can be naturally extended to the case of location traces by using d_∞ as the underlying distinguishability metric.

5.1 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 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 2. 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.

```
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 $\mathbf{z}$
```

Fig. 2: Independent Mechanism

5.2 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.

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

```
mechanism Step(r)(x)
  $(\epsilon_\theta, \epsilon_N, l) := \beta(\mathbf{r})$
  $\tilde{z} := \Omega(\mathbf{r})$
  $b := \Theta(\epsilon_\theta, l, \tilde{z})(x)$
  if $b == 0$ then $z := \tilde{z}$
  else $z := N(\epsilon_N)(x)$
  return $(z, b)$
```

Fig. 3: Predictive Mechanism

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 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 3, 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 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 **??**, 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.

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.

6 Evaluation

We experimentally verify the effectiveness of our mechanisms on the motivating example of a user performing various activities in a city, using two large data sets of GPS trajectories in the Beijing urban area ([40,41]). Geolife [40] collects the movements of several users, using a variety of transportation means, including walking, while in Tdrive [41] we find exclusively taxi drivers trajectories. Due to space restrictions, only a small part of the results are given here; a detailed evaluation is available in [8,10,11].

Optimal mechanism. To show the benefits of using a mechanism with optimal utility, we compare now the QL of the optimal mechanism (OPTQL) and of the planar Laplace (PL) when both are generated with the same privacy level ϵ. We can see the results in Figure 4. The OPTQL mechanism clearly offers a better utility to the user, while guaranteeing the same level of geo-indistinguishability.

Regarding the spanner approximation of the optimal mechanism, the relation between the dilation and the number of constraints is shown in Figure 4. It is clear that the

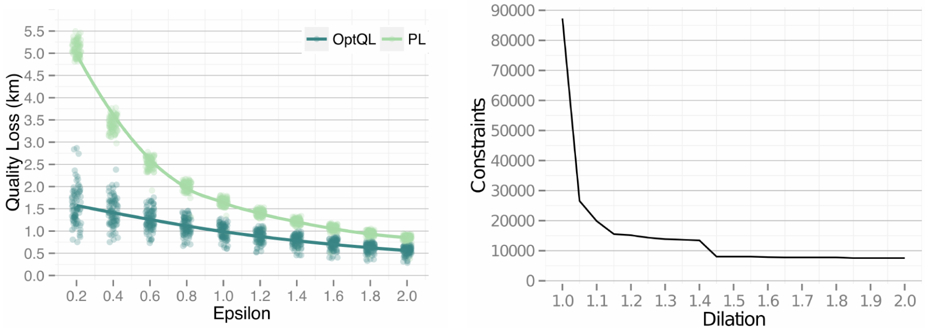

Fig. 4: Left: Quality loss of the OPTQL and PL mechanisms for different values of ϵ. The mechanisms were calculated for all users. Points represent the utility for every user, while the two lines join the medians for each mechanism and each value of ϵ. Right: Relation between the approximation ratio and the number of constraints in the linear program. This number is independent from the user and from the value of ϵ.

number of constraints decreases exponentially with respect to the dilation, and therefore even for small dilations (which in turn mean good approximations) the number of constraints is significantly reduced with the proposed approximation technique. For instance, we have 87250 constraints for $\delta = 1$ (the optimal case), and 25551 constraints for $\delta = 1.05$. This represents a decrease of 71% with respect to the optimal case, with only 1.05 approximation ratio.

Predictive mechanism. In order to model both frequent (easier to predict) as well as seldom users, the GPS traces were sampled with a different probability of *jumping*, i.e. performing a query with a long delay (one hour) after the previous one. The test included two budget managers, one optimizing QL for a fixed number of queries (fixed-rate), the other reducing budget consumption to prolong the use of the system at a fixed QL (fixed-ql). The results, shown in Figure 5, show considerable improvements with respect to independently applied noise, for both managers: we are able to decrease the average error up to 40% and the budget consumption rate up to 64%. The improvements are significant enough to broaden the applicability of geo-indistinguishability to cases impossible before: in our experiments we cover 30 queries with reasonable error which is enough for a full day of usage; alternatively we can drive the error down from 5 km to 3 km, which make it acceptable for a variety of applications.

7 Related work

Several related works have been already presented in Section 2, a few more are discussed in this section.

