A Uniformity-Based Approach to Location Privacy

Pericle Perazzo[a], Gianluca Dini[a]

[a]*Largo Lucio Lazzarino 1, 56122 – Pisa, Italy*

Abstract

As location-based services emerge, many people feel exposed to high privacy threats. Privacy protection is a major challenge for such services and related applications. A simple approach is *perturbation*, which adds an artificial noise to positions and returns an obfuscated measurement to the requester. Our main finding is that, unless the noise is chosen properly, these methods do not withstand attacks based on statistical analysis. In this paper, we propose UniLO, an obfuscation operator which offers high assurances on obfuscation uniformity, even in case of imprecise location measurement. We also deal with service differentiation by proposing three UniLO-based obfuscation algorithms that offer multiple contemporaneous levels of privacy. Finally, we experimentally prove the superiority of the proposed algorithms compared to the state-of-the-art solutions, both in terms of utility and resistance against inference attacks.

Keywords: privacy, location-based services, obfuscation techniques, uniform obfuscation

1. Introduction

Recent years have seen the widespread diffusion of cheap localization technologies. The most known is GPS, but there are many other examples, like cellular positioning, ultra-wide band positioning, etc. [1, 2, 3] The emergence of such technologies has brought to the development of *location-based services* (*LBS*) [4, 5, 6], which rely on the knowledge of location of people or things. The retrieval of people's location raises several privacy concerns, as it is personal, often sensitive, information. The indiscriminate disclosure of such data could have highly negative effects, from undesired location-based advertising to personal safety attempts.

A classic approach to the problem is to introduce strict access-control policies in the system [7, 8]. This approach has a main drawback: if the entity does not need complete (or exact) information, granting the access to it is a useless exposure of personal data. The "permit-or-deny" outcome of access control is often too rigid.

Email addresses: `pericle.perazzo@iet.unipi.it` (Pericle Perazzo), `gianluca.dini@iet.unipi.it` (Gianluca Dini)

Samarati and Sweeney [9, 10] proposed the concept of *k-anonymity*: a system offers a k-anonymity to a user if his identity is undistinguishable from at least $k-1$ other users. k-anonymity concepts have been applied to location privacy [11, 12, 13] by obfuscating the user's position in such a way to confuse it with the positions of other $k-1$ users. Location k-anonymity offers high levels of privacy, because it protects the user's identity. However, since k-anonymity does not permit the identification of the user, it is not applicable in services in which the user authenticates, e.g. payable services or location-based social networks. In addition, they require the presence of $k - 1$ users in the proximity, that could be missing, and a central anonymizer, that could not be fully trusted by the users.

A different and promising approach is *data obfuscation* [14, 15]. The aim is not to reach anonymity, but rather to artificially reduce the precision of location data before disclosing it. In this way, the service can still be delivered, but an adversary cannot infer other sensitive information. We focus on obfuscation through *noise perturbation* [16, 17]. An underrated problem in the literature is how to choose a suitable noise to effectively perturb data. We found that, if noise is not chosen properly, perturbation will not resist to attacks based on statistical inference. In particular, an obfuscation operator must offer a spacial *uniformity* of probability. Such a requirement is often postulated, rather than fulfilled, by state-of-the-art perturbation methods.

We propose UniLO, a location obfuscation operator able to guarantee uniformity even in the presence of imprecise location measurements. UniLO does not require a centralized and trusted obfuscator. We deal with service differentiation by proposing and comparing three UniLO-based obfuscation algorithms offering multiple contemporaneous levels of privacy. Finally, we experimentally prove that UniLO outperforms state-of-the-art perturbation algorithms both in terms of utility and resistance against inference attacks. This paper extends our previous work [18] with multiple levels of privacy and an in-depth analysis of the utility and the resistance against inference attacks. All the simulations scripts of the present paper can be downloaded from [19].

The rest of the paper is organized as follows. Section 2 analyzes some related works and the differences with UniLO techniques. Section 3 introduces some basic concepts concerning the system model and the terminology. Section 4 formally describes the agnostic adversary model, the concept of uniformity, and a way to quantify it. Section 5 presents the basic UniLO operator and show its properties in terms of uniformity. Section 6 presents the problem of offering multiple levels of privacy and three algorithms to adapt UniLO in this sense. Section 7 evaluates UniLO algorithms in terms of utility on an example location-based service. Section 8 evaluates UniLO algorithms in terms of resistance against inference attacks. Finally, the paper is concluded in Section 9.

2. Related works

Approaches for location privacy can be roughly divided in *identity protection* and *data protection*. Identity protection avoids the re-identification of anonymous users. Data protection avoids the disclosure of precise locations.

2.1. Identity-protection approaches

Gruteser and Grunwald [11] first applied *k-anonymity* approach in location-based services. The proposed solution involves the subdivision of the map in quadrants with different granularities. The user does not release his precise position, but a quadrant of the grid containing other $k-1$ users, in such a way his identity is confused with theirs. The k-anonymity approach is broadly used in many research works [12, 13, 20, 21, 22]. However, these methods require the presence of $k-1$ users in the proximity, that could be missing, and a central anonymizer, that could not be fully trusted by the users. In addition, they do not permit the identification of the user, so that they are not applicable in those cases in which the user authenticates himself, e.g. payable services or location-based social networks. Our approach aims at protecting the position, rather than the identity, and it is suitable also for authenticated users.

[23] and [24] approach the problem of *trajectory k-anonymity*, offering methods to protect user's privacy in continuous tracking systems. Although it could be extended in that sense, the present work focuses on single-position queries, as they encompass a wide range of location-based applications.

A problem complementary to anonymity is *pseudonym unlinkability* in tracking systems, usually approached with the technique of *mix zones* [25, 26, 27]. Mix zones are areas of the map where users cannot be tracked and change their pseudonym. By carefully placing and dimensioning such mix zones it is possible to thwart the adversary from linking two consecutive pseudonyms of the same user.

2.2. Data-protection approaches

Location obfuscation aims at reducing the precision of location data before disclosing it. This can be done by adding noise [14] (*noise-based obfuscation*) as well as with other methods, for example by replacing the exact position with a quadrant of a grid [15]. Research on this topic has focused mainly on what kind of service can be delivered with imprecise positions [15, 28, 29, 30]. The problem of generating such imprecise positions in a proper way is often underrated. In particular, the uniformity of the noise-based obfuscation is often postulated, rather than evaluated. As a result, the proposed solutions turn out to be poorly resistant against inference attacks. In this paper we focus entirely on noise-based obfuscation, so from now on we will omit the "noise-based" specification as implicit.

Ardagna et al. [14] proposed a set of obfuscation operators that perturb the location: radius enlargement, radius restriction, center shift. These operators transform a measurement area into an obfuscated one. Our approach guarantees both more private and more useful obfuscated areas. More private because UniLO noise significantly increases the uniformity of the resultant privacy areas. More useful because we always guarantee that the privacy areas contain the user's position. A service provider can thus rely on more powerful assumptions and offer more quality of service. In addition, in [14] the resistance against attacks relies on the fact that the adversary is unaware of the privacy preference of the user. This could be an optimistic assumption, which features a form of "security by obscurity" that should be avoided [31].

Krumm [16] surveyed many different obfuscation methods and applied them to real-life GPS traces. The objective was to prevent an attacker from inferring users' home positions. Krumm tried also a noise-based method, which involved noise with a Gaussian magnitude. He found that this method requires a high quantity of noise ($\sigma = 5\,\mathrm{Km}$) in order to effectively prevent inference attacks. Our approach offers higher levels of uniformity, and reduces the amount of noise needed to resist to inference attacks.

Dürr et al. [17] proposed an obfuscation approach with multiple levels of privacy. They build different "shares" which are random vectors concatenated to the user's position. They store the shares in different servers to avoid a single point of trust. Each service provider reconstructs the position by "fusing" one or more shares from one or more servers. The privacy level is proportional to the number of shares the service provider is allowed to access. The authors generate the shares as random vectors having uniform magnitude. Our obfuscation operators guarantee more resistance against inference attacks.

Inspired by differential privacy [32], Andrés et al. [33] introduced the concept of *ϵ-geo-indistinguishability*. The idea is that the user obtains more privacy in the surroundings of his true position, and less farther. To achieve this, they perturb the true position with a 2-dimensional extension of the Laplacian noise. Such a noise is highly non-uniform. As a consequence, geo-indistinguishability offers far less resistance to inference attacks compared to UniLO.

Other notable obfuscation-based approaches are [28, 29, 30]. All these works postulate uniformity rather than providing for it. In contrast, our approach offers guarantees on the obfuscation uniformity, even in presence of imprecise location measurements.

Another research track [34, 35, 36, 37, 38] applies *private information retrieval* (PIR) techniques to protect user's location. The objective is to provide a location-based service without disclosing the user's location at all. While PIR approaches offer strong and provable security, they are quite resource-demanding at the server side. Actually, they require complex, computational intensive cryptographic operations or the employment of trusted hardware architectures. In contrast obfuscation techniques only provide for statistical guarantees in terms of privacy, but they are more affordable for the service provider.

