Application of a predictive method to protect privacy of mobility data

Emilio Molina, Mirko Fiacchini, Arthur Goarant, Rémy Raes, Sophie Cerf, Bogdan Robu

Control Engineering Practice 156 (2025) 106223

Contents lists available at ScienceDirect

Control Engineering Practice

journal homepage: www.elsevier.com/locate/conengprac

Application of a predictive method to protect privacy of mobility data

Emilio Molina [a,*], Mirko Fiacchini [b], Arthur Goarant [c], Rémy Raes [c], Sophie Cerf [c], Bogdan Robu [b]

[a] Univ Lyon, INSA Lyon, Inserm, UCBL, CNRS, CREATIS, UMR5220, U1294, Villeurbanne, 69100, France
[b] Univ. Grenoble Alpes, CNRS, Grenoble INP, GIPSA-lab, Grenoble, 38000, France
[c] Univ. Lille, Inria, CNRS, Centrale Lille, UMR 9189 CRIStAL, Lille, F-59000, France

ARTICLE INFO

Keywords:
Optimal privacy
MPC
Location privacy

ABSTRACT

Users of geo-localized applications on mobile devices need protection to avoid threats to their privacy. Such protection should vary in time, to cope with the dynamical nature of mobility data. We present a method to protect the privacy of users of location-based services, based on Model Predictive Control techniques. We employ three different predictors for future movements: an exact predictor, which serves as the baseline for the best expected performance, and two additional predictors allowing for online implementation. One of these predictors assumes the user is moving in a way that minimizes privacy, while the other is a linear predictor. The method has been applied to two datasets, Privamov and Cabspotting, which contain mobility data collected from real users when using a mobile device. The method demonstrated an improvement in privacy compared to a state-of-the-art mechanism by approximately 12% increase for Privamov users and 5% for Cabspotting users, while maintaining the same level of utility.

1. Introduction

The widespread use of smartphones and similar devices has generated a big amount of data relative to the users. This work is concerned with mobility information data, that are shared to third parties when using location-based services, such as navigation applications, venue finders or sport tracking services. This information may reveal some of the user's Points Of Interest (POI), for instance, their home or place of work (Hariharan & Toyama, 2004). Knowing the user's POIs, it is possible to infer privacy-sensitive information such as their identity, social relationships, and even religious, political or sexual orientations (Gambs et al., 2010).

To prevent malicious usages of this information, location privacy protection mechanisms (LPPMs for short) have been proposed (Primault et al., 2019). This work focuses on online use cases, i.e., when a user repeatedly sends their location, and receives a continuous-like service. In those cases, LPPMs work at every transmission time, and use current and past locations. Most continuous LPPM are based on obfuscation (Jiang et al., 2021), that is, they modify the real location data before sending it to the service. Geo-Indistinguishability (Geo-I) (Andrés et al., 2013) is the reference state-of-the-art obfuscation-based LPPM. Inspired by the theoretical concept of differential privacy (Dwork, 2006), Geo-I consists in applying spatial noise to the location data before transmitting it. It has been extended to implement spatial adaptation (Chatzikokolakis et al., 2015; Koufogiannis & Pappas, 2016), temporal adaptation (Cerf et al., 2023), semantic adaptation (Min et al., 2024), and elasticity to protect isolated locations (Biswas & Palamidessi, 2024).

This work is an extension of the method presented in Molina et al. (2023a). It is based on optimization and model predictive control (MPC) techniques. The principle is to transmit an obfuscated location that maximizes the user's privacy, while maintaining an acceptable level of service usability, based on predictive information about the future mobility of the user. In previous work (Molina et al., 2023a), the hypothesis is made that one has access to the future position of the user. While it allows to motivate the benefit of prediction, it is an unrealistic assumption in practice. Further works (Molina et al., 2023b) relax this assumption, considering worst-case future mobility, at the cost of losing performance in terms of reachable privacy levels. Furthermore, the optimal mechanisms presented in Molina et al. (2023a, 2023b) have only been validated as a proof-of-concept, i.e., on the data of a single user.

In this work, we introduce two novelties. First, we address the limitation of previous work, which was to test the method under the unrealistic assumption that the future position is known. To overcome

* Corresponding author.
E-mail addresses: emilio.molina@creatis.insa-lyon.fr (E. Molina), mirko.fiacchini@gipsa-lab.fr (M. Fiacchini), arthur.goarant@inria.fr (A. Goarant), remy.raes@inria.fr (R. Raes), sophie.cerf@inria.fr (S. Cerf), bogdan.robu@gipsa-lab.fr (B. Robu).

https://doi.org/10.1016/j.conengprac.2024.106223
Received 26 March 2024; Received in revised form 23 October 2024; Accepted 18 December 2024
Available online 28 December 2024

this, we have designed a future mobility predictor. The second proposed contribution is the comprehensive evaluation of the protection mechanism performance over two real-life mobility datasets, Privamov (Mokhtar et al., 2017) and Cab-spotting (Piorkowski et al., 2022), gathering more than 630 users and more than half a million hours of data transmissions.

The evaluation considers three methods that use different future predictions: *(i)* using the real future position of a user *(ii)* assuming a pessimistic case, i.e., considering the future locations that minimize user's privacy *(iii)* estimating the future locations using a linear prediction-based method, developed for that purpose The presented approach is applied on two real datasets, to evaluate its validity and performance in ensuring a relevant privacy improvement compared to the unprotected case and to Geo-I. While the prediction scenario with exact knowledge serves as a baseline, the pessimistic scenario shows to be adequate when considering long horizons in the MPC setup, while the linear predictor is suited for short horizons. Additionally, the average execution time of each instance presented in this work is evaluated to ensure their online applicability.

Other optimal formulations of protection mechanisms have been proposed (Bordenabe et al., 2014; Oya et al., 2017; Shokri et al., 2012), some in a dynamical setup (Xiao & Xiong, 2015; Yu et al., 2017) adaptive to influence factors (Niu et al., 2022), or in feedback (also referred to knowing-and-learning) (Ma et al., 2023). Such approaches often result, however, in massive data distortion, leading to a useless location-based service (Krumm, 2007). In this work, we advocate using the property of predictability of the human mobility to improve privacy protection without reducing too much the service utility. While Chatzikokolakis et al. (2014) show it is a promising approach, their work implements a decision that is binary (transmit the predicted position or use Geo-I), and is repeated each time step. Conversely, our approach provides a fine-tuned protection that is based on an optimal mechanism, and uses mobility prediction on a horizon of future steps, allowing for a better global optimality of the results. Overall, our optimal predictive method uses a model that explicitly takes into account the dynamic variation of mobility data. It allows optimizing on a future horizon of several points, handling mobility behavior that has an inertia of more than one sampling period. Let us take an example: if the user has high speed (e.g. in a train) and start to slow down, it can be predicted that the user will stop in a future horizon (but not necessarily in the next timestep). In such case, the optimal mechanism allows anticipation and thus improves the performance of the protection mechanism. It is, to our knowledge, the first approach in the literature that exploits this dynamic aspect, allowing to address the challenges of practicality and personalization in location privacy (Jiang et al., 2021). Note that we do not aim at protecting a user against an attack consisting of predicting the user's next locations, as in, e.g., Qiu et al. (2023), Zhan et al. (2023), but rather use the mobility prediction to improve the obfuscation.

The structure of the paper is the following. Section 2 gives a background on mobility data and protection mechanisms, formally defines the optimal privacy problem, and presents the MPC-based control solution. Section 3 describes the three alternative predictors used to obtain the future position of a user. Section 4 gives the experimental setup, and Section 5 presents the results of applying the MPC method in both Privamov and Cabspotting datasets. Finally, Section 6 concludes the paper.

2. Background

The problem under study consists of transmitting obfuscated positions to safeguard the privacy of a user of a location-based service while preserving the usefulness of the transmitted positions.[1] In this context, the term "privacy" is defined as the difficulty with which a user's POIs can be detected. Conversely, the term "utility" is defined based on the hypothesis that if an application knows the user's actual location, its performance is optimal. Consequently, the greater the discrepancy between the actual and transmitted positions, the poorer the performance. The following subsection will define both terms mathematically. We note here that the notation $(\cdot,\cdot)$ denotes the concatenation of two (or more) vectors, e.g. $(z,w)=[z^\top\ w^\top]^\top$ for vectors z and w.

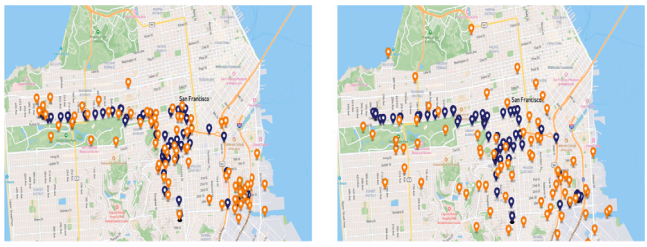


Fig. 1. Application of Geo-I to the mobility trace of a taxi user in San Francisco (Cabspotting dataset Piorkowski et al., 2022). Actual position (in blue) v.s. obfuscated ones with Geo-I (orange) for two different values of ε. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

To be precise, obfuscation in this context means to modify the real position by means of a protection mechanism. Consequently, the obfuscated position is transmitted to the mobile application instead of the actual one. In the state-of-the-art on this subject, we can identify Geo-I (Andrés et al., 2013) as one of the most widely used. The Geo-I obfuscation mechanism adds a spatial noise to the transmitted user's position in order to protect the real position. Given the actual position $l=(x,y)\in\mathbb{R}^2$, the transmitted position $\bar{l}=(\bar{x},\bar{y})\in\mathbb{R}^2$ for a location-based service after applying Geo-I is obtained as follows:

$$\bar{l}=l-\frac{W_{-1}(\frac{\alpha-1}{e})+1}{\varepsilon}\begin{pmatrix}\cos\ \theta\\ \sin\ \theta\end{pmatrix} \tag{1}$$

where W_{-1} is the -1 branch of the Lambert W function, e is Euler's number, α and θ are drawn uniformly in $[0,1)$ and $[0,2\pi)$ respectively. The parameter $\varepsilon>0$ allows one to manage the intensity of the disturbance injected by the Geo-I mechanism.