On the side of the optimal mechanism construction, the work closest to ours is [42], which independently proposes a linear programming technique to construct an optimal

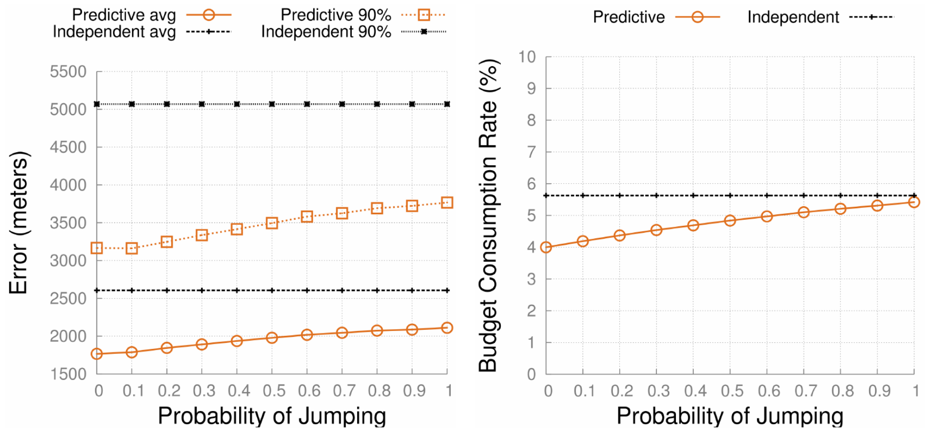

Fig. 5: (Left) Average and 90th percentile error for fixed-rate. (Right) Budget consumption rate for fixed-ql.

obfuscation mechanism wrt either the expected adversary error or geo-indistinguishability. Although there is an overlap in the main construction (the optimization problem of Section 4.2), most of the results are substantially different. The approximation technique of [42] consists of discarding some of the geo-indistinguishability constraints when the distance involved is larger than a certain lower bound. This affects the geo-indistinguishability guarantees of the mechanism, although the effect can be tuned by properly selecting the bound for discarding constraints. On the other hand, our approximation technique, based on spanning graphs, can be used to reduce the number of constraints from cubic to quadratic without jeopardizing the privacy guarantees, by accepting a small decrease on the utility.

On the side of the predictive mechanism, our work was mainly inspired by the median mechanism [43], a work on differential privacy for databases based on the idea of exploiting the correlation on the queries to improve the budget usage. The mechanism uses a concept similar to our *prediction* to determine the answer to the next query using only past answers. An analogous work is the multiplicative weights mechanism [44], again in the context of statistical databases. The mechanism keeps a parallel version of the database which is used to predict the next answer and in case of failure it is updated with a multiplicative weights technique.

A key difference from our context is that in the above works, several queries are performed against the *same database*. In our setting, however, the secret (the position of the user) is always changing, which requires to exploit correlations in the data. This scenario is explored also in [45] were the authors consider the case of an evolving secret and develop a differentially private counter.

Another work very close in spirit to ours is [46]. The authors of this paper also consider the problem of location privacy for location based services, and use random noise to conceal the actual location. However their work is mainly focused on exploiting the features of existing technology, and does not attempt to give a rigorous definition of privacy guarantees.

In a recent paper [3], Fawaz and Shin propose the *Location Privacy Guardian*, which is perhaps the most complete framework, in the current state of the art, for privacy protection within smartphone applications. They consider several potential sources of privacy breaches (profiling, tracking, etc.) and propose solutions for each of them. For location privacy, they use our Laplace mechanism.

8 Conclusion

In this paper we have presented a framework for achieving privacy in location-based applications, taking into account the desired level of protection as well as the side-information that the attacker might have about the user. The core of our proposal is a new notion of location privacy, that we call geo-indistinguishability. In order to ensure this kind of privacy protection in location-based services, we have proposed mechanisms that achieve geo-indistinguishability by perturbating the actual location with random noise. We have considered two kinds of mechanisms: the first one is universal, i.e., it does not depend on the user, and uses a bivariate version of the Laplace function as the density function of the noise. The second one is designed assuming a particular user, and for that user it achieves the optimal trade off between privacy and utility. This is done by formulating the optimal trade off as a linear programming problem, whose solution are the conditional probabilities that compose the noise matrix. Finally, we have considered the problem of traces, namely the repeated use of the mechanism to generate a sequence of points (a situation that may arise, for instance, when the user makes several requests to the service during a walk), and we have addressed the problem of the degradation of the level of privacy due to the correlation of the actual locations. We have proposed a method that limits the degradation by applying a prediction mechanism, which allows to generate new reported locations without applying the mechanism at each step. Finally, we have evaluated our methods and showed that they are a considerable improvement w.r.t. the state of the art, and that our proposal to limit the negative effects of the correlation in traces is effective in practice.