3. System Model

In our system, a *user* is someone whose location is measured by a *sensor*. A *service provider* is an entity that receives the user's location in order to provide for a *location-based service*. The user applies an *obfuscation operator* to location information prior to releasing it to the service provider. The obfuscation operator purposefully reduces the precision to guarantee a certain privacy level. Such a precision is defined by the user and reflects his requirements in terms of privacy. The more privacy the user requires, the less precision the obfuscation operator returns.

A location measurement is affected by an intrinsic error that limits its precision. Such an error depends on several factors including the localization technology, the quality of the sensor, the environment conditions. If the measurement error is small compared to the obfuscation, as it happens in professional GPS receivers, it can be approximated to zero. Otherwise, as it happens in cheap GPS receivers mounted on smartphones, or in Wi-Fi and cellular positioning, we cannot neglect it. This implies that the location cannot be expressed as a geographical point but rather as a neighborhood of it. We assume that locations are always represented as *planar circular areas*, because it is a good approximation for many location techniques [14, 39, 40]. We will use the notation $A = \langle \mathbf{C}, r \rangle$ to mean that A is a circle with center $\mathbf{C}$ and radius r. A measurement area (Fig. 1) is defined as follows:

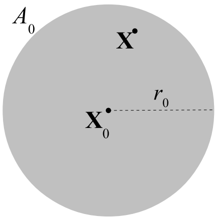


Figure 1: Measurement area

Definition 1 (Measurement area). *Let* $\mathbf{X}$ *be the* actual position *of the user. A* measurement area *is a circle* $A_0 = \langle \mathbf{X}_0, r_0 \rangle$*, such that* $\mathbf{X} \in A_0$ (Accuracy Property)*. We call* $\mathbf{X}_0$ *the* measured position *and* r_0 *the* error radius*.*

The Accuracy Property guarantees that the measurement area contains the user, or, equivalently, that the distance $\overline{\mathbf{X}\mathbf{X}_0}$ does not exceed the error radius. We assume that the error radius is constant over time. This means either that the precision does not change over time, or that we consider the worst-case precision.

A user specifies his privacy preference in terms of a *privacy radius* $r_1 > r_0$, meaning that he wishes to be located with a precision not better than r_1. The privacy radius is quite an easy metric to be understood by the users. This improves the overall usability of the obfuscation system. The task of an obfuscation operator is to produce a *privacy area* A_1 with radius r_1, appearing to the provider as a measurement area with a lower precision.

Definition 2 (Privacy area). *Let* $\mathbf{X}$ *be the* actual *position of the user. A* privacy area *is a circle* $A_1 = \langle \mathbf{X}_1, r_1 \rangle$ *with* $r_1 > r_0$*, such that* $\mathbf{X} \in A_1$ (Accuracy Property)*. We call* $\mathbf{X}_1$ *the* obfuscated position *and* r_1 *the* privacy radius*.*

Definition 3 (Obfuscation operator). *Let* A_0 *be a measurement area, and* $r_1 > r_0$ *a privacy radius. An obfuscation operator* $\mathrm{obf}(\cdot)$ *transforms* A_0 *into a privacy area* A_1*:*

$$A_1 = \mathrm{obf}\left(A_0\right) \tag{1}$$

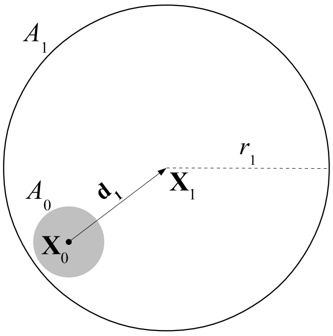
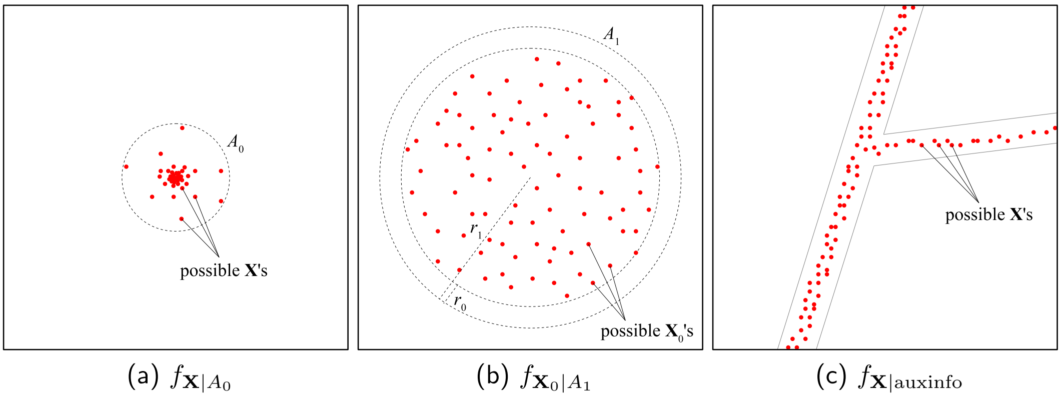


Figure 2: Obfuscation and shift vector



(a) $f_{\mathbf{X}|A_0}$ (b) $f_{\mathbf{X}_0|A_1}$ (c) $f_{\mathbf{X}|\text{auxinfo}}$

Figure 3: Adversarial information

With reference to Fig. 2, in order to produce a privacy area, the obfuscation operator applies both an enlargement and a translation of the measurement area. The enlargement aims at decreasing the precision and thus achieving the desired privacy radius. The translation is made through a randomly selected *shift vector* $\mathbf{d}_1$, i.e., $\mathbf{X}_0 + \mathbf{d}_1 = \mathbf{X}_1$. The obfuscator has to keep the shift vector secret.

The enlargement and translation operations must be such that, when composed, the resulting privacy area satisfies the Accuracy Property. We state the following:

Proposition 1. *A privacy area A_1 fulfills the Accuracy Property iff:*

$$\|\mathbf{d}_1\| \leq (r_1 - r_0) \tag{2}$$

Proof. The proof stems directly from geometrical considerations (cfr. Fig. 2).

4. Agnostic adversary and uniformity index

For the scope of the present paper, every service provider receiving an obfuscated position is a potential adversary. We assume the adversary knows the privacy area and the error radius. She aims at discovering the actual user's position. Since it cannot be known with infinite precision, the result of the attack will have a probabilistic nature. From now on, we will use the notation $f_{a|b}$ to refer to the conditional probability density function of the random variable a given the information b.

Three pieces of information could help the adversary: (1) the employed localization technology; (2) the employed obfuscation operator; (3) other auxiliary information. They are modeled by three probability densities in $\mathbb{R}^2$:

1. The *error density* $f_{\mathbf{X}|A_0}$ (Fig. 3a), which describes the actual position given a measurement of it. We have no control over this density. Our obfuscation operators are supposed to be unaware of it. As a consequence, they are flexible enough to be applicable with every kind of density (i.e. with every kind of measurement technology), as long as it is bounded by the error radius. On the contrary, the adversary is supposed to know the error density. For example she should assume it is Gaussian, as it is usually done in GPS measurements [41].

2. The *obfuscation density* $f_{\mathbf{X}_0|A_1}$ (Fig. 3b), which describes the measured position given an obfuscated version of it. We can control this density, and this is our main weapon against the adversary. The adversary can compute this density by analyzing the obfuscation operator, which is considered to be publicly known. She starts from the inverse density $f_{A_1|\mathbf{X}_0}$, which describes the possible output of the obfuscation operator, and then applies Bayesian inference. In noise-based obfuscation, $f_{A_1|\mathbf{X}_0}$ depends on the density of $\mathbf{d}_1$, while $f_{\mathbf{X}_0|A_1}$ depends on the density of $-\mathbf{d}_1$.
3. The *auxiliary information density* $f_{\mathbf{X}|\text{auxinfo}}$ (Fig. 3c), which describes the position given a set of auxiliary information. Examples of auxiliary information are the street map where the users are moving, the average distribution of the users in a given city at a given hour, the average daily behavior of a particular user, etc. We have no control over this density and it is hard to make hypotheses on it. The adversary could have much or little information.

Dealing with auxiliary information is a recurring problem in privacy topics [9, 32]. The adversary could use it in several ways. A simple attack consists in cutting away the zones of the privacy area where the user cannot be, basing on a public street map. In this way, the adversary restricts the effective privacy area of the user. A common approach [42, 43] involves enlarging the privacy radius to "compensate" the area loss. However, this comes at a price on data utility. Depending on the street map, the privacy radius could become much larger, and this could make it useless for the aim of providing the service. In this paper we preferred not to do that, and let the user free of choosing his final privacy radius. In practice, we provide an assurance on the radius, rather than on the area. We believe that a radius (e.g. being cloaked within 500 m) is a more understandable and usable privacy metric than an area (e.g. being cloaked inside 1 Km^2, irregularly shaped on a street map).

