# Quantifying Interdependent Privacy Risks with Location Data

Alexandra-Mihaela Olteanu, *Student Member, IEEE,* Kévin Huguenin, *Member, IEEE,*
Reza Shokri, *Member, IEEE,* Mathias Humbert, *Member, IEEE,* Jean-Pierre Hubaux, *Fellow, IEEE*

**Abstract**—Co-location information about users is increasingly available online. For instance, mobile users more and more frequently report their *co-locations* with other users in the messages and in the pictures they post on social networking websites by tagging the names of the friends they are with. The users' IP addresses also constitute a source of co-location information. Combined with (possibly obfuscated) location information, such co-locations can be used to improve the inference of the users' locations, thus further threatening their location privacy: As co-location information is taken into account, not only a user's reported locations and mobility patterns can be used to localize her, but also those of her friends (and the friends of their friends and so on). In this paper, we study this problem by quantifying the effect of co-location information on location privacy, considering an adversary such as a social network operator that has access to such information. We formalize the problem and derive an optimal inference algorithm that incorporates such co-location information, yet at the cost of high complexity. We propose some approximate inference algorithms, including a solution that relies on the belief propagation algorithm executed on a general Bayesian network model, and we extensively evaluate their performance. Our experimental results show that, even in the case where the adversary considers co-locations of the targeted user with a single friend, the median location privacy of the user is decreased by up to 62% in a typical setting. We also study the effect of the different parameters (e.g., the settings of the location-privacy protection mechanisms) in different scenarios.

**Index Terms**—Location privacy; co-location; inference; social networks

## 1 Introduction

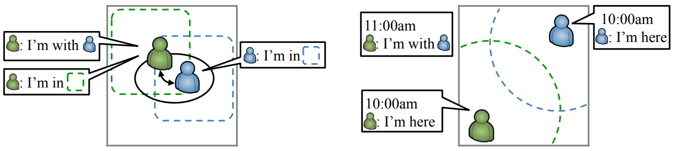
Social networks, and in particular location-based social networks, have become immensely popular. Every day, millions of users post information, including their locations, about themselves, but also about their friends. An emerging trend, which is the focus of this paper, is to report co-locations with other users on social networks, *e.g.*, by tagging friends on pictures they upload or in the messages they post.[1] For instance, our preliminary survey involving 132 Foursquare users, recruited through Amazon Mechanical Turk, reveals that 55.3% of the participants report co-locations in their check-ins and that for the users who do so, on average, 2.84%±0.06 of their check-ins contain co-location information. In fact, co-location information can be obtained in many different ways, such as automatic face recognition on pictures (which contains the time and location at which the picture was taken in their EXIF data, *e.g.*, Facebook's Photo Magic [2]), Bluetooth-enabled device sniffing and reporting neighboring devices. Similarly, users who connect from the same IP address are likely to be attached to the same Internet access point, thus providing evidence of their co-location.

Attacks exploiting both location and co-location information (as mentioned in [3]) can be quite powerful, as we show in this paper. Figure 1 depicts and describes two instances in which co-location can improve the performance of a localization attack, thus degrading the location privacy of the users involved. It is clear that the proper exploitation of such information by an attacker can be complex because he has to consider jointly the (co-)location information collected about a potentially large number of users. This is due to the fact that, in the presence of co-location information, a user's location is correlated with that of her friends, which is in turn correlated to that of their own friends and so on.

This family of attacks and their complexity is precisely the focus of this paper. More specifically, we make the following four contributions: (1) We identify and formalize the localization problem with co-location information, we propose an optimal inference algorithm and analyze its complexity. We show that, in practice, the optimal inference algorithm is intractable due to the explosion of the state space size. (2) We describe how an attacker can drastically reduce the computational complexity of the attack by means of well-chosen approximations. We present a polynomial-time heuristic based on a limited set of considered users (i.e., optimal inference with the data of only two or three users) and an approximation based on the belief propagation (BP) algorithm executed on a general Bayesian network model of the problem (approximate inference with the data of all the