Acknowledgments. This work was partially supported by the MSR-INRIA joint lab, by the European Union 7th FP project MEALS, by the project ANR-12-IS02-001 PACE, and by the INRIA Large Scale Initiative CAPPRIS.

References

1. Ball, J.: Angry birds and 'leaky' phone apps targeted by nsa and gchq for user data. The Guardian (2014) `http://www.theguardian.com/world/2014/jan/27/nsa-gchq-smartphone-app-angry-birds-personal-data`.
2. Please Rob Me: `http://pleaserobme.com/`.
3. Fawaz, K., Shin, K.G.: Location privacy protection for smartphone users. In: Proc. of CCS, ACM Press (2014) 239–250

4. Brownlee, J.: This creepy app isn't just stalking women without their knowledge, it's a wake-up call about facebook privacy [update]. Cult of Mac (2012) http://www.cultofmac.com/157641/this-creepy-app-isnt-just-stalking-women-without-their-knowledge-its-a-wake-up-call-about-facebook-privacy/.
5. Dwork, C.: Differential privacy. In: Proc. of ICALP. Volume 4052 of LNCS., Springer (2006) 1–12
6. Chatzikokolakis, K., Andrés, M.E., Bordenabe, N.E., Palamidessi, C.: Broadening the scope of Differential Privacy using metrics. In: Proc. of PETS. Volume 7981 of LNCS., Springer (2013) 82–102
7. Shokri, R., Theodorakopoulos, G., Boudec, J.Y.L., Hubaux, J.P.: Quantifying location privacy. In: Proc. of S&P, IEEE (2011) 247–262
8. Andrés, M.E., Bordenabe, N.E., Chatzikokolakis, K., Palamidessi, C.: Geo-indistinguishability: differential privacy for location-based systems. In: Proc. of CCS, ACM (2013) 901–914
9. Shokri, R., Theodorakopoulos, G., Troncoso, C., Hubaux, J.P., Boudec, J.Y.L.: Protecting location privacy: optimal strategy against localization attacks. In: Proc. of CCS, ACM (2012) 617–627
10. Chatzikokolakis, K., Palamidessi, C., Stronati, M.: A predictive differentially-private mechanism for mobility traces. In: Proc. of PETS. Volume 8555 of LNCS., Springer (2014) 21–41
11. Bordenabe, N.E., Chatzikokolakis, K., Palamidessi, C.: Optimal geo-indistinguishable mechanisms for location privacy. In: Proc. of CCS. (2014)
12. Hoh, B., Gruteser, M.: Protecting location privacy through path confusion. In: Proc. of SecureComm, IEEE (2005) 194–205
13. Herrmann, M., Troncoso, C., Diaz, C., Preneel, B.: Optimal sporadic location privacy preserving systems in presence of bandwidth constraints. In: Proc. of WPES. (2013)
14. Theodorakopoulos, G., Shokri, R., Troncoso, C., Hubaux, J., Boudec, J.L.: Prolonging the hide-and-seek game: Optimal trajectory privacy for location-based services. CoRR **abs/1409.1716** (2014)
15. Olteanu, A., Huguenin, K., Shokri, R., Hubaux, J.: Quantifying the effect of co-location information on location privacy. In: Proc. of PETS. LNCS, Springer (2014) 184–203
16. Gruteser, M., Grunwald, D.: Anonymous usage of location-based services through spatial and temporal cloaking. In: Proc. of MobiSys, USENIX (2003)
17. Gedik, B., Liu, L.: Location privacy in mobile systems: A personalized anonymization model. In: Proc. of ICDCS, IEEE (2005) 620–629
18. Mokbel, M.F., Chow, C.Y., Aref, W.G.: The new casper: Query processing for location services without compromising privacy. In: Proc. of VLDB, ACM (2006) 763–774
19. Kido, H., Yanagisawa, Y., Satoh, T.: Protection of location privacy using dummies for location-based services. In: Proc. of ICDE Workshops. (2005) 1248
20. Shankar, P., Ganapathy, V., Iftode, L.: Privately querying location-based services with SybilQuery. In: Proc. of UbiComp, ACM (2009) 31–40
21. Bamba, B., Liu, L., Pesti, P., Wang, T.: Supporting anonymous location queries in mobile environments with privacygrid. In: Proc. of WWW, ACM (2008) 237–246
22. Duckham, M., Kulik, L.: A formal model of obfuscation and negotiation for location privacy. In: Proc. of PERVASIVE. Volume 3468 of LNCS., Springer (2005) 152–170
23. Xue, M., Kalnis, P., Pung, H.: Location diversity: Enhanced privacy protection in location based services. In: Proc. of LoCA. Volume 5561 of LNCS., Springer (2009) 70–87
24. Machanavajjhala, A., Kifer, D., Abowd, J.M., Gehrke, J., Vilhuber, L.: Privacy: Theory meets practice on the map. In: Proc. of ICDE, IEEE (2008) 277–286
25. Ho, S.S., Ruan, S.: Differential privacy for location pattern mining. In: Proc. of SPRINGL, ACM (2011) 17–24