We cannot suppose how much auxiliary information the adversary knows, and we cannot make the adversary "forget" it. Therefore, our true aim is to give her no additional information other than the simple one: "$\mathbf{X}$ is inside A_1." We model such a requirement with the concept of *ideal obfuscation*:

Definition 4 (Ideal obfuscation).

$$f_{\mathbf{X}|\text{auxinfo},A_1}(x,y) = f_{\mathbf{X}|\text{auxinfo},\mathbf{X}\in A_1}(x,y) \tag{3}$$

An obfuscator which performs ideal obfuscation is an *ideal obfuscator*. Note that "given A_1" in the left term of Equation 3 differs from "given $\mathbf{X} \in A_1$" in the right term. The former means that the adversary knows the privacy area generated by the obfuscation operator. The latter means that the adversary knows that the user is inside an area A_1, not necessarily generated by an obfuscation operator. Intuitively, in order Equation 3 to hold, the obfuscation operator should produce a privacy area in such a way that the actual position is uniformly distributed inside it. We state the following:

Definition 5 (Uniformity Property). *A privacy area A_1 fulfills the Uniformity Property iff $f_{\mathbf{X}|A_1}(x,y)$ is uniform over A_1. An obfuscator fulfills the Uniformity Property iff all the produced privacy areas fulfill the Uniformity Property.*

Theorem 1. *An obfuscator which offers Uniformity Property is ideal.*

PROOF. From the definition of conditional probability, we have that:

$$f_{\mathbf{X}|\text{auxinfo},\mathbf{X}\in A_1} = \begin{cases} \frac{f_{\mathbf{X}|\text{auxinfo}}}{\iint_{A_1} f_{\mathbf{X}|\text{auxinfo}}\,\mathrm{d}x\mathrm{d}y} & \text{in } A_1 \\ 0 & \text{outside} \end{cases} \tag{4}$$

On the other hand, if Uniformity Property holds:

$$f_{\mathbf{X}|A_1} = \begin{cases} \frac{1}{\text{size}(A_1)} & \text{in } A_1 \\ 0 & \text{outside} \end{cases} \tag{5}$$

Combining (5) with the auxiliary information:

$$\begin{aligned} f_{\mathbf{X}|\text{auxinfo},A_1} &= \frac{f_{\mathbf{X}|\text{auxinfo}} \cdot f_{\mathbf{X}|A_1}}{\iint_{\mathbb{R}^2} f_{\mathbf{X}|\text{auxinfo}} \cdot f_{\mathbf{X}|A_1}\,\mathrm{d}x\mathrm{d}y} \\ &= \begin{cases} \frac{f_{\mathbf{X}|\text{auxinfo}}}{\iint_{A_1} f_{\mathbf{X}|\text{auxinfo}}\,\mathrm{d}x\mathrm{d}y} & \text{in } A_1 \\ 0 & \text{outside} \end{cases} \\ &= f_{\mathbf{X}|\text{auxinfo},\mathbf{X}\in A_1} \end{aligned} \tag{6}$$

Theorem 1 tells us that Uniformity Property is important regardless of the auxiliary information the adversary has, because it gives her no additional one.

No obfuscation system can provide uniformity against an adversary holding some auxiliary information. This is because the adversary will have a non-uniform $f_{\mathbf{X}|\text{auxinfo}}$, which is an a-priori probability density of the user's position. In order to study the uniformity of a generic obfuscation operator, we suppose an adversary *who ignores any auxiliary information*, i.e. whose $f_{\mathbf{X}|\text{auxinfo}}$ is uniform over the whole world map. We will call such an adversary the *agnostic adversary*. The agnostic adversary is purely theoretic, however: (a) it permits us to study the uniformity of obfuscation operators; and (b) if an obfuscation operator enjoys Uniformity Property against the agnostic adversary, it also gives no additional information to a real adversary (it is *ideal*).

4.1. Uniformity index

We use the agnostic adversary to measure the uniformity of a generic obfuscation method. Basing on the error density and the obfuscation density, the agnostic adversary computes the *pdf* $f_{\mathbf{X}|A_1}(x,y)$ of the user's position. After that, she defines a *confidence goal* $c \in (0,1]$ and computes the smallest area $\hat{A}^c \subseteq A_1$ which contains the user with a probability c. We call this area the *smallest c-confidence area.*

Definition 6 (Smallest c-confidence area).

$$\hat{A}^c = \arg\min_{A\in\mathcal{A}^c} \{\text{size}(A)\} \tag{7}$$

where:

$$\mathcal{A}^c = \left\{A | A \subset \mathbb{R}^2, P\left\{\mathbf{X} \in A | A_1\right\} = c\right\} \tag{8}$$

$$P\left\{\mathbf{X} \in A | A_1\right\} = \iint_A f_{\mathbf{X}|A_1}(x,y)\,\mathrm{d}x\mathrm{d}y \tag{9}$$

The smallest c-confidence area is the adversary's most precise estimation of the actual position, and it will cover the zones where $f_{\mathbf{X}|A_1}(x, y)$ is more concentrated. The adversary can find it by means of a Monte Carlo approach. First, she synthesizes many "measurement-plus-obfuscation" operations, finding many tuples with the form:

$$\langle \text{actual pos.}, \text{measured pos.}, \text{obfuscated pos.} \rangle$$

Then, she selects only those tuples whose obfuscated position matches with the one she wants to deobfuscate. The actual positions of the selected tuples follows $f_{\mathbf{X}|A_1}(x, y)$. Finally, the adversary determines the smallest c-confidence area by connecting the zones having highest concentrations. The smaller $\hat{A}^c$, the more precisely the adversary locates the user. A good obfuscation operator should keep $\hat{A}^c$ as larger as possible. This is done by making the obfuscation as uniform as possible. The best case occurs when the Uniformity Property is fulfilled, and the obfuscator is ideal. Unfortunately, it is impossible to provide for Uniformity Property in the general case. As an example, think about a measurement with a Gaussian error density, followed by a small obfuscation ($r_1 \approx r_0$). Independently of which noise the obfuscator adds, the final *pdf* will be dominated by the Gaussian component, thus it will be strongly non-uniform. Depending on the error density, every obfuscation will produce some "irregularities," over which the *pdf* is not perfectly uniform. Thus, we developed a way to *quantify* the uniformity of an obfuscation.

Another way to state the Uniformity Property is the following:

Proposition 2. *A privacy area A_1 fulfills the Uniformity Property iff:*

$$\forall A \subseteq A_1, P\{\mathbf{X} \in A | A_1\} = \frac{\text{size}(A)}{\text{size}(A_1)} \tag{10}$$

That is, each region of the privacy area contains the user with a probability proportional to its size. In such a case:

$$\text{size}(\hat{A}^c) = c \cdot \text{size}(A_1) \tag{11}$$

Otherwise:

$$\text{size}(\hat{A}^c) \leq c \cdot \text{size}(A_1) \tag{12}$$

The uniformity can be quantified by means of Eq. 12, by measuring how much, for a given c, size$(\hat{A}^c)$ gets close to $c \cdot$ size(A_1). We define the following *uniformity index* by fixing $c = 90\%$:

Definition 7 (Uniformity index).

$$\text{unif}(A_1) = \frac{\text{size}(\hat{A}^{90\%})}{90\% \cdot \text{size}(A_1)} \tag{13}$$

The uniformity index ranges from 0% (worst case), if the user's position is perfectly predictable, to 100% (best case), if the user's position is perfectly uniform. A uniformity index of 100% is necessary and sufficient for the Uniformity Property.

Uniformity index is proportional to the lack of precision of the attack. For example, if a privacy area of $400\,\text{m}^2$ has a uniformity index of 80%, the agnostic adversary cannot find his position (with 90% confidence) with more precision than $80\% \cdot 90\% \cdot 400 = 288\,\text{m}^2$. Note that the uniformity index is not our *privacy metric*, but rather an estimator of the obfuscation resistance. Our true privacy metric is still the privacy radius, which is chosen by the user as a preference.

4.2. Time-correlation of user's position

An adversary could use *past* and *present* privacy areas in order to infer the current position. For example, she could take two privacy areas generated very close in time and locate the user inside their intersection, supposing that he has not moved too much in the meanwhile. Another possibility is to do the same with two privacy areas generated at times when the user visits a recurring place (e.g. his home at 8:00 and at 18:00). All these attacks are based on the fact that positions at different instants are similar or strictly correlated (*time-correlation attacks*). This problem is orthogonal to the one of providing uniformity within the single privacy area, and can be addressed separately. The simplest way to counteract time-correlation attacks is to provide for *reuse policies*, i.e. algorithms to reuse past privacy areas in certain cases. For example, the obfuscator could reuse the same privacy area in case of two queries close in time and space, or in case of queries from a recurring place. In this way we avoid the possibility of intersections. The reuse policies should be tailored and evaluated depending on the kind of user: his average query frequency, his daily mobility model, etc. Some simulators can help in doing this [44]. Though these aspects are interesting, we did not investigate them in the present paper. We focused on developing a set of high-resistant obfuscation operators that are flexible enough to be extended with reuse policies. The operators we present are ready to be deployed if the queries can be assumed to be uncorrelated (e.g. randomly walking users making sporadic queries), and they should be integrated with reuse policies otherwise.

5. UniLO obfuscation operator

UniLO (**Uni**form **L**ocation **O**bfuscation) [18] adds to the measured position a shift vector $\mathbf{d}_1 = (\mu \cos\varphi, \mu \sin\varphi)$, where μ is the magnitude and φ is the angle. μ and φ have the following probability densities (Fig. 4):

$$f_\Phi(\varphi) = \begin{cases} \frac{1}{2\pi} & \varphi \in [0, 2\pi) \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

$$f_M(\mu) = \begin{cases} 2\mu/(r_1 - r_0)^2 & \mu \in [0, r_1 - r_0] \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

These densities produce shift vectors with magnitude less than or equal to $r_1 - r_0$, and a perfectly uniform spacial probability density. This produces a good level of uniformity of $f_{\mathbf{X}|A_1}$. However, remind that $f_{\mathbf{X}|A_1}$ also depends on the error density, over which we have no control. So $f_{\mathbf{X}|A_1}$ will not be perfectly uniform in the general case.