Fig. 1 shows the mobility data of a user (in blue) and two instances of the Geo-I mechanism (in orange). The image on the left uses a larger ε value than the one used in the image on the right, producing a mobility trace closer to the actual one: its utility is greater. As shown in Fig. 1, in this kind of mechanisms, it is possible to generate an obfuscated position close to the actual one, maintaining a good level of utility but not a good level of privacy (stopping points, i.e. POIs, are fairly identifiable in the image on the left). On the contrary, it is possible to generate a highly obfuscated trace preserving good levels of privacy, but affecting the utility of the application because the transmitted positions are far from the actual one, as shown in the image on the right. Playing with the ε parameter allows leveraging both the privacy and utility of the transmitted data. This trade-off between utility and privacy in Geo-I was the focus of the method proposed in Cerf et al. (2023). Going further, an optimal predictive formulation for computing the obfuscated position is presented in Molina et al. (2023a). The present paper is built on this formulation and control method.

To assess an individual's privacy level, the history of their movements is typically utilized. For instance, a location is designated as a Point of Interest (POI) if an individual spends at least 15 min at the same location (see for example Cerf et al., 2023; Primault et al., 2019). During this 15-minute period, the transmission times are not uniformly distributed, and thus the number of transmitted points varies depending on the selected time window. This variability can affect the privacy measures and the implementation of protection mechanisms, as the

[1] In this paper, we consider the representation of positions as a bidimensional vector in the plane.

transmission instants are not regular. It is particularly challenging for MPC techniques, which are typically employed to manage dynamical systems that have been uniformly discretized in time. Furthermore, to predict future positions, fixing the time at which the position will be predicted makes the prediction easier to manage. Therefore, for the sake of simplicity, a uniform resampling is used. This entails the utilization of synthetic temporal points, at which a user may or may not transmit a position, with the duration between consecutive points remaining constant.

The next section introduces the necessary tools and recalls the main step in the formulation of the optimal predictive approach proposed in Molina et al. (2023a); before, novel prediction techniques that overcome the limitations of those presented in Molina et al. (2023a, 2023b), are presented in Section 3.

2.1. Optimal predictive obfuscation

Consider a time interval $[0,\tau]$ uniformly discretized in M points $\{t_k\}_{k=1}^{N}$, and $\Delta t := t_{k+1} - t_k$ the time between two consecutive instants. We denote by $l(k) = (x(k), y(k)) \in \mathbb{R}^2$ the actual user's position and by $\bar{l}(k) = (\bar{x}(k), \bar{y}(k)) \in \mathbb{R}^2$ the transmitted one at time t_k. At that time, the privacy is a function of the positions transmitted during the last τ units of time, specifically within the time window $[t_k - \tau, t_k]$. We denote by $N(k)$, with $0 \le N(k) \le M$, the total transmitted positions in that period and $\{\bar{l}(k-N(k)+1), \dots, \bar{l}(k)\}$ the respective transmitted positions. The **privacy** measure corresponds to the following function p proposed in Cerf et al. (2023) and refined in Molina et al. (2023a):

$$p(k) = \frac{1}{N(k)} \sum_{i=1}^{N(k)} \|\bar{l}(k - N(k) + i) - c(k)\|_2 \tag{2}$$

where $c(k)$ is the centroid of $\{\bar{l}(k-N+1), \dots, \bar{l}(k)\}$ defined by:

$$c(k) = \frac{1}{N(k)} \sum_{i=1}^{N(k)} \bar{l}(k - N(k) + i).$$

This function takes a low value (close to 0) when the user does not move significantly during N time steps. Conversely, for a highly mobile user, it takes a high value. Thus, the user's POIs, i.e., the points where the user spends significant time, will exhibit a low privacy value and will then highlight the most important moments to protect.

We measure the **utility** loss of a transmitted position in terms of its distance from the actual position. This follows the idea that the positions transmitted to a location-based service that are closer to the real one should produce more accurate information and/or recommendations than those that are farther away, and therefore they have a lower loss of utility. The utility loss function is expressed as:

$$q(k) = \|l(k) - \bar{l}(k)\|_2. \tag{3}$$

At time t_k, the problem consists in obfuscating a position, maximizing the privacy and minimizing the utility loss. This bi-objective **optimization problem** can be reformulated as:

$$\begin{aligned} \bar{l}^*(k) = \arg\max_{\bar{l}(k)\in\mathbb{R}^2} \quad & p(k) \\ \text{s.t.} \quad & \|l(k) - \bar{l}(k)\|_2^2 \le \bar{q}(k)^2, \end{aligned} \tag{4}$$

where $\bar{q}(k)$ is a parameter representing an upper bound on the utility loss. Alternatively, a second formulation is also possible:

$$\begin{aligned} \bar{l}^*(k) = \arg\min_{\bar{l}(k)\in\mathbb{R}^2} \quad & \|l(k) - \bar{l}(k)\|_2^2 \\ \text{s.t.} \quad & p(k) \ge \underline{p}(k), \end{aligned} \tag{5}$$

$\underline{p}(k)$ being a parameter corresponding to a lower bound of the privacy. Both reformulations are non-convex optimization problems representing distinct approaches to address the bi-objective problem. In these optimization problems, the bounds $\bar{q}(k)$ and $\underline{p}(k)$ are parameters that must be specified by the user based on the performance requirements.

2.2. Model predictive control approach

In this section we recall an approach, based on MPC theory, to determine online the fictitious noise to be added to the current position before its transmission to conceal the conflicting aims of privacy preservation and utility loss reduction. In Molina et al. (2023a), the problem (4) was exploited to propose a method that maximizes the current privacy using not only current and past positions, but also a prediction in the future steps of a user. This MPC-based method uses a transition system that stores the information about the previous N time steps at each time t_k. In addition, since a transmission might or might not have occurred at any time, a binary variable $n(k)$ is introduced into the transition system. This variable is defined as:

$$n(k) = \begin{cases} 1 & \text{if the position was transmitted at time } t_k \\ 0 & \text{otherwise.} \end{cases}$$

Thus, the state of the system is $\mathbf{z}(k) = (\mathbf{x}(k), \mathbf{y}(k), \mathbf{n}(k)) \in \mathbb{R}^N \times \mathbb{R}^N \times \{0,1\}^N$. Note that we use bold notation to refer to vectors. Vectors $\mathbf{x}$ and $\mathbf{y}$ act as buffers, storing the last N location states, and $\mathbf{n}$ the last N transmission occurrences. These buffers, in fact, contain the required past information that defines the privacy value and then can be considered as the state of a system that updates at every instant.

The transition system for the state $\mathbf{z}$ is then defined as:

$$\mathbf{z}(k+1) = \mathcal{A} \cdot \mathbf{z}(k) + \mathcal{B} \cdot u(k), \tag{6}$$

where

$$\mathcal{A} = \begin{pmatrix} A & 0 & 0 \\ 0 & A & 0 \\ 0 & 0 & A \end{pmatrix}, \qquad \mathcal{B} = \begin{pmatrix} b & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & b \end{pmatrix},$$

with $u(k) = (x(k+1), y(k+1), n(k+1))$ and

$$A = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix} \in \mathbb{R}^{N\times N}, \qquad b = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix} \in \mathbb{R}^N,$$

and, then, A and B are the matrices implementing a FIFO (first in first out) access strategy to any of the three buffers. Note that as solution of this system we obtain:

$$\begin{aligned} \mathbf{x}_i(k) &= x(k+i-N), \\ \mathbf{y}_i(k) &= y(k+i-N), \\ \mathbf{n}_i(k) &= n(k+i-N). \end{aligned}$$

where i is to the ith coordinate of the vectors $\mathbf{x}, \mathbf{y}$ and $\mathbf{n}$.

Since we are considering that, in the last N time-steps, the position could be transmitted or not, we have to adapt the definition of privacy to only consider the transmitted positions in the measure. The privacy function written in terms of the state of the transition system is:

$$\mathbf{p}(\mathbf{z}(k)) = \frac{\sum_{i=1}^{N} ((\mathbf{x}_i(k) - x_c(z(k)))^2 + (\mathbf{y}_i(k) - y_c(z(k)))^2) \cdot \mathbf{n}_i(k)}{\sum_{i=1}^{N} \mathbf{n}_i(k)}, \tag{7}$$

with $(x_c(k), y_c(k))$ the centroid calculated using

$$x_c(z(k)) = \frac{\sum_{i=1}^{N} \mathbf{x}_i(k) \cdot \mathbf{n}_i(k)}{\sum_{i=1}^{N} \mathbf{n}_i(k)}, \qquad y_c(z(k)) = \frac{\sum_{i=1}^{N} \mathbf{y}_i(k) \cdot \mathbf{n}_i(k)}{\sum_{i=1}^{N} \mathbf{n}_i(k)}. \tag{8}$$

The MPC method consists in solving, at each instant t_k where a position is transmitted (that is, when $n(k) = 1$), the following non-convex optimization problem which uses the prediction of the position in the H future time-steps:

$$\max_{(\delta x_i,\delta y_i)_{i=1}^H \in \mathbb{R}^H \times \mathbb{R}^H} \quad \sum_{j=1}^{H} \mathbf{p}(\bar{z}(k+j))$$

$$\text{s.t. } \bar{z}(k+i) = \mathcal{A}\bar{z}(k+i-1) + \mathcal{B}\bar{u}(k+i-1), \quad i \in \{1,\dots,H\},$$

$$\bar{u}(k+i-1) = \begin{pmatrix} x(k+i)+\delta x_i \\ y(k+i)+\delta y_i \\ n(k+i) \end{pmatrix}, \quad i \in \{1,\dots,H\},$$

$$\delta x_i^2 + \delta y_i^2 \le \bar{q}^2(k+i-1), \quad i \in \{1,\dots,H\},$$

$$\bar{z}(k) = \bar{z}_H(k).$$

It consists in minimizing the average of the future privacy, respecting a maximal utility loss $\bar{q}(k)$ at each time-step. The controls $(\delta x_i, \delta y_i)_{i=1}^H$ are the obfuscation signal to add to the real position (x, y) and therefore the second to last constraint imposes a bound $\bar{q}$ on the norm of this fictitious noise, defined as utility loss in (3), to be added to the real position before transmission. The variables $\bar{z}$ and $\bar{u}$ are introduced as auxiliaries of the optimization problem, related to z and u in (6) and finally $\bar{z}_H(k)$ stores the transmitted position until time-step $k-1$. The last constraint, indeed, is necessary to impose that the predicted trajectory $\bar{z}$ starts with the current state, namely the more recent transmission data.