- *Manuscript received: April 14, 2016;*
- *This article is a revised and extended version of a paper that appears in the Proceedings of the 14th Privacy Enhancing Technologies Symposium (PETS 2014) Olteanu* et al. *[1].*
- *A.-M. Olteanu and J.-P. Hubaux are with EPFL, Lausanne, Switzerland (e-mail: alexandramihaela.olteanu@epfl.ch; jean-pierre.hubaux@epfl.ch).*
- *K. Huguenin is with LAAS-CNRS, Université de Toulouse, CNRS, France (e-mail: kevin.huguenin@laas.fr).*
- *R. Shokri is with University of Texas at Austin, TX, USA (e-mail: shokri@cs.utexas.edu).*
- *M. Humbert is with Saarland University, Saarbrücken, Germany (e-mail: humbert@cs.uni-saarland.de).*

[1] Note that the fact that a users tags one of her friends in a post does not necessarily mean that they are co-located; our formalism takes this fact into account. This is one of the novelty of our extended version.



(a) (b)

Fig. 1. Examples showing how co-location information can be detrimental to privacy. (a) A user reports being in a given area, and a second user reports being in another (overlapping) area and that she is co-located with the first user. By combining these pieces of information, an adversary can deduce that both users are located in the intersection of the two areas, thus narrowing down the set of possible locations for both of them. (b) Two users (initially apart from each other, at 10am) declare their exact individual location. Later (at 11am), they meet and report their co-location without mentioning where they are. By combining these pieces of information, the adversary can infer that they are at a place that is reachable from both of the initially reported locations in the amount of time elapsed between the two reports.

users). (3) Using a mobility dataset, we evaluate and compare the performance of the different solutions in different scenarios, with different settings. The belief propagation-based solution, which does not appear in the first version of this work [1], gives significantly better results (in terms of the performance of the inference) than the heuristic. (4) We propose and evaluate some countermeasures (i.e., social-aware location-privacy protection mechanisms) including fake co-locations reporting and coordinated location disclosure. This last contribution also constitutes new content with respect to the first version of this work [1]. In this revised and extended version, we also update the formalism and the evaluation to take into account the fact that users can report being co-located when, in fact, they are not. Our experimental results show that, even in the case where the adversary considers co-locations with only a single friend of the targeted user, the median location privacy of the user is decreased by up to 62% in a typical setting. Even in the case where a user does not disclose any location information, her privacy can decrease by up to 21% due to the information reported by other users. A paramount finding of our work is that users partially lose control over their location privacy as co-locations and individual location information disclosed by other users substantially affect their own location privacy. Our experimental results also show that a simple countermeasure (i.e., coordinated location disclosure) can reduce the privacy loss by up to 50%. To the best of our knowledge, this is the first attempt to quantify the effects of co-location information that stems from social relationships, on location privacy; thus making a connection between OSNs and location privacy.

The remainder of the paper is organized as follows. In Section 2, we define and formalize the system model. In Section 3, we present the optimal localization attack for $N$ users and assess its complexity. In Section 4, we show how this complexity can be reduced by means of approximations. In Section 6, we briefly analyze the co-location problem from a differential privacy perspective. In Section 7, we report on the experimental evaluation of the localization attack with co-locations. In Section 5, we propose and evaluate some countermeasures. In Section 8, we survey the related work. In Section 9, we conclude the paper and suggest directions for the future work.

## 2 System Model and Formalization

We consider a set of mobile users who move in a given geographical area. While on the go, users make use of some online services to which they communicate potentially obfuscated location (*i.e.,* where they are) and co-location information (*i.e.,* who they are with). Note that such information could be communicated unintentionally by the users (e.g., leaked from their IP addresses) without their even knowing it. We consider that a curious service provider (referred to as the adversary) wants to infer the location of the users from this information, hence tracking them over time. In order to carry out the inference attack, based on which the location privacy of the users is evaluated, the adversary would model the users as described below. Our model is built upon [4] and uses similar notations.