26. Chen, R., Ács, G., Castelluccia, C.: Differentially private sequential data publication via variable-length n-grams. In: Proc. of CCS, ACM (2012) 638–649
27. Dewri, R.: Local differential perturbations: Location privacy under approximate knowledge attackers. IEEE Trans. on Mobile Computing **99**(PrePrints) (2012) 1
28. Cheng, R., Zhang, Y., Bertino, E., Prabhakar, S.: Preserving user location privacy in mobile data management infrastructures. In: Proc. of PET. Volume 4258 of LNCS., Springer (2006) 393–412
29. Ardagna, C.A., Cremonini, M., Damiani, E., di Vimercati, S.D.C., Samarati, P.: Location privacy protection through obfuscation-based techniques. In: Proc. of DAS. Volume 4602 of LNCS., Springer (2007) 47–60
30. Khoshgozaran, A., Shahabi, C.: Blind evaluation of nearest neighbor queries using space transformation to preserve location privacy. In: Proc. of SSTD. Volume 4605 of LNCS., Springer (2007) 239–257
31. Ghinita, G., Kalnis, P., Khoshgozaran, A., Shahabi, C., Tan, K.L.: Private queries in location based services: anonymizers are not necessary. In: Proc. of SIGMOD, ACM (2008) 121–132
32. Gambs, S., Killijian, M.O., del Prado Cortez, M.N.: Show me how you move and i will tell you who you are. Trans. on Data Privacy **4**(2) (2011) 103–126
33. Gambs, S., Killijian, M., del Prado Cortez, M.N.: De-anonymization attack on geolocated data. In: Proc. of TrustCom 2013, IEEE (2013) 789–797
34. Primault, V., Mokhtar, S.B., Lauradoux, C., Brunie, L.: Differentially private location privacy in practice. In: Proc. of MoST 2014, IEEE (2014)
35. Dwork, C., Mcsherry, F., Nissim, K., Smith, A.: Calibrating noise to sensitivity in private data analysis. In: Proc. of TCC. Volume 3876 of LNCS., Springer (2006) 265–284
36. Lange, K., Sinsheimer, J.S.: Normal/independent distributions and their applications in robust regression. J. of Comp. and Graphical Statistics **2**(2) (1993) 175–198
37. Location Guard: `https://github.com/chatziko/location-guard`.
38. Narasimhan, G., Smid, M.: Geometric spanner networks. CUP (2007)
39. Sack, J., Urrutia, J.: Handbook of Computational Geometry. Elsevier (1999)
40. Zheng, Y., Xie, X., Ma, W.Y.: Geolife: A collaborative social networking service among user, location and trajectory. IEEE Data Eng. Bull. **33**(2) (2010) 32–39
41. Yuan, J., Zheng, Y., Zhang, C., Xie, W., Xie, X., Sun, G., Huang, Y.: T-drive: driving directions based on taxi trajectories. In: GIS. (2010) 99–108
42. Shokri, R.: Optimal user-centric data obfuscation. Technical report, ETH Zurich (2014) `http://arxiv.org/abs/1402.3426`.
43. Roth, A., Roughgarden, T.: Interactive privacy via the median mechanism. In: Proc. of STOC. (2010) 765–774
44. Hardt, M., Rothblum, G.N.: A multiplicative weights mechanism for privacy-preserving data analysis. In: FOCS, IEEE (2010) 61–70
45. Dwork, C., Naor, M., Pitassi, T., Rothblum, G.N.: Differential privacy under continual observation. In: STOC, ACM (2010) 715–724
46. Merrill, S., Basalp, N., Biskup, J., Buchmann, E., Clifton, C., Kuijpers, B., Othman, W., Savas, E.: Privacy through uncertainty in location-based services. In: 2013 IEEE 14th Int. Conf. on Mobile Data Management, IEEE Computer Society (2013) 67–72