UniLO fulfills the following properties:

- *Accuracy Property.* The privacy area always contains the user (Theorem 2).
- *High uniformity index.* UniLO outperforms all the other noise shapes used in the literature in terms of uniformity index, for all values of r_1/r_0.
- *Uniformity Property as* $r_0 \to 0$. With highly precise sensors, UniLO tends to be an ideal obfuscator. (Theorem 3).

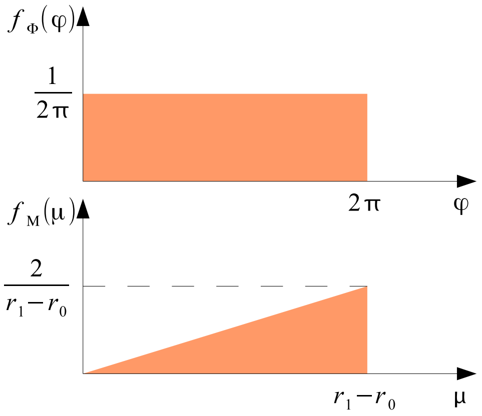


Figure 4: φ and μ *pdf*'s of a UniLO vector

Theorem 2. UniLO *fulfills Accuracy Property.*

Proof. By construction, $\|\mathbf{d}_1\| \leq r_1 - r_0$. Hence, from Prop. 1, Accuracy holds.

Theorem 3. *As* $r_0 \rightarrow 0$, UniLO *fulfills Uniformity Property.*

Proof. If $r_0 \rightarrow 0$, A_0 will narrow to a point, with $\mathbf{X} \equiv \mathbf{X}_0$, and the probability density of the magnitude in Eq. 15 will become:

$$f_M(\mu) = \begin{cases} 2\mu/r_1^2 & \mu \in [0, r_1] \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

To show the Uniformity, we have to pass from the polar representation to the Cartesian representation. So we have to transform the densities $f_M(\mu)$, $f_\Phi(\varphi)$ to the joint density $f_{X,Y}(x,y)$. In order to perform this variable change, we equal the areas of the rectangle spaced by dx and dy, and of the annulus sector spaced by dμ and dφ:

$$\mathrm{d}x\mathrm{d}y = \frac{(\mu + \mathrm{d}\mu)^2 - \mu^2}{2}\mathrm{d}\varphi = \mu \cdot \mathrm{d}\mu\mathrm{d}\varphi \tag{17}$$

Then, we equal the probabilities inside them:

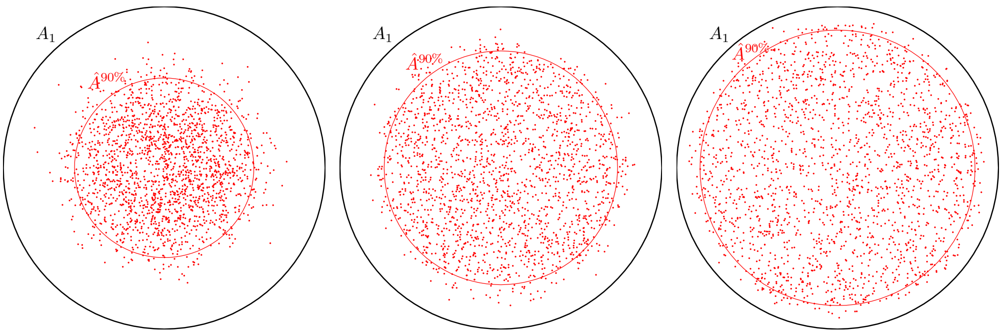
$$f_{X,Y}(x,y)\,\mathrm{d}x\mathrm{d}y = f_M(\mu)\,\mathrm{d}\mu \cdot f_\Phi(\varphi)\,\mathrm{d}\varphi \tag{18}$$

$$= \begin{cases} 2\mu/r_1^2\mathrm{d}\mu \cdot \frac{1}{2\pi}\mathrm{d}\varphi & \mu \leq r_1 \\ 0 & \text{otherwise} \end{cases} \tag{19}$$

From Equations 17 and 19, we have:

$$f_{X,Y}(x,y) = \begin{cases} \frac{1}{r_1^2\pi} & \sqrt{x^2+y^2} \leq r_1 \\ 0 & \text{otherwise} \end{cases} \tag{20}$$

which is spatially uniform in A_1.



(a) $r_1/r_0 = 2$ (b) $r_1/r_0 = 4$ (c) $r_1/r_0 = 10$

Figure 5: UniLO spatial distribution (2000 Monte Carlo runs)

We will use the following notation:

$$\mathbf{d}_1 = \text{UniLO}(r_1, r_0)$$

to say that $\mathbf{d}_1$ is a shift vector created by the UniLO operator with privacy radius r_1 and precision radius r_0. UniLO operator will be our basic block to build more complex obfuscators.

We evaluated the uniformity index of UniLO on simulated location measurements. The error on the location measurements was assumed to follow a Gaussian distribution, as it is usually done in GPS [41]. We truncated the distribution at $r_0 = 3\sigma$, so that no sample falls outside the measurement area. Such a truncated Gaussian distribution differs from the untruncated one for only 1% of samples. The tests aim at evaluating the uniformity of UniLO with respect to the ratio r_1/r_0 (*radius ratio*).

Figure 5 shows the statistical distribution of $\mathbf{X}$ in A_1 for different values of the radius ratio. We note that the distribution tends to be perfectly uniform as $r_1/r_0 \to \infty$. The inner areas are the smallest 90%-confidence areas.

We compared UniLO with other state-of-the-art obfuscation noises[1]:

- Gaussian noise, used for modeling 2-dimensional measurement errors.

- Krumm's noise, used by Krumm to perturb GPS data [16]. Krumm's noise has a uniformly distributed angle and a magnitude drawn from a Gaussian distribution.

- Andrés' noise, used by Andrés et al. [33]. This noise is a 2-dimensional extension of the Laplacian noise, and it is used to achieve *geo-indistinguishability*. Refer to [33] for further information.

- Dürr's noise, used by Dürr et al. in their "*a-posteriori* share generation algorithm" [17]. This is the simplest 2-dimensional noise: it has a uniformly distributed

[1] In case of unbounded noises (e.g. Gaussian), we fulfilled Accuracy Property by truncating their magnitude at $(r_1 - r_0)$. To make meaningful comparisons, we tailored the parameters in such a way to truncate always 1% of the samples. Namely, we tailored $\sigma = (r_1 - r_0)/3$ for Gaussian noise, $\sigma = (r_1 - r_0)/2.6$ for Krumm's noise [16], and $\epsilon = 6.5/(r_1 - r_0)$ for Andrés' noise [33].

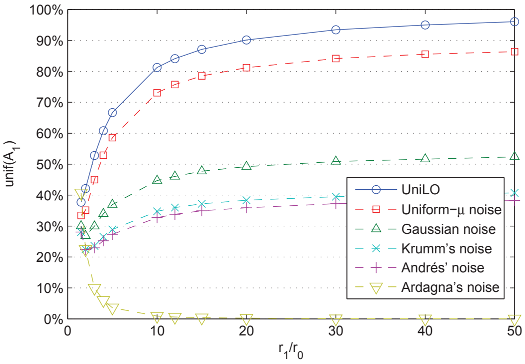


Figure 6: unif (A_1) wrt the radius ratio (500K Monte Carlo runs for each point)

angle and a uniformly distributed magnitude. We compare UniLO with this because it is the obfuscation method most similar to ours.

- Ardagna's noise, used by Ardagna et al. in their location obfuscation operators [14]. These are a set of obfuscation operators that reduce/enlarge/shift the measurement area to produce the privacy area. The user expresses his privacy preference in terms of *final relevance*, which is assumed to be unknown by the adversary. With "Ardagna's noise" we refer here to the cumulative effect of (a) the random selection of the final relevance, (b) the random selection of the obfuscation operator, and (c) the random selection of the shift angle. These obfuscation operators do not guarantee the Accuracy Property, and the user could be outside the privacy area. Refer to [14] for further information.

Figure 6 shows the uniformity indexes of the noises. We can see that UniLO outperforms all the other noises for all the radius ratii. In the average case, Ardagna's noise is particularly easy to predict, because it has not been designed to thwart statistical attacks. On the other hand, it enjoys quite a high uniformity for very small privacy radii ($r_1 < 2r_0$). However, such an improved uniformity is obtained at the cost of violating the Accuracy Property, and thus possibly degrading the utility of the service. Krumm's and Gaussian noises are not so good at obfuscating. We believe this is the reason why Krumm needed a surprisingly high quantity of noise ($\sigma = 5\,\text{Km}$) to effectively withstand inference attacks [16]. Andrés' noise for geo-indistinguishability is quite predictable too.

6. Multiple levels of privacy

A user may require different privacy radii for different services. He can require high levels of privacy for some services, for instance a friend-finder service, and small levels of privacy for others, for instance safety-related services. In general, an obfuscator must offer a user *a set* of N possible privacy radii, and must create *a set* of N random shift vectors, one for each privacy radius. The error radius of the sensor can be considered as the minimum privacy radius. In other words, the smallest privacy area is the measurement itself.