### 2.1 Users

We consider a set $\mathcal{U} = \{u_1, \ldots, u_N\}$ of $N$ mobile users who move within a given geographical area that is partitioned into $M$ regions (locations) $\mathcal{R} = \{R_1, \ldots, R_M\}$. Time is discrete and we consider the state of the system (including the locations of the users) at the successive time instants $\{1, \ldots, T\}$. The region in which a user $u \in \mathcal{U}$ is at time instant $t \in \{1, \ldots, T\}$ is called the *actual location* of the user and is denoted by $a_u(t)$. The mobility of the users is modeled by a first order time-homogeneous Markov chain. We denote by $p_u(\rho, r)$ the probability that user $u$ moves from region $\rho$ to region $r$ during one time instant, and by $\pi_u(r)$ the probability that user $u$ is in region $r$ at time $t$ (*i.e.,* the stationary distribution of $p_u$). We call a co-location the fact that two users are at the same location at some point in time. The fact that users $u$ and $v$ are co-located at time $t$ means that $a_u(t) = a_v(t)$; we denote by $u \leftrightsquigarrow_t v$ the fact that a co-location between users $u$ and $v$ at time $t$ is reported (by either of them), and we consider an associated binary variable $c_{u,v}(t)$; specifically, $c_{u,v}(t) = c_{v,u}(t) = 1$ if $u \leftrightsquigarrow_t v$ and $c_{u,v}(t) = c_{v,u}(t) = 0$ otherwise. Note, however, that the fact that a co-location is reported does not necessarily mean that the users are really co-located (for instance, on OSNs, users sometimes tag a friend in a picture just to get their attention, although they are not actually together). We consider the process of users reporting co-location information to be probabilistic. Specifically, for any pair of users $u$ and $v$, the probability of reporting a co-location, knowing both their actual locations is denoted by

$$g_{u,v}(r, r') \triangleq \Pr\left(u \leftrightsquigarrow_t v \,|\, a_u(t) = r, a_v(t) = r'\right) \qquad (1)$$

This assumes that co-locations reported by a user at different time instants are being reported independently of each other and of those reported by other users. Additionally, we assume that the reporting process for any user does not depend on time. Intuitively, this co-location reporting function can incorporate social ties (users report co-locations on social networks only with their friends), selective reporting of co-location (not every time that Alice is with Bob in the same location does she report that on their favorite social network) as well as erroneous co-locations

(Alice might tag Bob in a picture, even though he is not really in that picture). Examples of erroneous co-locations also include the case where Alice and Bob have the same IP address but they are not together (e.g., they make use of the same proxy)–more generally it includes the false positives of the underlying co-location detection technique used by the adversary–as well as fake co-locations possibly reported by users to protect their privacy. The probabilistic co-location reporting function is assumed to be known to the adversary; in practice, it could be learned from models of the users' behaviors or from ground-truth data, or, when applicable, from theoretical models of the underlying technical co-location detection method. Concrete examples of co-location reporting functions are given in Section 7. We assume all user-reported co-locations are observed by an adversary.

### 2.2 Location-Privacy Protection Mechanisms

In order to protect their privacy, we assume that users rely on location-privacy protection mechanisms (LPPM) for obfuscating their individual location information before they communicate it to an online service provider. We denote by $u \,@_t\, r'$ the fact that user $u$ reports being at location $r'$ at time $t$ to the online service. The online service observes only the obfuscated location of the users, which we denote by $o_u(t)$ for user $u$ at time $t$. We denote by $\mathcal{R}'$ the set of obfuscated locations; typically $\mathcal{R}'$ is the power set of $\mathcal{R}$, as LPPMs can return a set of locations instead of only one location. Typical LPPMs replace the actual location of a user with another location (*i.e.,* adding noise to the actual location) or merge several regions (*i.e.,* reducing the granularity of the reported location). We model an LPPM by a function that maps a user's actual location to a random variable that takes values in $\mathcal{R}'$, that is, the user's obfuscated location. This means that the locations of a user at different time instants are obfuscated independently of each other and of those of other users. This also means that the way a user's locations are obfuscated does not depend on time. Formally, an LPPM is defined by the function $f_u(r, r')$ that denotes the probability that the LPPM used by $u$ obfuscates location $r$ to $r'$, *i.e.,* $\Pr\left(o_u(t) = r' \mid a_u(t) = r\right)$. Excluding the co-location information, our model corresponds to a hidden Markov model (HMM) [5]. We assume that co-location information is not obfuscated and users do not rely on pseudonyms.[2] We denote by $\mathbf{o}(t)$ the vector of the observed locations of all the users at time $t$. More generally, we use bold notations to denote a vector of values of all users. We define $C_t = \{c_{u,v}(t)\}_{u,v\in\mathcal{U}}$ and $C = \bigcup_{t=1..T} C_t$.