For more details, see Molina et al. (2023a). In the following, this method will be referred to as $MPC-H$.

3. Predictions of future positions

A key point of the method, which was not addressed in Molina et al. (2023a), is the use of predicted positions: the better the prediction, the better the algorithm behavior. For this purpose, we introduce in this paper a linear predictor of future movements. We compare it to two other predictors: future knowledge (Molina et al., 2023a) and worst-case approach (Molina et al., 2023b). These three predictors are described in the following subsections.

3.1. Exact future prediction

The first predictor corresponds to an oracle that provides the real future position. While not realistic in practice, this is an important case for comparison as it has no uncertainties regarding the future, and making it the best scenario in terms of prediction. Despite the fact that this case is not admissible in an *online* implementation since the future positions are not known in advance, the results obtained using this prediction in an *offline* implementation should represent an ideal approach and provide an upper bound on the possible privacy gain achievable by using the MPC method.

3.2. Pessimistic prediction

The second approach provides a prediction of the position that minimizes the privacy function, using the information from the last $N-1$ time steps. That position is solution of the following optimization problem:

$$\min_{\bar{l}_N \in \mathbb{R}^2} \frac{1}{N} \sum_{k=1}^{N} \left\| \bar{l}_k - \frac{\sum_{j=1}^{N} \bar{l}_j}{N} \right\|^2$$

which solution can be expressed in terms of the MPC variables, as demonstrated in Proposition 1 in Molina et al. (2023b), by the following formula:

$$\bar{l}_{wc}(z(k)) = \frac{\sum_{i=1}^{N-1} (\mathbf{x}_i(k), \mathbf{y}_i(k)) \cdot \mathbf{n}_i(k)}{\sum_{i=1}^{N-1} \mathbf{n}_i(k)}. \tag{9}$$

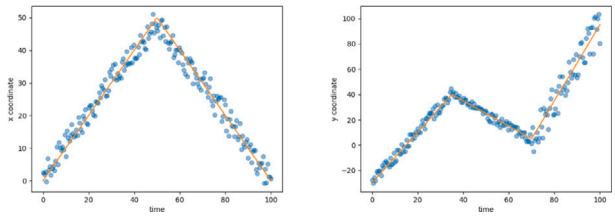


Fig. 2. Academic example demonstrating the application of piece-wise linear approximation to mobility data.

The pessimistic predictor assumes that the user will move, in all H future instants, towards the positions that minimize the privacy function. Eq. (9) is then iteratively applied at each step. This prediction should be beneficial in cases where the user's privacy is significantly low, although in the other cases, this approach may be very conservative because this worst-case prediction may often be inaccurate. The use of (9) allows an online implementation of this scenario, with a reasonable computation time for calculating the prediction.

3.3. Linear prediction

The third predictor is a linear method to estimate the future positions. In the literature, it is possible to find several works dedicated to the characterization of human mobility, mainly using machine learning techniques. One of the goals of this characterization is to estimate the future movement of an individual. A comprehensive review of existing methods is available in Toch et al. (2019).

In our mobile context, it is necessary to have a lightweight and fast predictor that can be implemented in a smartphone or similar device. For this reason, we decided to use a linear method instead of a more powerful method based on, e.g., neural networks. The Fast Linear Interpolation (FLI) algorithm presented in Raes et al. (2024) is therefore used for this purpose.

This method is a piecewise linear approximation technique that was originally developed to increase the storage capacity of mobility data on mobile devices. Although its original purpose was not intended for predicting the future spatial evolution of an individual, we can take advantage of its linear structure to infer future positions.

Given a set of positions, the original algorithm generates, for each coordinate, a sequence of linear affine functions $f_x^i(t) = S_x \cdot (t - t_i) + x_i$ and $f_y^j(t) = S_y \cdot (t - t_j) + y_j$, where (f_x^i, f_y^j) are the approximated positions at time t, S_x and S_y the slope of the current x and y segments respectively, x_i and y_j the horizontal and vertical position at times t_i and t_j respectively.

Fig. 2 shows an example of such linear interpolation, where the position values are represented in blue while in orange are depicted the linear functions obtained for each coordinate. There are two functions for the x coordinate and three for the y coordinate.

The number of linear functions in each sequence is determined by the algorithm. This number depends on a parameter noted γ, which determines when it is necessary to switch to another function. For lower values of γ the algorithm may lead to an overfitting model, while higher values may lead to an underfitting model. In our numerical experiments, we set $\gamma = 0.05$.

Note that this predictor assumes a user moving in a piece-wise straight line for short periods of time. The linear prediction assumes that the user will continue moving in the same direction (e.g., we do not predict any turns), with a speed that is dependent on its current movement.

Table 1
Statistics *per user* of Privamov database.

Indicators	Elapsed time (h)	Number of points		Period (s)		Period (s) using 95%	
		Raw	Resampled	Raw	Resampled	Raw	Resampled
mean	2285.3	81 330	28 090	303.4	870.6	10	30.4
std	2578.8	174 185	60 295	526.5	1491.8	0.2	2.3
min	0.853	126	48	10.3	30.4	10	30
25%	299.1	5111	1791	47.7	136.7	10	30
50%	1221.5	19 639	6843	129.4	375.0	10	30
75%	3409.5	81 105	27 615	303.2	865.7	10	30
max	9818.5	1 123 719	388 575	3255.2	9060.2	11.8	51.8

Table 2
Statistics per user of Cabspotting database.

Indicators	Elapsed time (h)	Number of points		Period (s)		Period (s) using 95%	
		Raw	Resampled	Raw	Resampled	Raw	Resampled
mean	530	20 932	19 835	125.8	133.5	54.9	56.6
std	101.2	6204	5789	614.6	1672.5	2.9	3.8
min	3.6	59	54	40.7	52.4	33	30
25%	556.4	18 721	17 836	79.6	84	53.7	55.2
50%	560	22 813	21 591	87.5	92	54.5	57.1
75%	561	25 037	23 748	101.1	106.5	55.6	57.5
max	575.4	49 367	38 339	14 380.9	15 627	92.3	94

Online linear prediction. The linear interpolation algorithm presented so far is an offline modeling method, requiring the complete dataset to give the linear models. Our setup differs in two aspects: (i) the prediction is online, and only requires past data, (ii) the model is used for prediction, that is future positions are drawn from the current linear function. In detail, to estimate future positions, we apply the linear prediction method to $\mathbf{x}(k)$ and $\mathbf{y}(k)$, that is the data from the past time window, and use the last linear function in the sequence to do the prediction. Although this prediction approach is relatively simple, it can be remarkably effective for short prediction horizons. Moreover, this method has already been implemented in iOS and Android, although in the offline setting, showing that the implementation of this scenario on a real mobile device is possible. It is very important to note here that the goal is not to have the most accurate prediction, but to show that the proposed algorithm performs better even with a slightly inaccurate future prediction.

4. Evaluation setup

This section describes and characterizes the datasets utilized in the numerical simulation, along with the metrics employed to assess the performance of the methods. The datasets employed are presented first, together with the compared obfuscation methods, which results are detailed in the subsequent sections. To conclude this section, we evaluate the performance of the two predictors.

Datasets. To evaluate the performance of the method, we use Privamov (Mokhtar et al., 2017) and Cabspotting (Piorkowski et al., 2022), that are two large datasets containing mobility traces of real users. A mobility trace refers to a time series of one or more spatial points (e.g., latitude and longitude tuples in our case) along with an identifier (e.g., actual position transmitted or not in our case) and/or application-specific information (Bhati & Eckhoff, 2019). Privamov contains the mobility traces of 96 users living, working or studying around Lyon city, France. The data were collected using a crowd-sensing application installed on smartphones that the volunteers used as their primary phone. The application collected data each time the system was used (e.g., change of location, new Wi-Fi scan, etc.). Cabspotting comprises the mobility traces of 536 taxis collected over 30 days in the San Francisco Bay Area. Each taxi was equipped with a GPS receiver that sent location updates (timestamp, identifier, geo-coordinates) to a central server. In the next two subsections and in Tables 1 and 2, we provide more details on both datasets. In particular, we present statistical information that demonstrates the variability between the two datasets. This will help us to conclude on the robustness of the method.

Sampling. Since the time between two consecutive transmission points is not constant, i.e., the discretization of $[0, \tau]$ is not uniform, we resampled the transmission traces using a uniform time discretization with a time step of 30 s, i.e. $\Delta t = 30$. As it is shown in the columns of Tables 1 and 2 relative to raw data, the two datasets have different transmission periods. For Privamov this value is $10s$ and for Cabspotting it is around $55s$. Therefore, for the sake of consistent application of the method, we have chosen $30s$, since it is an intermediate value for both sets of data. The resampling was computed as follows: for a time t_k in the uniform discretization, we assign to $(x(k), y(k))$ the mean of the positions transmitted in $[t_k, t_k + \Delta t)$. If no position has been transmitted in this period, we set 0 to $n(k)$, otherwise it takes the value 1. To ensure reasonable runtimes for applying the method to every user, we limit the analysis to the first 20000 seconds of each user in both datasets.