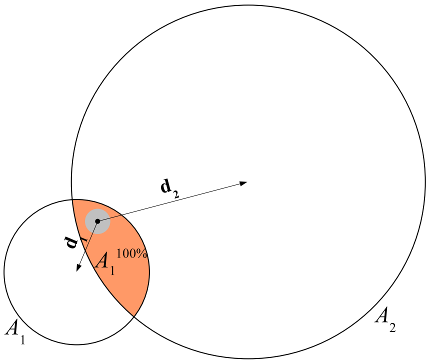


Figure 7: Collusion attack

Let $\rho = \{r_0, r_1, r_2, \cdots, r_N\}$, with $r_0 < r_1 < r_2 < \cdots < r_N$, be the *privacy radius set*, i.e. the set of the privacy radii provided by the obfuscator. Then:

- $\{\mathbf{d}_i : i = 1, 2, \cdots, N\} = \delta$ is the *shift vector set*,
- $\{\mathbf{X}_i = \mathbf{X}_0 + \mathbf{d}_i : i = 1, 2, \cdots, N\}$ is the *center set*,
- $\{A_i = \langle \mathbf{X}_i, r_i \rangle : i = 1, 2, \cdots, N\}$ is the *privacy area set*.

We will refer to r_{i-1} and r_{i+1} as, respectively, the *previous* and the *successive* privacy radii of r_i. The same convention holds for shift vectors and privacy areas. We will use the notation $\hat{A}_i^c$ to refer to the smallest c-confidence area found by an agnostic adversary able to access to the i-th privacy level.

6.1. On collusion attack

A subtle attack is possible when two or more service providers collude. Let us suppose that a service provider knowing A_1 colludes with a service provider knowing A_2. If the shift vectors are not chosen wisely, the adversaries can intersect A_1 and A_2 (Fig. 7) to find a smaller area containing the user. To avoid this possibility, an obfuscator should force each privacy area to enclose all the smaller ones. We state the following:

Definition 8 (Inclusion Property). *A privacy area A_i ($i \geq 2$) fulfills the Inclusion Property iff $A_{i-1} \subset A_i$. An obfuscator fulfills the Inclusion Property iff all the produced privacy areas fulfill the Inclusion Property.*

With Inclusion Property, we assure that a group of adversaries (even unlimited in number) has not more power than the most powerful of them, i.e. the one accessing to the smallest privacy area.

If a privacy area must enclose the previous one, the distance between the centers must not be larger than the radii difference. Formally:

Proposition 3. *A privacy area A_i ($i \geq 2$) fulfills the Inclusion Property iff:*

$$\|\mathbf{d}_i - \mathbf{d}_{i-1}\| \leq (r_i - r_{i-1}) \tag{21}$$

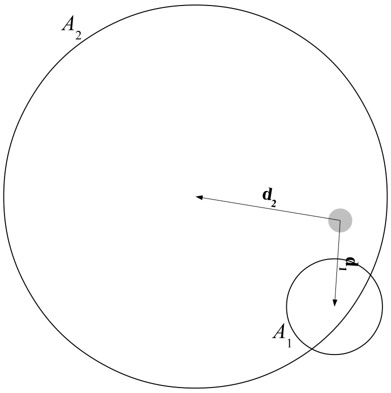


Figure 8: IV-UniLO example

It is worth to stress that the Inclusion Property is not mandatory. In particular, it can be released if both the system prevents service providers from accessing different privacy levels, and different service providers do not collude. The Inclusion Property lowers the uniformity index of the privacy areas.

6.2. UniLO *for multiple levels of privacy*

We will now adapt the basic UniLO operator for offering a set ρ of N shift vectors. The simpler solution is to apply N times UniLO, obtaining N shift vectors independent of each other. Formally:

$$\mathbf{d}_i = \text{UniLO}(r_i, r_0)\ \forall i$$

We will refer to this solution as *Independent Vectors* UniLO (IV-UniLO). Figure 8 shows an example with $\rho = \{r_0, r_1 = 4r_0, r_2 = 16r_0\}$.

IV-UniLO trivially fulfills the Accuracy Property for all the privacy areas. It also offers a good level of uniformity, especially for large privacy radii ($r_i \gg r_0$). Figure 8 shows that A_2 does not enclose A_1. Thus, IV-UniLO does not fulfill the Inclusion Property and does not defend against collusion.

6.3. VC-UniLO*: Vector Chain* UniLO

The idea of VC-UniLO is to fulfill Inclusion by assuring that the distance between $\mathbf{X}_1$ and $\mathbf{X}_2$ never goes beyond $(r_2 - r_1)$. To do this, we create $\mathbf{d}_2$ as the sum of $\mathbf{d}_1$ and an *incremental vector* $\mathbf{d}_{1,2}$, which is a random vector with maximum magnitude $(r_2 - r_1)$. The incremental vector represents in fact the distance between $\mathbf{X}_1$ and $\mathbf{X}_2$. The same procedure is repeated for $\mathbf{d}_3 \cdots \mathbf{d}_N$. In this way, we fulfill both Accuracy and Inclusion, as stated by the following two Theorems:

Theorem 4. VC-UniLO *fulfills the Accuracy Property for all the privacy areas.*

Proof. We prove this by induction. From Theorem 2, A_1 fulfills Accuracy. If Accuracy holds for A_{i-1}, then $\|\mathbf{d}_{i-1}\| \leq (r_{i-1} - r_0)$. By construction $\|\mathbf{d}_{i-1,i}\| \leq (r_i - r_{i-1})$. It

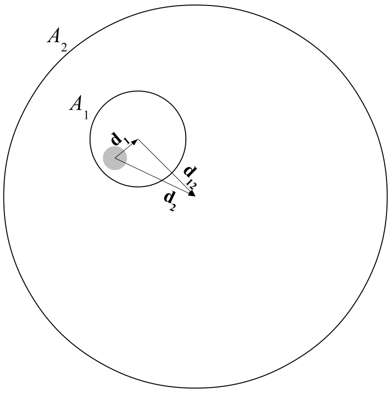


Figure 9: VC-UniLO example

follows that:

$$\begin{aligned} \|\mathbf{d}_i\| &\leq \|\mathbf{d}_{i-1}\| + \|\mathbf{d}_{i-1,i}\| \\ &\leq (r_{i-1} - r_0) + (r_i - r_{i-1}) \\ &= (r_i - r_0) \end{aligned}$$

Hence, from Prop. 1, Accuracy Property holds for all privacy areas.

Theorem 5. VC-UniLO *fulfills the Inclusion Property for all the privacy areas.*

Proof. We consider the generic privacy area A_i. By construction, $\|\mathbf{d}_i - \mathbf{d}_{i-1}\| = \|\mathbf{d}_{i-1,i}\| \leq (r_i - r_{i-1})$. Hence, from Prop. 3, Inclusion Property holds for all privacy areas.

For $\mathbf{d}_{i-1,i}$ we choose vectors created by UniLO operator:

$$\mathbf{d}_{i-1,i} = \text{UniLO}(r_i, r_{i-1})$$

This is the simplest choice and still offers a good level of uniformity for A_i. To sum up, VC-UniLO algorithm creates the shift vectors with the following formula:

$$\mathbf{d}_i = \begin{cases} \text{UniLO}(r_1, r_0) & i = 1 \\ \mathbf{d}_{i-1} + \text{UniLO}(r_i, r_{i-1}) & i > 1 \end{cases}$$

The i-th shift vector is created by concatenating vectors, hence the name *Vector Chain.* Figure 9 shows an example with $\rho = \{r_0, r_1 = 4r_0, r_2 = 16r_0\}$.

VC-UniLO defends against collusion but offers a lower uniformity index than IV-UniLO. The problem is that $\|\mathbf{d}_i\|$ $(i > 1)$ has a low probability of being large. In fact, $\mathbf{d}_i$ is the sum of two vectors ($\mathbf{d}_{i-1}$ and $\mathbf{d}_{i-1,i}$) and its magnitude gets close to the maximum (r_i) only if the vectors are aligned on the same direction and both have high magnitudes. This is a very rare event. In the majority of cases, $\mathbf{d}_i$ will have a small magnitude. So the user will be near the center with greater probability than near the borders. This limits the uniformity of the resultant privacy area A_i.

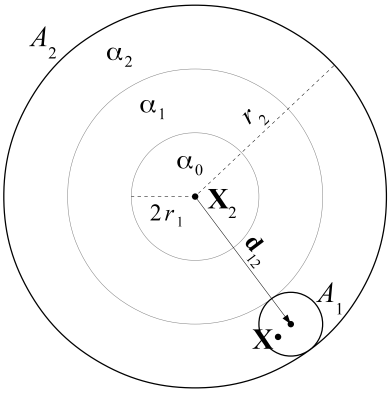


Figure 10: p-Partitionability regions

Forcing $\mathbf{d}_{i-1,i}$ to have the same direction as $\mathbf{d}_{i-1}$ is not a viable strategy, because it would make the centers $\mathbf{X}_i$, $\mathbf{X}_{i-1}$ and $\mathbf{X}_0$ aligned. Therefore, an adversary knowing A_{i-1} and A_i would automatically have a preferred direction where to find A_0. In general, $\mathbf{d}_{i-1,i}$ should be independent of the value of $\mathbf{d}_{i-1}$.

6.4. DVC-UniLO*: Discrete Vector Chain* UniLO

The idea of DVC-UniLO is to improve the uniformity index of VC-UniLO by changing the way the incremental vectors are built. We will first introduce the *p-Partitionability Property*, which is a weaker form of Uniformity, and then present DVC-UniLO, which offers such a property.

Ensuring the Uniformity Property is a hard problem, since it states that *all the possible* regions of the privacy area contain the user with a probability proportional to their size. A weaker requirement is to ensure this for at least *some* regions. We define p-Partitionability Property, which states that *at least* p regions, which partition the whole privacy area, have such a property. Formally:

Definition 9 (p-Partitionability Property). *A privacy area A_i fulfills the p-Partitionability Property iff the partition of equally-spaced concentric annuli $\mathcal{P}(A_i) = \{\alpha_0, \ldots, \alpha_{p-1}\}$ (Fig. 10) divides A_i in such a way that:*

$$\forall j,\ P\left\{\mathbf{X} \in \alpha_j | A_i\right\} = \frac{\text{size}(\alpha_j)}{\text{size}(A_i)} \tag{22}$$

DVC-UniLO fulfills the p-Partitionability of A_i by leveraging on the Accuracy of A_{i-1}. With reference to Figure 10, suppose that A_1 contains $\mathbf{X}$ and the magnitude of $\mathbf{d}_{1,2}$ is equal to $5r_1$. Then, we can be sure that $\mathbf{X}$ is inside the annulus α_2. If we generate such a magnitude with probability 5/9, then $\mathbf{X}$ will be inside α_2 with the same probability, which is proportional to the size of α_2. We can repeat the same reasoning for the other annuli α_0 and α_1. The magnitude μ of the vector $\mathbf{d}_{1,2}$ becomes a discrete random variable having the *probability mass function* (*pmf*) shown in Figure 11. In this way, A_2 fulfills 3-Partitionability Property. Obviously this method is possible only if $2r_{i-1}$ divides exactly r_i.

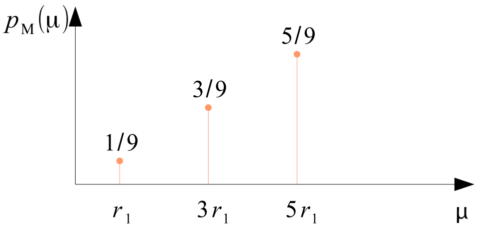


Figure 11: μ *pmf* of a discrete UniLO vector (example of Fig. 10)

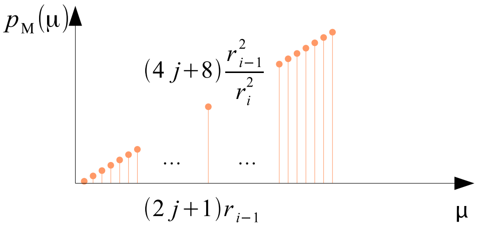


Figure 12: μ *pmf* of a discrete UniLO vector

By generalizing the formula we obtain the following *pmf* for the magnitude of $\mathbf{d}_{i-1,i}$:

$$p_M(\mu) = \begin{cases} (8j+4)\frac{r_{i-1}^2}{r_i^2} & \mu = (2j+1)r_{i-1} \\ 0 & \text{otherwise} \end{cases} \tag{23}$$

$$\text{where} \quad j = 0 \ldots p-1, \qquad p = \frac{r_i}{2r_{i-1}}$$

The *pmf* is depicted in Fig. 12. We call *discrete* UniLO *vector* a shift vector with such a magnitude and a uniform angle. We will use the following notation:

$$\mathbf{d}_{i-1,i} = \text{D-UniLO}(r_i, r_{i-1})$$

to say that $\mathbf{d}_{i-1,i}$ is a vector created by the discrete UniLO operator with privacy radius r_i and precedent privacy radius r_{i-1}.

If $2r_{i-1}$ does not divide r_i, DVC-UniLO will behave like VC-UniLO. The general formula for creating shift vectors is the following:

$$\mathbf{d}_i = \begin{cases} \text{UniLO}(r_1, r_0) & i = 1 \\ \mathbf{d}_{i-1} + \text{D-UniLO}(r_i, r_{i-1}) & i > 1,\ r_i = 2pr_{i-1} \\ \mathbf{d}_{i-1} + \text{UniLO}(r_i, r_{i-1}) & \text{otherwise} \end{cases}$$
$$\text{where} \quad p \in \mathbb{N}$$

It is trivial to show that DVC-UniLO fulfills Accuracy and Inclusion Properties for all the privacy areas, like VC-UniLO does. In addition, we state the following:

Theorem 6. *For each privacy area A_i having $i \geq 2$ and $r_i = 2pr_{i-1}$,* DVC-UniLO *fulfills p-Partitionability Property.*

PROOF. In the following, $\mu = \|\mathbf{d}_{i-1,i}\|$, and $\mu_j = (2j+1)r_{i-1}$. Let us compute the probability $P\{\mathbf{X} \in \alpha_j\}$.

$$
\begin{aligned}
& P\{\mathbf{X} \in \alpha_j\} = && (24)\\
= \quad & P\{\mu = \mu_j\} \cdot P\{\mathbf{X} \in \alpha_j | \mu = \mu_j\} + && (25)\\
+ \quad & P\{\mu \neq \mu_j\} \cdot P\{\mathbf{X} \in \alpha_j | \mu \neq \mu_j\} && (26)
\end{aligned}
$$

If A_{i-1} enjoys Accuracy Property and $\mu = \mu_j$, then the user will surely be in annulus α_j. Thus, $P\{\mathbf{X} \in \alpha_j | \mu = \mu_j\} = 1$. On the other hand, $P\{\mathbf{X} \in \alpha_j | \mu \neq \mu_j\} = 0$ for the same reason. Hence:

$$
\begin{aligned}
P\{\mathbf{X} \in \alpha_j\} &= P\{\mu = \mu_j\} && (27)\\
&= (8j+4)\frac{r_{i-1}^2}{r_i^2} && (28)\\
&= \frac{\text{size}(\alpha_j)}{\text{size}(A_i)} && (29)
\end{aligned}
$$

The proof is complete.

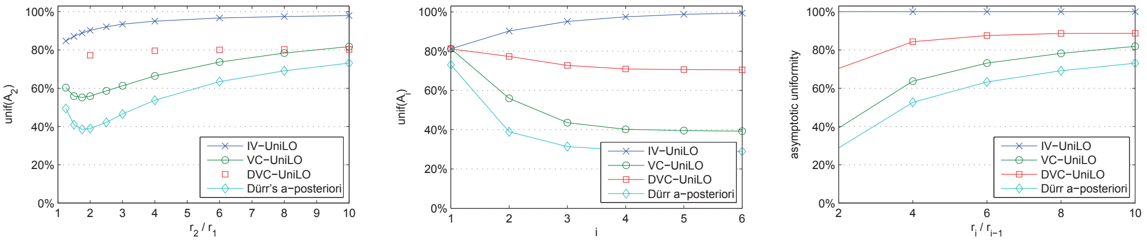
DVC-UNILO is an improvement of VC-UNILO. It fulfills the Inclusion Property and offers a better uniformity.

6.5. Uniformity analysis

We performed Monte Carlo simulations to compute the uniformity indexes of the privacy areas produced by IV-UNILO, VC-UNILO and DVC-UNILO under different conditions. We also compared them with "*a-posteriori* share generation algorithm" by Dürr et al. [17] which offers multiple levels of privacy with a perturbation approach. Dürr used a noise uniform in angle and uniform in magnitude to obfuscate user's positions. Actually, Dürr's algorithm dealt only with location measurements with infinite precision ($r_0 = 0$). To make meaningful comparisons, we adapted it to deal with finite-precision localization technologies. This is easily done by creating shift vectors with maximum magnitude equal to $r_1 - r_0$, as UNILO-based algorithms do. We simulated a localization technology with $r_0 = (1/10)r_1$. Tests showed that our algorithms outperform Dürr's ones in terms of uniformity.

Figure 13a shows the uniformity index of the second-level privacy area wrt r_2/r_1, with a precise localization technology. Note that IV-UNILO gets closer to the optimum than all the other methods. DVC-UNILO improves the performance of VC-UNILO when r_2/r_1 is not too large. Dürr's algorithm performs always worse than UNILO-based algorithms.

The performance of DVC-UNILO remains high at higher levels of privacy. Figure 13b shows the uniformity indexes of A_1–A_6 with $r_i = 2r_{i-1}$ $(i > 1)$. The tests revealed that all the four methods approach constant values at higher privacy levels: 28.8% for Dürr's algorithm, 39.2% for VC-UNILO, 70.4% for DVC-UNILO, and 100.0% for IV-UNILO. The asymptotic value of the uniformity index $\text{unif}(A_\infty)$ depends only on the algorithm employed and on the radius ratio r_i/r_{i-1}. Figure 13c shows $\text{unif}(A_\infty)$ wrt the radius ratio. We can easily see that UNILO-based algorithms outperforms Dürr's obfuscation algorithm.