Competitors. We evaluate the performance of the MPC method in comparison with the unprotected case, that is, a user transmitting only the actual positions, and also in comparison with Geo-I mechanism. We arbitrarily set $\varepsilon = 0.03$ in Eq. (1), which generates a high disturbance for Privamov's users but a low disturbance for Cabspotting's users. To make a fair comparison, we first run Geo-I for each user, and then, using the new positions generated by Geo-I, we compute their utility loss using Eq. (3). Finally, we use this value as an upper bound for the utility loss in the MPC method, i.e., we assign these values to $\bar{q}(k)$. This method insures fairness in comparing privacy values, as it allows for the same utility loss.

Other parameters. In our simulations, we vary the horizon of the prediction H, from 1 to 7. This means that we use a prediction between 0.5 and 3.5 min. Since we already observe a tendency for each predictor when $H = 7$, and given the low predictability of mobility over large horizons, higher values of H are not studied. Note that $H = 1$ means to optimize only the current time, then the results for this horizon will be the same for the three predictors used. Regarding the buffer dimension, we set $N = 30$, so we calculate the privacy value based on the last 15 minutes. This value corresponds to that given in Cerf et al. (2023), Primault et al. (2019) for the definition of a POI (the place where the user stays for at least 15 min).

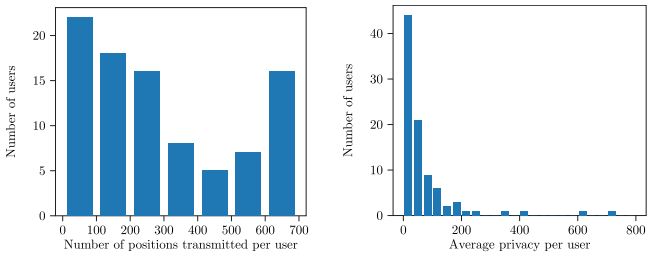


Fig. 3. Histogram of the number of positions sent and the average real privacy of Privamov users. This information corresponds to the first 20,000 s of resampled data.

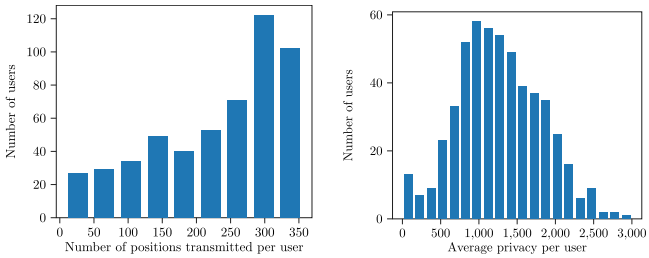


Fig. 4. Histogram of the number of positions sent and the average real privacy of Cabspotting users. This information corresponds to the first 20,000 s of resampled data.

4.1. Privamov dataset

We present statistical information that highlights the differences between Privamov users and Cabspotting users. This diversity allows us to demonstrate the robustness of the method applied to two datasets with different user characteristics. The total number of transmitted positions in this database is 7807730. Looking at the information per user, the average number of transmitted positions is 81330, with a high dispersion as shown in Table 1. In fact, the user with the lowest number of transmissions shared 126 data points, and the user with the highest number of transmissions shared 1123719 points, which is 4 orders of magnitude larger.

The elapsed time between the first and the last transmitted position is also a parameter with a high dispersion. The average per user is 2285 hours, but the user with the smaller recording time did so only for about 5 minutes, while the user with the highest time recording positions was 9514.4 h, which is approximately 396 days.

Regarding the **transmission period**, we calculate the time difference between two consecutive transmissions for each user at each transmission point. The average obtained per user is $303s$, i.e., each user transmits its position every 303 s *on average*. However, if we take for each user the lower *95th percentile* of the differences between two consecutive points, the average drops to 10.02 s. This is explained by the fact that the transmission period when a user has activated the application is 10 s, but when the application is turned off, it can take a long time to turn it on again, affecting the overall average. The summary of the above statistical information, along with other measures of dispersion, is shown in Table 1 columns labeled Raw.

After resampling, the total number of transmitted points was reduced to 2696695, which is 65.5% less. The values in the Raw columns are updated and displayed in the Resample columns of the Table 1.

In the interval $[0, 20000]$ over which we apply the method, there are a total of 27414 transmitted positions, which corresponds approximately to 228.5 h of continuous transmission, or 9.5 days. For the same period of time, each user has a different number of transmitted points. We removed 3 users that have less than 7 points after re-sampling as the transmitted data is too few and scattered to apply the method. In the end we have 93 users and the average number of positions transmitted per user is 289. The user who transmitted the smallest number of positions did so 8 times, and the user who transmitted the most did so 660 times. The variance of this data is large, as shown in Fig. 3. Regarding the average privacy computed for each user, before applying a protection mechanism, we observe in Fig. 3 that the majority of the values are lower than 100, and half of the users have a privacy lower than 35. This can be explained as Privamov users have a higher diversity of transportation means with low speed as walking or biking. Therefore, in 15 min, they cover less territory than a Cabspotting user who is only moving by car.

The statistical information reveals that each user is fairly represented in the dataset. Additionally, it is possible to note that Privamov users have low real privacy, and therefore, we expect a significant gain in privacy by applying the $MPC - H$ method.

4.2. Cabspotting dataset

We now present a statistical description of the Cabspotting dataset, highlighting two important differences between Cabspotting and Privamov users: the average period between two transmissions and the inherent privacy levels.

Cabspotting dataset (Piorkowski et al., 2022) contains 536 users and 11219955 transmitted positions. Although this dataset has 5 times more users than Privamov, it just contains 1.5 times more mobility data. The average of transmitted positions per user is 20932, which is lower than with Privamov. The dispersion of these data is also lower than in Privamov. The average period of transmission is $55s$. The statistical description of the dataset is shown in Table 2 Resampled columns.

After the resampling, the statistical indicators do not change much, as shown in the Resampled columns of Table 2. This could be explained by the fact that the average transmission time (54.9 s using 95%) is greater than the period imposed by the resampling (30 s). The transmitted positions are reduced to 10631783, which is only 5.25% less than before resampling.

We take the positions of each user in the first 20000s (the same windows of time taken for Privamov's users). In this period we apply the method to 123340 positions, which corresponds to 1027.8 h or 42.8 days of uninterrupted transmission. The distribution of the number of transmitted positions per user is shown in Fig. 4. The histogram of the average privacy per user reveals high privacy values for the majority of users. On average, the privacy values are 50 times higher than those of Privamov users. This is consistent with the fact that Cabspotting users are always moving by car, and therefore at a higher speed.

Similar to the Privamov dataset, all users are fairly represented when applying the method. An important difference is the privacy level. For Cabspotting's users, privacy is already high, and then we do not expect a significant gain in privacy as for Privamov's users. Moreover, after comparing the transmission periods, it was found that Privamov's users have an average period of 10 s, while Cabspotting's users have an average period of 55 s. For this reason, we applied a 30-second resampling to standardize the method.

4.3. Metrics

We now describe the metrics used to evaluate and compare the different protection strategies in the two datasets.

We denote by $p_v^H(k)$ the privacy of a user v at time-step t_k after applying the MPC method with a time horizon H using one of the three predictors. Analogously, we denote the privacy in the unprotected case and after applying Geo-I by $p_v(k)$ and $p_v^G(k)$, respectively. We then introduce two metrics, that measure the percentage gain at time t_k of the MPC method over the unprotected case and Geo-I. These correspond to:

$$gain_v(k,H) = \frac{p_v^H(k) - p_v(k)}{p_v(k)}, \quad gain_v^G(k,H) = \frac{p_v^H(k) - p_v^G(k)}{p_v^G(k)}.$$

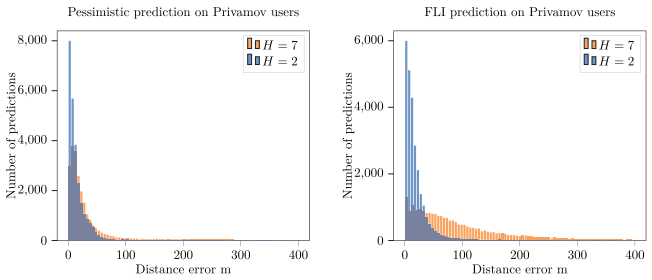


Fig. 5. Distance errors in future locations using Pessimistic and FLI predictors on the Privamov dataset.

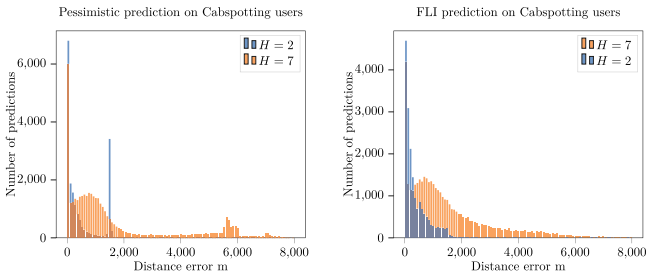


Fig. 6. Distance errors in future locations using Pessimistic and FLI predictors on the Cabspotting dataset.

We remark that in some instances, $p_v(k)$ could be 0 if a user did not move for N consecutive time-steps. In that case $gain_v(k, H)$ would be infinite, then, for statistical purposes we only use the times k when $p_v(k) > 0$.

To evaluate the total performance of the method for each user, we use:

$$gain_v(H) = \text{mean}\{gain_v(k, H) : k \in \{1, \dots, M\}, n(k) = 1\}, \tag{10}$$

$$gain_v^G(H) = \text{mean}\{gain_v^G(k, H) : k \in \{1, \dots, M\}, n(k) = 1\}, \tag{11}$$

that is, the average of the gains obtained at each time that a position is transmitted. With these expressions, we identify the variability of the results depending on the user. To evaluate a general performance, independent of the user, we use:

$$gain(H) = \text{mean}\{gain_v(k, H) : v \text{ a user}, k \in \{1, \dots, M\}, n(k) = 1\}, \tag{12}$$

$$gain^G(H) = \text{mean}\{gain_v^G(k, H) : v \text{ a user}, k \in \{1, \dots, M\}, n(k) = 1\}. \tag{13}$$

These expressions can be interpreted as the privacy gains that an arbitrary user should expect, in average, when using the method $MPC - H$.