(a) unif (A_2) with $r_1 = 10r_0$ (b) unif (A_i) with $r_1 = 10r_0$ and $r_i = 2r_{i-1}$ (c) unif (A_∞)

Figure 13: Uniformity index (500K Monte Carlo runs for each point)

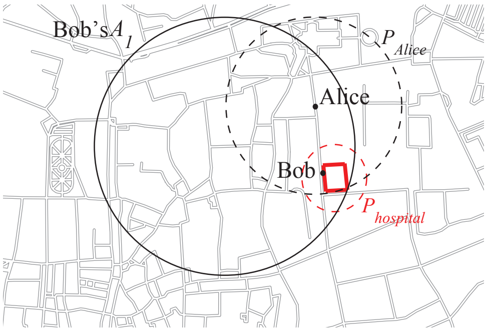


Figure 14: "Close friends" application

To sum up, in order to guarantee an optimal level of uniformity, the privacy radius set must be configured wisely. In particular, it is always better to set the first privacy radius far greater than the error radius of the sensor ($r_1 \gg r_0$). In addition, if we want to defend against collusion attacks, it is better to use DVC-UniLO and set each privacy radius to be the double or quadruple of the previous one. In this way, we have both a good granularity on the privacy radii, and a good uniformity index, which tends to 70%–84% (with collusion resistance) or 100.0% (without collusion resistance) with the growing of i.

7. Utility Analysis on an Example Application

We will describe now an example social application, called "close friends", in which users share their obfuscated positions with their friends. Alice wants to find out which of her friends are in her proximity. We define "being in the proximity of Alice" as "being at a distance of 400 meters or less from Alice". The service provider gathers the obfuscated positions of Alice's friends and sends them to Alice. While Alice knows her own position, the locations of her friends are obfuscated. Suppose Bob is one of Alice's friends. Since Alice does not know his exact location, the question "is Bob in my proximity?" will necessarily have a probabilistic answer.

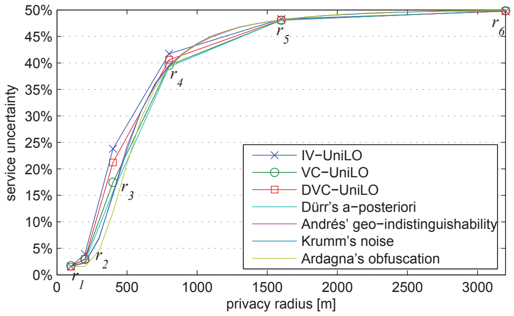


Figure 15: Uncertainty of "close friends" service (1000 Monte Carlo runs for each point)

The problem can be modeled as depicted in Fig. 14. Alice builds a circle centered on its position and with 400 meters of radius (*proximity area*, P_{Alice}), and computes the intersection between that area and the privacy area of Bob (A_1). If Bob is inside this intersection, he will be in Alice's proximity. The probability that such an event happens is:

$$P\left\{\text{Bob} \in P_{\text{Alice}}\right\} = \iint_{P_{Alice} \cap A_1} f_{X,Y}\left(x,y\right)\,\mathrm{d}x\mathrm{d}y \tag{30}$$

To make such a calculus, Alice should perform a statistical analysis of Bob's position and then compute numerically the integral. This operation is quite inefficient. However, if A_1 is assumed to be Uniform and Accurate, Eq. 30 will simplify in:

$$P\left\{\text{Bob} \in P_{\text{Alice}}\right\} \approx \frac{\text{size}(P_{\text{Alice}} \cap A_1)}{\text{size}(A_1)} \tag{31}$$

Alice performs this calculus only for each friend whose $\mathbf{X}_1$ is nearer than $r_1 + 400\,\text{m}$. The others have no intersection, and thus 0% probability.

We evaluated the utility of the presented obfuscation operators in our example "close friends" application. Our utility metric is the mean *uncertainty* in the service's answer. We define the uncertainty as the absolute difference between the computed proximity probability and the true answer, i.e. 1 if the friend is close, 0 otherwise. More formally, if P_{Alice} is the proximity area of Alice and A_i is the privacy area of Bob:

$$\text{uncert}(A_i) = \left|\frac{\text{size}(P_{\text{Alice}} \cap A_i)}{\text{size}(A_i)} - \text{prox}(\text{Bob})\right| \tag{32}$$

$$\text{prox}(\text{Bob}) = \begin{cases} 1 & \text{if Bob is in the proximity} \\ 0 & \text{otherwise} \end{cases} \tag{33}$$

Low values of uncertainty mean that the computed answers are close to the true answers. In the simulations, Bob's position is taken in Alice's proximity with 50% probability. Locations are measured with $r_0 = 10\,\text{m}$, and the privacy radii follow a geometric progression $\rho = \{100\,\text{m}, 200\,\text{m}, 400\,\text{m}, \ldots\}$. Figure 15 shows the mean uncertainty of Alice using the "close friends" service, versus the privacy preferences of her friends. We can see that the uncertainty depends mainly on the size of the privacy area, and only

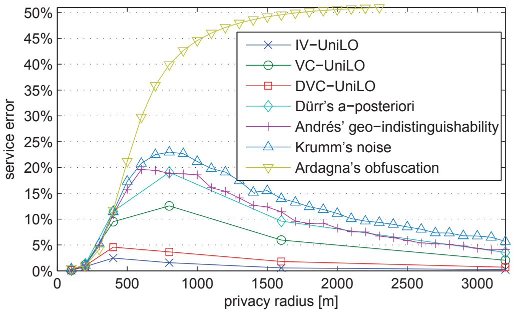


Figure 16: Error of “close friends” service (1000 Monte Carlo runs for each point)

marginally on the obfuscation operator. For $i > 5$, corresponding to a privacy radius $r_5 = 1.6\,\mathrm{Km}$, the obfuscated positions lose their utility in determining the proximity. This suggests that for this kind of service, $i \in [0, 5]$ is a suitable range of privacy preferences.

Together with the utility, it is interesting to measure the error that Alice makes in considering the privacy areas as uniform when they are not. Many privacy-aware services [14, 28] postulates the uniformity, rather than providing it. In practice, they use an approximate calculus (Equation 31) instead of an exact one (Equation 30). The impact of such an approximation can be quite high, if the obfuscation does not provide for Uniformity and Accuracy Properties. We evaluated this by measuring the mean *service error*, i.e. the mean absolute difference between the probability computed with and without the approximation. More formally:

$$\mathrm{error}(A_i) = \left| \frac{\mathrm{size}(P_{\mathrm{Alice}} \cap A_i)}{\mathrm{size}(A_i)} - P\left\{\mathrm{Bob} \in P_{\mathrm{Alice}}\right\} \right| \qquad (34)$$

Low values of service error mean that the computed proximity probabilities are close to the real ones. We compared UniLO algorithms with other noises, namely Dürr's “*a-posteriori* share generation algorithm” [17], Krumm's noise [16], and Ardagna's obfuscation operators [14]. Figure 16 shows the results of the simulations. We can see that the service's mean error depends mainly on the uniformity of the obfuscation noise. IV-UniLO and DVC-UniLO perform near to the optimum of 0% error, as they are highly uniform and they respect Accuracy Property. On the contrary, Ardagna's obfuscation performs particularly bad, because it respects neither Uniformity nor Accuracy.

8. Resistance against Inference Adversary

In previous sections we used the concept of agnostic adversary to measure the uniformity of an obfuscation operator. Now we introduce a realistic adversary, which owns auxiliary information (the map), and we show that a better uniformity improves the resistance against such a threat.

The *inference adversary* tries to infer sensitive information from the user's position and other auxiliary information. Let us suppose that there is a “sensitive point” on the map, in the sense that the proximity to that point can allow the adversary to infer

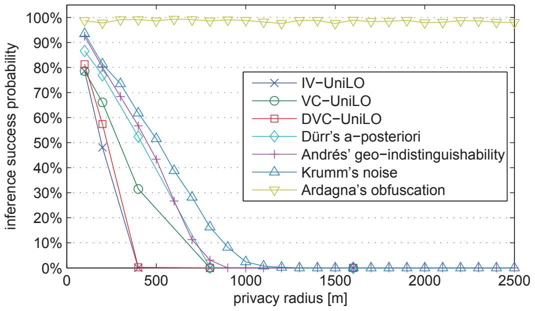


Figure 17: Success of inference attack (1000 Monte Carlo runs for each point)

sensitive information about the user. An example of that could be a hospital for the cancer treatment. If Bob sends his position from inside or from the close proximity of the hospital, the adversary could easily infer his health condition. Such an adversary could be the "close friends" service provider, or Alice herself. Let us suppose that Bob actually is in the hospital, and that the adversary knows his privacy area. The adversary performs a statistical analysis of Bob's position, knowing the localization technology and the obfuscation operator employed. Then she uses her auxiliary information by excluding those zones that cannot contain users (inside walls, rivers, etc.). The result of this analysis is a probability distribution over the map. Finally, the adversary computes the probability that Bob is in the hospital close proximity, say, inside a proximity area P_{hospital} of 200 meters of radius (cfr. Fig. 14). If such a probability is 50% or more, the adversary successfully infers the health condition of Bob.