In Section 5, we employ the previous metrics and show the obtained results.

4.4. Validation of predictors

Fig. 5 illustrates the performance of the two predictors on the Privamov dataset. This figure depicts the errors (in meters) obtained after applying each predictor to cases $H = 2$ and $H = 7$. Both predictors perform well for short time horizons, but the linear predictor (FLI) becomes less accurate as the time horizon increases. However, the prediction errors are almost all below 200 m, which is acceptable for many mobile applications. Note that the prediction accuracy is not the main objective of our approach: privacy protection is thoroughly evaluated in Section 5.

The performance of the predictors on the users of Cabspotting dataset is presented in Fig. 6. The figure shows errors that are reasonable for a good performance of many mobile applications. It is noted that the errors are larger than those observed in the Privamov dataset. However, this is to be expected given that Cabspotting users are always moving in a car and cover longer distances, and the sampling frequency of data is lower, which leads to larger errors. While these results allow for a better analysis of our MPC-based protection mechanism, prediction performance is not our main objective.

5. Experimental results

This section presents the results obtained by applying the MPC method with three different predictors and on two datasets. Section 5.1 illustrates the effect on a single user. Section 5.2 shows the average privacy gains independently of the user. Section 5.3 presents the results obtained per user. Section 5.4 presents the gain distribution for horizon $H = 3$, followed by the runtime results in Section 5.5. Experiments and analysis presented are reproducible using our openly available Python code https://github.com/ox217/Validation_MPC_optimal_location_privacy.

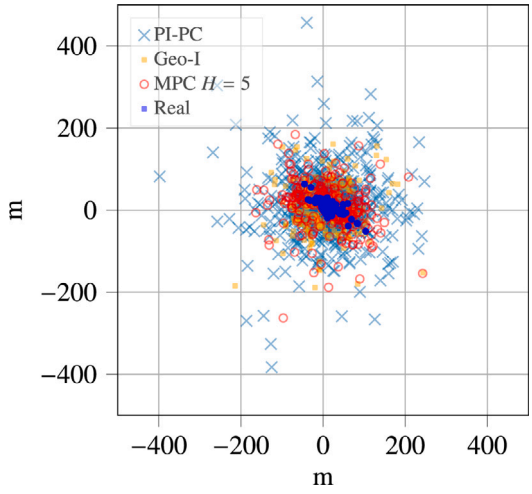


Fig. 7. Mobility trace for user 7 of Privamov.

5.1. Example over a single user

To demonstrate the efficiency of the method on an arbitrary user, we present the outcomes of $MPC - 5$, employing the predictor that provides precise future information, on user 7 in the Privamov dataset. In this specific instance, a comparison is also made with the PI-PC (Proportional Integral with Pre-Compensation) controller proposed by Cerf et al. (2021), for which the code is available for comparison purposes. The objective of this obfuscation mechanism is to maintain the privacy function around a target value, fixed at 75 for the present user. It is worth noting that the comparison with this approach cannot be fair and is thus only indicative, as the PI-PC has no constraint on its utility budget. In the following sections, we will only make a comparison with Geo-I. It should be noted that Geo-I serves as a benchmark of comparison for most works in the mobility privacy field, consequently, it represents an appropriate point of comparison for our work. Fig. 7 shows the real position for this user and the results after applying Geo-I, PI-PC and the MPC method. It is possible to see the effect of obfuscation by noting how the transmitted locations spread out in space.

Fig. 8 shows the temporal evolution of the privacy value for these positions, with Geo-I and $MPC - 5$ both using the same utility loss. We can see that, for this user, the privacy value obtained using the MPC method is almost always higher than the real and Geo-I methods. However, with regard to PI-PC, there are instances where the MPC

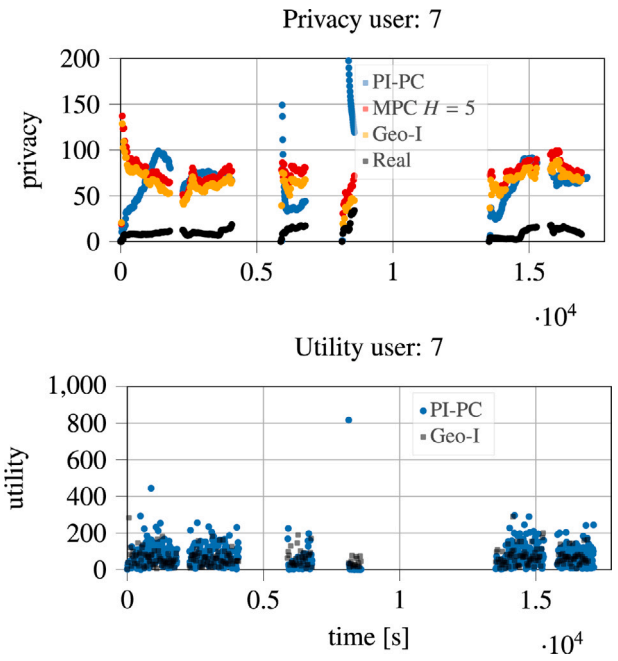


Fig. 8. The first image shows privacy for user 7. The bottom plot shows the utility loss from Geo-I, which is also used in the MPC method.

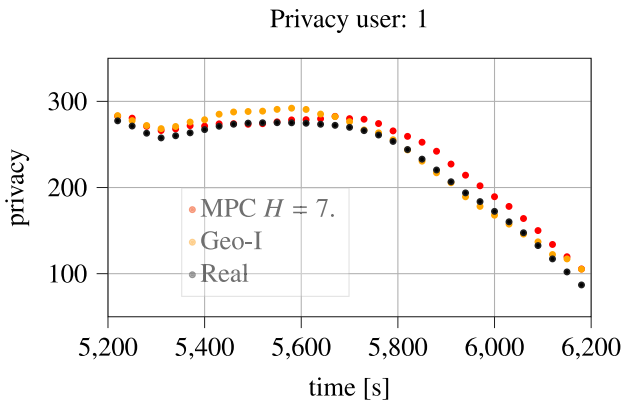


Fig. 9. Privacy for user 1 between 5200 s and 6200 s.

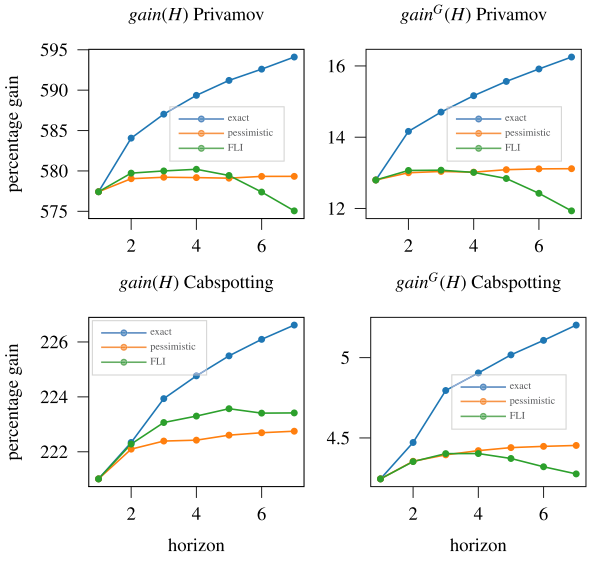


Fig. 10. Comparison using three predictors on Privamov (top line) and Cabspotting (bottom line) datasets. Left plots show $gain(H)$ (average privacy gain against real data using all users, see Eq. (12)) and right plots present $gain^G(H)$ (average privacy gain Geo-I data using all users, see Eq. (13)).

method exhibits considerably higher values, while in other instances, the values are markedly lower. A comparison of the utility loss reveals that the utility loss for Geo-I and PI-PC is similar, except in a few cases where the PI-PC controller has a very high peak.

In this example, the average privacy value in the unprotected case is 10.13 while with Geo-I it is 63.48. The metrics (10) and (11) computed for this user are $gain_7(5) = 8.52$ and $gain_7^G(5) = 0.12$ respectively. This means that using $MPC - 5$, the average privacy gain over the real positions is 852% and 12% over Geo-I.

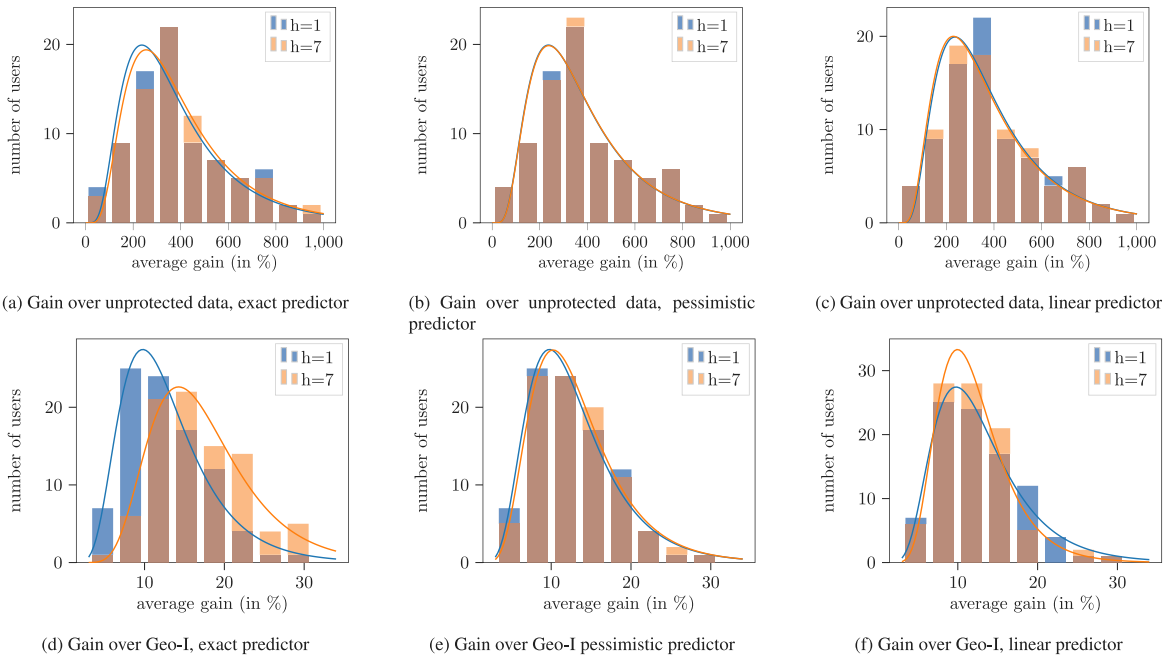
Fig. 9 illustrates the privacy of a second user (1 of Privamov) using the exact predictor. A noteworthy phenomenon occurs between 5400 and 5600 s. It can be observed that the Geo-I method is capable of surpassing MPC $H = 7$. This can be interpreted as an anticipatory period during which the MPC method sacrifices some instances with lower privacy in order to subsequently achieve a superior level of privacy. Note that the anticipation can reach longer horizons than those set in the MPC algorithm ($H = 7$) due to the recomputation of the optimization at each timestep in the MPC. The effects of this phenomenon are evident between 5600 s and 6200 s, where the MPC also outperforms Geo-I. During this period, the user velocity decreases (seen here through a reduction of the real privacy metric, e.g. measure of dispersion of the positions). This example illustrates the advantage of MPC and its dynamic approach to privacy management, anticipating future positions. This advantage is statistically validated in the following sections, which show that MPC exhibits superior performance compared to Geo-I *on average* but not on all data points, and performs better as the horizon increases (with the real predictor). Note that, however, the privacy with the MPC is always higher than that of unprotected data (real points).

The results of these two users are a proof of concept that our MPC-based solution allows to gain privacy compared to state-of-the-art solutions, with the same utility preservation. In the following, we will consolidate these results with extensive experimentation on two full datasets, and study the impact of the choice of the predictor.

5.2. Average privacy gain

Starting from this section, we present the results obtained with the three different predictors on both datasets when we apply the MPC method to every user of both datasets over the first 20,000 s of each user. In this section, we present the values obtained using the metrics $gain(H)$ and $gain^G(H)$, which summarize the performance over all users and instances executed per user. The results are shown in Fig. 10. This figure summarizes the performance of the method for each predictor and dataset with respect to the horizon H used. It is worth noting that the gain over real data is in a different order of magnitude than the gain over Geo-I. This is expected because the unprotected case has a larger margin for improvement than Geo-I, which is also a protection mechanism. When comparing the datasets, it was found that the gains obtained using Privamov users were approximately 2.5 times larger than those using Cabspotting users. This is not surprising because, as shown in Section 4.2, the privacy of Cabspotting users in the unprotected case is much greater than that of Privamov users.

In terms of the protection mechanism, the predictor that accurately predicts the future performs the best in both datasets. Its performance improves as H increases, and the difference between this predictor and the other two becomes bigger. While this is not a realistic solution, it demonstrates that the more we know about the future, the greater is the benefit of using the MPC method. In Fig. 10, it can be observed that the linear predictor performance is better than the pessimistic predictor for H from 1 to 4, except in one case, and worse for H from 5 to 7. This



(a) Gain over unprotected data, exact predictor

(b) Gain over unprotected data, pessimistic predictor

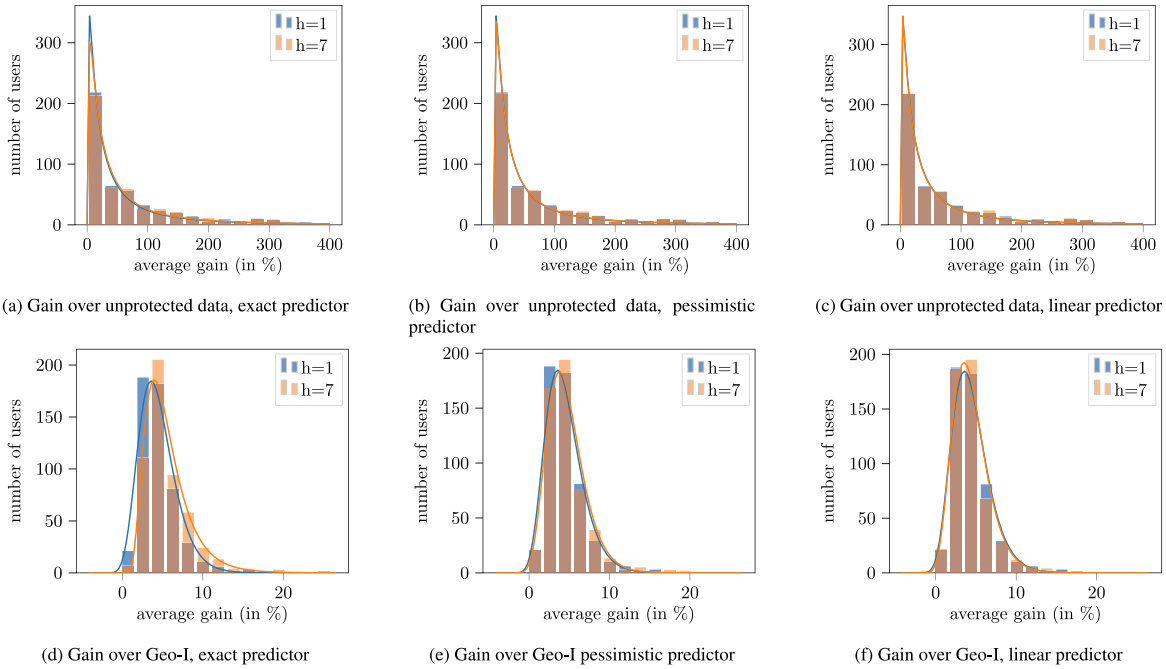
(c) Gain over unprotected data, linear predictor

(d) Gain over Geo-I, exact predictor

(e) Gain over Geo-I pessimistic predictor

(f) Gain over Geo-I, linear predictor

Fig. 11. Histograms of privacy gain per **Privamov** user using three predictors.



(a) Gain over unprotected data, exact predictor

(b) Gain over unprotected data, pessimistic predictor

(c) Gain over unprotected data, linear predictor

(d) Gain over Geo-I, exact predictor

(e) Gain over Geo-I pessimistic predictor

(f) Gain over Geo-I, linear predictor

Fig. 12. Histograms of privacy gain per **Cabspotting** user using three predictors.

can be explained by the fact that the linear predictor is more accurate for lower values of H, and becomes less accurate as H increases. This highlights the fact that improvement of the predictor performance is a promising research direction to further improve privacy gains. In contrast, the pessimistic predictor remains relatively constant regardless of H. Despite these differences, both predictors have similar gain values. Additionally, for short prediction horizons, their performance is comparable to that of the exact predictor.

To conclude this section, we note from Fig. 10 that the horizons $H = 3$ or $H = 4$ would be the best to choose in an online implementation. The exact predictor demonstrates that, in the ideal case, performance increases with H. However, for $H \geq 4$, the linear predictor starts to perform worse and the pessimistic predictor has the same performance, sometimes even better, but requires more computational resources.

These indicators demonstrate the effectiveness of our method, consistently yielding positive results. The following section provides further details on the gains achieved for each individual user.

5.3. Results analysis by user

In this section, we present statistical results per user. For this purpose, we use the metrics $gain_v(H)$ and $gain_v^G(H)$ with $H = 1$ and $H = 7$. Figs. 11 and 12 displays the frequencies of the values of these metrics among the users of each dataset. Each figure represents a

different dataset. The first row of each figure shows the gain over the unprotected data, and the second row shows the gain over Geo-I. Each column corresponds to a different predictor used: exact, pessimistic and linear respectively. Each histogram was fitted with a lognormal distribution, represented by a solid line in each image.

Both figures confirm the significant privacy gain of the $MPC-H$ method compared to the unprotected case, particularly for Privamov users. The privacy gains for Cabspotting users are more concentrated in lower values. The reason for this has already been discussed in previous sections. Compared to the competitor Geo-I, the gain is not as high, but it is always positive regardless of the predictor and horizon used. This evidence indicates that, on average, the $MPC-H$ method outperforms the Geo-I method for each user.

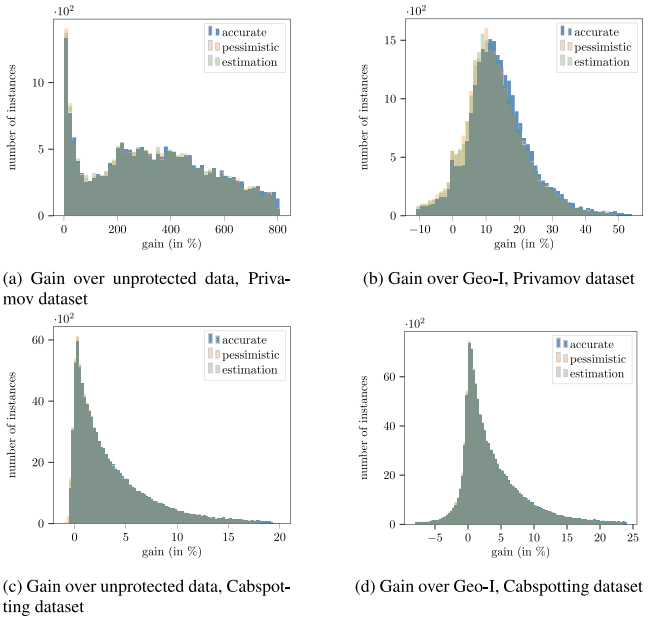
Regarding the predictor used, the gain difference over unprotected data is unclear. In fact, Fig. 10 shows that the relative difference among predictors for this case is not as significant as for the Geo-I case. When examining the gain over Geo-I distributions, the effect becomes clearer, which supports the conclusions of Section 5.2. In fact, in both datasets, the exact predictor performs best when $H=7$. This conclusion is based on the fact that the respective histograms are skewed to the right. This confirms the effect of the prediction horizon on the gain, as it shifts to the right when transitioning from $H=1$ to $H=7$. The pessimistic and linear predictors exhibit similar performance, with only a slight difference. When passing from $H=1$ to $H=7$, the effect is almost imperceptible when using the pessimistic predictor, but for the linear one, the gain is more concentrated to the left. This is confirmed when examining the lognormal curves. In Figs. 11 and 12 images (b) and (e) show almost identical histograms and lognormal curves for the two values for H. However, in images (c) and (f), the lognormal curve of $H=7$ has a higher peak than the curve for $H=1$.