We evaluated the success probability of the inference adversary on a real map of Pisa city center, extracted from public OpenStreetMap data [45]. Figure 17 shows the probability that the adversary has in guessing the health condition of Bob wrt his privacy radius. We can see that UniLO algorithms offer perfect protection even for small privacy radii (400 meters for IV-UniLO and DVC-UniLO). Ardagna et al.'s obfuscation offers no protection against inference attacks.

9. Conclusion

We proposed UniLO, a location obfuscation operator able to guarantee uniformity even in the presence of imprecise location measurements. UniLO does not require a centralized and trusted obfuscator. We dealt with service differentiation by proposing and comparing three UniLO-based obfuscation algorithms offering multiple contemporaneous levels of privacy. Finally, we experimentally proved that UniLO outperforms state-of-the-art perturbation algorithms both in terms of utility and resistance against inference attacks.

Acknowledgments

This work has been supported by EU FP7 Integrated Project PLANET (grant no. FP7-257649), and by Italian Research Project TENACE (pr. no. 20103P34XC_008).

References

[1] F. Gustafsson, F. Gunnarsson, Mobile positioning using wireless networks, IEEE Signal Processing Magazine 22 (4) (2005) 41–53.

[2] M. Anisetti, C. A. Ardagna, V. Bellandi, E. Damiani, S. Reale, Map-based location and tracking in multipath outdoor mobile networks, IEEE Transactions on Wireless Communications 10 (3) (2011) 814–824.

[3] G. Sun, J. Chen, W. Guo, K. R. Liu, Signal processing techniques in network-aided positioning, IEEE Signal Processing Magazine 22 (4) (2005) 12–23.

[4] L. Barkuus, A. Dey, Location-based services for mobile telephony: a study of users privacy concerns, in: Proceedings of INTERACT'03, IOS, 2003, pp. 709–712.

[5] T. D'Roza, G. Bilchev, An overview of location-based services, BT Technology Journal 21 (1) (2003) 20–27.

[6] F. Espinoza, P. Persson, A. Sandin, H. Nyström, E. Cacciatore, M. Bylund, GeoNotes: Social and navigational aspects of location-based information systems, Tech. Rep. T2001/08, Swedish Institute of Computer Science (SICS) (May 2001).

[7] S. Duri, M. Gruteser, X. Liu, P. Moskowitz, R. Perez, M. Singh, J.-M. Tang, Framework for security and privacy in automotive telematics, in: Proceedings of WMC'02, ACM, 2002, pp. 25–32.

[8] G. Myles, A. Friday, N. Davies, Preserving privacy in environments with location-based applications, IEEE Pervasive Computing 2 (1) (2003) 56–64.

[9] P. Samarati, L. Sweeney, Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression, Tech. rep., Computer Science Laboratory SRI International (1998).

[10] P. Samarati, Protecting respondents identities in microdata release, IEEE Transactions on Knowledge and Data Engineering 13 (6) (2001) 1010–1027.

[11] M. Gruteser, D. Grunwald, Anonymous usage of location-based services through spatial and temporal cloaking, in: Proceedings of MobiSys'03, ACM, 2003, pp. 31–42.

[12] P. Kalnis, G. Ghinita, K. Mouratidis, D. Papadias, Preventing location-based identity inference in anonymous spatial queries, IEEE Transactions on Knowledge and Data Engineering 19 (12) (2007) 1719–1733.

[13] B. Gedik, L. Liu, Protecting location privacy with personalized k-anonymity: Architecture and algorithms, IEEE Transactions on Mobile Computing 7 (1) (2008) 1–18.

[14] C. A. Ardagna, M. Cremonini, S. De Capitani di Vimercati, P. Samarati, An obfuscation-based approach for protecting location privacy, IEEE Transactions on Dependable and Secure Computing 8 (1) (2011) 13–27.

[15] S. Mascetti, C. Bettini, D. Freni, X. S. Wang, S. Jajodia, Privacy-aware proximity based services, in: Proceedings of MDM'09, IEEE, 2009, pp. 31–40.

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

[17] F. Dürr, P. Skvortsov, K. Rothermel, Position sharing for location privacy in non-trusted systems, in: Proceedings of PerCom'11, IEEE, 2011, pp. 189–196.

[18] G. Dini, P. Perazzo, Uniform obfuscation for location privacy, in: Proceedings of DBSec'12, Springer, 2012, pp. 90–105.

[19] Matlab simulation scripts for UniLO. [link].
URL `www.iet.unipi.it/g.dini/download/code/UNILO-simulations.zip`

[20] M. F. Mokbel, C.-Y. Chow, W. G. Aref, The new Casper: query processing for location services without compromising privacy, in: Proceedings of VLDB'06, ACM, 2006, pp. 763–774.

[21] T. Wang, L. Liu, Privacy-aware mobile services over road networks, in: Proceedings of VLDB'09, VLDB Endowment, 2009, pp. 1042–1053.

[22] G. Ghinita, K. Zhao, D. Papadias, P. Kalnis, A reciprocal framework for spatial k-anonymity, Information Systems 35 (3) (2010) 299–314.

[23] T. Xu, Y. Cai, Exploring historical location data for anonymity preservation in location-based services, in: Proceedings of INFOCOM'08, IEEE, 2008, pp. 547–555.

[24] O. Abul, F. Bonchi, M. Nanni, Never walk alone: Uncertainty for anonymity in moving objects databases, in: Proceedings of ICDE'08, IEEE, 2008, pp. 376–385.

[25] A. R. Beresford, F. Stajano, Location privacy in pervasive computing, IEEE Pervasive Computing 2 (1) (2003) 46–55.

[26] J. Freudiger, R. Shokri, J.-P. Hubaux, On the optimal placement of mix zones, in: Proceedings of PETS'09, Springer, 2009, pp. 216–234.

[27] B. Palanisamy, L. Liu, MobiMix: Protecting location privacy with mix-zones over road networks, in: Proceedings of ICDE'11, IEEE, 2011, pp. 494–505.

[28] R. Cheng, Y. Zhang, E. Bertino, S. Prabhakar, Preserving user location privacy in mobile data management infrastructures, in: Proceedings of PETS'06, Springer, 2006, pp. 393–412.

[29] M. L. Yiu, C. S. Jensen, X. Huang, H. Lu, SpaceTwist: Managing the trade-offs among location privacy, query performance, and query accuracy in mobile services, in: Proceedings of ICDE'08, IEEE, 2008, pp. 366–375.

[30] M. Duckham, L. Kulik, A formal model of obfuscation and negotiation for location privacy, in: Proceedings of the Pervasive'05, Springer, 2005, pp. 152–170.

[31] B. Schneier, Secrecy, security, and obscurity (May 2002).
URL `www.schneier.com/crypto-gram-0205.html`

[32] C. Dwork, Differential privacy, in: Proceedings of ICALP'06, Springer, 2006, pp. 1–12.

[33] M. E. Andrés, N. E. Bordenabe, K. Chatzikokolakis, C. Palamidessi, Geo-indistinguishability: Differential privacy for location-based systems, in: Proceedings of SIGSAC'13, ACM, 2013, pp. 901–914.

[34] G. Ghinita, P. Kalnis, A. Khoshgozaran, C. Shahabi, K.-L. Tan, Private queries in location based services: anonymizers are not necessary, in: Proceedings of the 2008 ACM SIGMOD international conference on Management of data, ACM, 2008, pp. 121–132.

[35] R. Paulet, M. G. Koasar, X. Yi, E. Bertino, Privacy-preserving and content-protecting location based queries, in: Data Engineering (ICDE), 2012 IEEE 28th International Conference on, IEEE, 2012, pp. 44–53.

[36] H. Hu, J. Xu, Q. Chen, Z. Yang, Authenticating location-based services without compromising location privacy, in: Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data, ACM, 2012, pp. 301–312.

[37] S. Papadopoulos, S. Bakiras, D. Papadias, Nearest neighbor search with strong location privacy, Proceedings of the VLDB Endowment 3 (1-2) (2010) 619–629.

[38] A. Khoshgozaran, C. Shahabi, H. Shirani-Mehr, Location privacy: going beyond K-anonymity, cloaking and anonymizers, Knowledge and Information Systems 26 (3) (2011) 435–465.

[39] P. A. Zandbergen, Accuracy of iPhone locations: A comparison of assisted GPS, WiFi and cellular positioning, Transactions in GIS 13(s1) (2009) 5–26.

[40] A. Pal, Localization algorithms in wireless sensor networks: Current approaches and future challenges, Network Protocols and Algorithms 2 (1) (2010) 45–74.

[41] B. Hofmann-Wellenhof, H. Lichtenegger, J. Collins, Global Positioning System: Theory and Practice, Springer, 2001.

[42] C. A. Ardagna, M. Cremonini, G. Gianini, Landscape-aware location-privacy protection in location-based services, Journal of Systems Architecture 55 (4) (2009) 243–254.

[43] P. Skvortsov, F. Dürr, K. Rothermel, Map-aware position sharing for location privacy in non-trusted systems, in: Proceedings of Pervasive'12, Springer, 2012, pp. 388–405.

[44] F. Giurlanda, P. Perazzo, G. Dini, HUMsim: A privacy-oriented human mobility simulator, in: Proceedings of S-CUBE'14 (to appear), Springer, 2014.

[45] OpenStreetMap. [link].
URL `www.openstreetmap.org`