5.4. Results analysis along time

In this section, we present the results that have the highest level of granularity. For every user v and instant k, we compute the values of $gain_v(k,H)$ and $gain_v^G(k,H)$. Fig. 13 presents the distribution of these values, independent of the users. As mentioned above, the optimal choice for online implementation would be to set H to either 3 or 4. Therefore, we will only present the results for $H=3$.

When comparing the gain over unprotected data and Geo-I in Fig. 13, differences in magnitude can be observed. The main difference is that the gain over unprotected data is almost always positive (Fig. 13(a) and (c)), but for the gain over Geo-I, there are non-negligible instances where the gain is negative (Fig. 13(b) and (d)). There is a possible explanation to this effect: a negative gain value over unprotected data is due to a non-convergence of the optimization solver. This assertion is supported by the fact that the deobfuscation option is always feasible and that it always yields a gain of 0, while the negative gain values observed over Geo-I can be attributed to the nature of the underlying algorithm. Given that our algorithm relies on future predictions, it may sacrifice some instances that achieve a modest degree of privacy improvement in favor of significantly higher improvements in the future. It is also important to note that knowing only three steps may provide short-term privacy benefits, but as the horizon of prediction is not so large, it may result in the generation of a trace that is not optimal for the minimization of privacy in the medium term. This is because using a large horizon allows producing a trace that better predicts future scenarios with low privacy.

For a horizon $H=3$, there is no significant difference between the predictors. When comparing the results in both datasets, it is again verified that the privacy improvements are more important for Privamov users than for Cabspotting users.



(a) Gain over unprotected data, Privamov dataset

(b) Gain over Geo-I, Privamov dataset

(c) Gain over unprotected data, Cabspotting dataset

(d) Gain over Geo-I, Cabspotting dataset

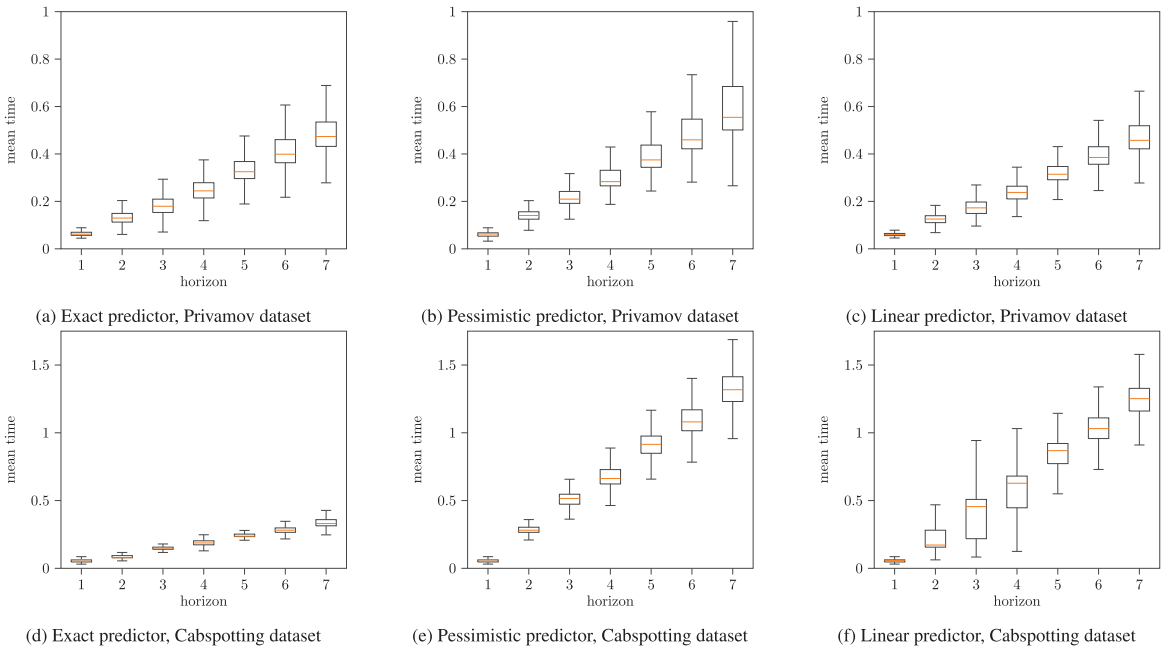
Fig. 13. Comparison using three predictors and horizon $H=3$. The two plot at the top correspond to Privamov users, and the two at the bottom to Cabspotting users.

5.5. Comparison of execution times

To conclude the results section, we present a box plot in Fig. 14 comparing the statistics of runtime taken by the method when applied at a singular time t_k. We compare the execution times between the different predictors. For Privamov, they take similar execution times (Fig. 14(a), (b) and (c)), while for Cabspotting, the method using the exact predictor (Fig. 14(d)) takes notably less time than the others (Fig. 14(e) and (f)). Slowest cases for Privamov dataset are around 1 s and for Cabspotting around 1.5 s. From the same figure, it is possible to note that in general, the average runtime for Privamov is lower than that for Cabspotting. Finally, it is possible to conclude that the execution time is increasing with the horizon H, which is expected from the fact that when increasing H, the optimization problem solver increases the number of variables and constraints. Overall, since the execution times are significantly lower than the time between two transmission positions (30 s), so they are reasonable for our setup.

6. Conclusion

In this paper, the problem of obfuscating a mobility position, maximizing privacy and minimizing the utility loss, has been considered. This work presents an extension of Molina et al. (2023a), wherein the method was tested on a single user, assuming the implausible case that the future positions were known. This paper presents an extensive evaluation of its performance on two datasets, Privamov and Cabspotting, which contain mobility data traces from real users of mobile devices. Both datasets define two types of users. While Privamov contains mobile users employing various means of transport, Cabspotting contains only users moving by car. This difference affects both the speed and spatial distortion of data, and therefore their privacy risk and their need for protection vary. Additionally, this paper also employs three predictors to forecast the users' movements. The first predictor provides the precise future location and, although it cannot be implemented online, it serves as an upper bound on the amount of privacy that can be gained. The second predictor assumes that the user is moving towards positions that minimize privacy, while the third one uses a linear method to estimate future positions. Both the second and the third approaches could be implemented in a realistic scenario.



(a) Exact predictor, Privamov dataset (b) Pessimistic predictor, Privamov dataset (c) Linear predictor, Privamov dataset

(d) Exact predictor, Cabspotting dataset (e) Pessimistic predictor, Cabspotting dataset (f) Linear predictor, Cabspotting dataset

Fig. 14. The figure displays box plots that present statistical information on the runtime of the method applied at singular times t_k. Each line represents a dataset and each column represents a predictor.

Evaluation results, conducted using the metrics proposed in section 4.3, demonstrate that the MPC-based method enhances the privacy by at least 580% on average, in comparison to the unprotected position and by between 12% and 17% in relation to the competitor Geo-I, for users of the Privamov dataset. For users of the Cabspotting dataset, the average privacy gain is at least 221% over the actual privacy and between 4% and 6% over using Geo-I. The significant difference between the results of both datasets can be attributed to the fact that Cabspotting users tend to have higher speeds, resulting in higher privacy levels; thus, the protection mechanism generally has lower effects.

Regarding the three predictors used, it is clear, in both datasets, that using exact knowledge of the future, the MPC performance increases as more information about the future is employed. However, the other two predictors also show promising results. In fact, for a short prediction horizon, their performance is close to that of the exact predictor. Assuming pessimistic future movements results in an average privacy value that is almost constant with respect to the prediction horizon. Using a linear predictor for future positions shows that a trade-off is to be found on the horizon to be considered. Indeed, while MPC performance benefits from a longer horizon, the precision of the prediction decreases for points farther in the future as expected. The search for a more accurate predictor with low execution time and complexity is planned for future work. Additional future work could explore adapting the method to preserve additional user characteristics that enhance privacy, such as movement velocity and acceleration. Implementation of the approach on a real smartphone application should be additionally considered.

CRediT authorship contribution statement

Emilio Molina: Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Mirko Fiacchini:** Writing – review & editing, Supervision, Methodology, Investigation, Formal analysis, Conceptualization. **Arthur Goarant:** Writing – review & editing, Validation, Software, Methodology. **Rémy Raes:** Writing – review & editing, Supervision, Methodology. **Sophie Cerf:** Writing – review & editing, Writing – original draft, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Bogdan Robu:** Writing – review & editing, Methodology, Investigation, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work has been partially supported by MIAI@Grenoble Alpes (ANR19-P3IA-0003) and by the French LabEx PERSYVAL-Lab (ANR-11-LABX-0025-01).

References

Andrés, Miguel E., Bordenabe, Nicolás E., Chatzikokolakis, Konstantinos, & Palamidessi, Catuscia (2013). Geo-indistinguishability: differential privacy for location-based systems. In *Proceedings of the 2013 ACM SIGSAC conference on computer & communications security* (pp. 901–914). New York, NY, USA: Association for Computing Machinery, ISBN: 9781450324779, http://dx.doi.org/10.1145/2508859.2516735.

Ben Mokhtar, Sonia, Boutet, Antoine, Bouzouina, Louafi, Bonnel, Patrick, Brette, Olivier, Brunie, Lionel, Cunche, Mathieu, D 'Alu, Stephane, Primault, Vincent, Raveneau, Patrice, Rivano, Herve, & Stanica, Razvan (2017). PRIVA'MOV: Analysing Human Mobility Through Multi-Sensor Datasets. In *netMob 2017*. Milan, Italy: https://projet.liris.cnrs.fr/privamov/project/dataset, URL https://inria.hal.science/hal-01578557.

Bhati, Bhawani Shanker, & Eckhoff, David (2019). Synthetic mobility traces. In *Encyclopedia of cryptography, security and privacy* (pp. 1–5). Berlin, Heidelberg: Springer Berlin Heidelberg, ISBN: 978-3-642-27739-9, http://dx.doi.org/10.1007/978-3-642-27739-9_1734-1.

Biswas, Sayan, & Palamidessi, Catuscia (2024). PRIVIC: A privacy-preserving method for incremental collection of location data. *Proceedings on Privacy Enhancing Technologies*, [ISSN: 2299-0984] 582–596. http://dx.doi.org/10.56553/popets-2024-0033, URL https://petsymposium.org/popets/2024/popets-2024-0033.php.

Bordenabe, Nicolás E., Chatzikokolakis, Konstantinos, & Palamidessi, Catuscia (2014). Optimal geo-indistinguishable mechanisms for location privacy. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security* (pp. 251–262). New York, NY, USA: Association for Computing Machinery, ISBN: 9781450329576, http://dx.doi.org/10.1145/2660267.2660345.

Cerf, Sophie, Bouchenak, Sara, Robu, Bogdan, Marchand, Nicolas, Primault, Vincent, Mokhtar, Sonia Ben, Boutet, Antoine, & Chen, Lydia Y. (2021). Automatic privacy and utility preservation for mobility data: A nonlinear model-based approach. *IEEE Transactions on Dependable and Secure Computing*, *18*(1), 269–282. http://dx.doi.org/10.1109/TDSC.2018.2884470.

Cerf, Sophie, Robu, Bogdan, Marchand, Nicolas, & Bouchenak, Sara (2023). Privacy protection control for mobile apps users. *Control Engineering Practice*, [ISSN: 0967-0661] *134*, Article 105456. http://dx.doi.org/10.1016/j.conengprac.2023.105456, URL https://www.sciencedirect.com/science/article/pii/S0967066123000254.

Chatzikokolakis, Konstantinos, Palamidessi, Catuscia, & Stronati, Marco (2014). A predictive differentially-private mechanism for mobility traces. In Emiliano De Cristofaro, & Steven J. Murdoch (Eds.), *Privacy enhancing technologies* (pp. 21–41). Cham: Springer International Publishing, ISBN: 978-3-319-08506-7, http://dx.doi.org/10.1007/978-3-319-08506-7_2.

Chatzikokolakis, Konstantinos, Palamidessi, Catuscia, & Stronati, Marco (2015). Constructing elastic distinguishability metrics for location privacy. *Proceedings on Privacy Enhancing Technologies*, [ISSN: 2299-0984] *2015*(2), 156–170. http://dx.doi.org/10.1515/popets-2015-0023.

Dwork, Cynthia (2006). Differential privacy. In Michele Bugliesi, Bart Preneel, Vladimiro Sassone, & Ingo Wegener (Eds.), *Automata, languages and programming* (pp. 1–12). Berlin, Heidelberg: Springer, ISBN: 978-3-540-35908-1, http://dx.doi.org/10.1007/11787006_1.

Gambs, Sébastien, Killijian, Marc-Olivier, & del Prado Cortez, Miguel Núñez (2010). Show me how you move and I will tell you who you are. In *Proceedings of the 3rd ACM sIGSPATIAL international workshop on security and privacy in GIS and LBS* (pp. 34–41). New York, NY, USA: Association for Computing Machinery, ISBN: 9781450304351, http://dx.doi.org/10.1145/1868470.1868479.

Hariharan, Ramaswamy, & Toyama, Kentaro (2004). Project lachesis: Parsing and modeling location histories. In Max J. Egenhofer, Christian Freksa, & Harvey J. Miller (Eds.), *Geographic information science* (pp. 106–124). Berlin, Heidelberg: Springer, ISBN: 978-3-540-30231-5, http://dx.doi.org/10.1007/978-3-540-30231-5_8.

Jiang, Hongbo, Li, Jie, Zhao, Ping, Zeng, Fanzi, Xiao, Zhu, & Iyengar, Arun (2021). Location privacy-preserving mechanisms in location-based services: A comprehensive survey. *ACM Computing Surveys*, [ISSN: 0360-0300] *54*(1), http://dx.doi.org/10.1145/3423165.

Koufogiannis, Fragkiskos, & Pappas, George J. (2016). Location-dependent privacy. In *2016 IEEE 55th conference on decision and control* (pp. 7586–7591). http://dx.doi.org/10.1109/CDC.2016.7799441.

Krumm, John (2007). Inference attacks on location tracks. In Anthony LaMarca, Marc Langheinrich, & Khai N. Truong (Eds.), *Pervasive computing* (pp. 127–143). Berlin, Heidelberg: Springer, ISBN: 978-3-540-72037-9, http://dx.doi.org/10.1007/978-3-540-72037-9_8.

Ma, Zhuo, Xu, Shuai, Liu, Bo, & Cao, Jiuxin (2023). LPP2KL: Online location privacy protection against knowing-and-learning attacks for LBSs. *IEEE Transactions on Computational Social Systems*, *10*(1), 234–245. http://dx.doi.org/10.1109/TCSS.2022.3142078.

Min, Minghui, Zhu, Haopeng, Li, Shiyin, Zhang, Hongliang, Xiao, Liang, Pan, Miao, & Han, Zhu (2024). Semantic adaptive geo-indistinguishability for location privacy protection in mobile networks. *IEEE Transactions on Vehicular Technology*, 1–6. http://dx.doi.org/10.1109/TVT.2024.3354881.

Molina, Emilio, Fiacchini, Mirko, Cerf, Sophie, & Robu, Bogdan (2023). Optimal privacy protection of mobility data: a predictive approach. In *IFAC WC 2023 - 22nd IFAC world congress*. Yokohama, Japan: IFAC, URL https://hal.science/hal-04040962.

Molina, Emilio, Fiacchini, Mirko, Cerf, Sophie, & Robu, Bogdan (2023). React to the Worst: Lightweight and proactive protection of location privacy. *IEEE Control Systems Letters*, *7*, 2371–2376. http://dx.doi.org/10.1109/LCSYS.2023.3286989, URL https://hal.science/hal-04128118.

Niu, Ben, Li, Qinghua, Wang, Hanyi, Cao, Guohong, Li, Fenghua, & Li, Hui (2022). A framework for personalized location privacy. *IEEE Transactions on Mobile Computing*, *21*(9), 3071–3083. http://dx.doi.org/10.1109/TMC.2021.3055865.

Oya, Simon, Troncoso, Carmela, & Pérez-González, Fernando (2017). Back to the drawing board: Revisiting the design of optimal location privacy-preserving mechanisms. In *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security* (pp. 1959–1972). New York, NY, USA: Association for Computing Machinery, ISBN: 9781450349468, http://dx.doi.org/10.1145/3133956.3134004.

Piorkowski, Michal, Sarafijanovic-Djukic, Natasa, & Grossglauser, Matthias (2022). CRAWDAD epfl/mobility. In *IEEE dataport*. http://dx.doi.org/10.15783/C7J010.

Primault, Vincent, Boutet, Antoine, Mokhtar, Sonia Ben, & Brunie, Lionel (2019). The long road to computational location privacy: A survey. *IEEE Communications Surveys & Tutorials*, *21*(3), 2772–2793. http://dx.doi.org/10.1109/COMST.2018.2873950.

Qiu, Shuyuan, Pi, Dechang, Wang, Yanxue, & Liu, Yufei (2023). Novel trajectory privacy protection method against prediction attacks. *Expert Systems with Applications*, [ISSN: 0957-4174] *213*, Article 118870. http://dx.doi.org/10.1016/j.eswa.2022.118870, URL https://www.sciencedirect.com/science/article/pii/S0957417422018887.

Raes, Rémy, Ruas, Olivier, Luxey-Bitri, Adrien, & Rouvoy, Romain (2024). Compact Storage of Data Streams in Mobile Devices. In *Proceedings of the 24th international conference on distributed applications and interoperable systems (dAIS'24)*, *dAIS'24 - 24th international conference on distributed applications and interoperable systems*. Groningen, Netherlands: LNCS, URL https://hal.science/hal-04535716.

Shokri, Reza, Theodorakopoulos, George, Troncoso, Carmela, Hubaux, Jean-Pierre, & Le Boudec, Jean-Yves (2012). Protecting location privacy: optimal strategy against localization attacks. In *Proceedings of the 2012 ACM conference on computer and communications security* (pp. 617–627). New York, NY, USA: Association for Computing Machinery, ISBN: 9781450316514, http://dx.doi.org/10.1145/2382196.2382261.

Toch, Eran, Lerner, Boaz, Ben-Zion, Eyal, & Ben-Gal, Irad (2019). Analyzing large-scale human mobility data: a survey of machine learning methods and applications. *Knowledge and Information Systems*, *58*, 501–523. http://dx.doi.org/10.1007/s10115-018-1186-x.

Xiao, Yonghui, & Xiong, Li (2015). Protecting locations with differential privacy under temporal correlations. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security* (pp. 1298–1309). New York, NY, USA: Association for Computing Machinery, ISBN: 9781450338325, http://dx.doi.org/10.1145/2810103.2813640.

Yu, Lei, Liu, Ling, & Pu, Calton (2017). Dynamic differential location privacy with personalized error bounds.. In *Network and distributed system security symposium*. http://dx.doi.org/10.14722/ndss.2017.23241.

Zhan, Yuting, Haddadi, Hamed, & Mashhadi, Afra (2023). Privacy-aware adversarial network in human mobility prediction. *Proceedings on Privacy Enhancing Technologies*, [ISSN: 2299-0984] 556–570. http://dx.doi.org/10.56553/popets-2023-0032, URL https://petsymposium.org/popets/2023/popets-2023-0032.php.