### 2.3 Adversary

The adversary, typically an online service provider (or an external observer who has access to this information, *e.g.,* another user of the social network), has access to the observed locations and co-locations of one or several users and seeks to locate users, at a given time instant, namely, carry out a *localization attack*[3]. Because of the co-location information, the locations of the users are not independent, thus when attacking the location of a given user, the adversary takes into account information potentially about all the users. The attack is performed *a posteriori*, meaning that the adversary has access to the observed traces over the complete period, namely $\{\mathbf{o}(t)\}_{t=1..T}$ and $C$, at the time of the attack. In addition to the observations during the time period of interest (*i.e.,* $\{1, \ldots, T\}$), the adversary has access to some of the users' past location traces, from which he builds individual mobility profiles for these users, under the form of transition probabilities $\{p_u\}_{u\in\mathcal{U}}$. See [4] for more details about the knowledge construction, in particular, on how the mobility profiles can be built from obfuscated traces with missing locations. The mobility and co-location reporting profiles constitute, together with the knowledge of the LPPMs used by the users (including their parameters), the adversary's *background knowledge* $\mathcal{K} = \{p_u(\cdot,\cdot)\}_{u\in\mathcal{U}}, \{f_u(\cdot)\}_{u\in\mathcal{U}}, \{g_{u,v}(\cdot,\cdot)\}_{u,v\in\mathcal{U}}$.

The output of a localization attack that targets user $u$ at time $t$, is a *posterior probability distribution* over the set $\mathcal{R}$.

$$h_t^u(r) \triangleq \Pr\left(a_u(t) = r \mid \{\mathbf{o}(t)\}_{t=1..T}, C, \mathcal{K}\right) \quad . \tag{2}$$

### 2.4 Location-Privacy Metric

The location privacy $\mathrm{LP}_u(t)$ of user $u$ at time $t$, with respect to a given adversary, is captured by the expected error of the adversary when performing a localization attack [4]. Given the output $h_t^u(\cdot)$ of the localization attack, the location privacy writes

$$\mathrm{LP}_u(t) \triangleq \sum_{r\in\mathcal{R}} h_t^u(r) \cdot d(r, a_u(t)) \ , \tag{3}$$

where $d(\cdot,\cdot)$ denotes a distance function on the set $\mathcal{R}$ of regions, typically the Haversine distance between the centers of the two regions.

## 3 Optimal Localization Attack

Without co-location information (as in [4]) and under the assumptions described in the previous section, the localization problem translates to solving an HMM inference problem, for which the *forward-backward* algorithm is a known solution. Essentially, the forward-backward algorithm defines forward and backward variables that take into account the observations before and after time $t$, respectively. The forward variable is the joint probability of location of user at time $t$ and all the observations up to, and including, time $t$. The backward variable is the conditional probability of all observations after time $t$, given the actual location of user at that time instant. Then, the posterior probability distribution of the possible locations for the targeted user is obtained by combining (*i.e.,* multiplying and normalizing) the forward and backward variables. With co-location information, the locations of the users are not mutually independent: as soon as two users are co-located at some point in time $t$, their locations, before and after time $t$, become dependent. Actually, the fact that two users meet a same third user (even if they meet her at different time instants) suffices to create some dependencies between their locations; this means that, to perform the localization attack on a user, the adversary must

[2] Note that even if pseudonyms are used, the identity of the users can be inferred by using their social network [6] or their locations [4]. We make this assumption because our main target scenario is users *posting* information attached to their real identities on social networks.

[3] This attack is somewhat similar to correlation attacks on continuous location-based queries with cloaking as presented in [7].

take into account the locations (*i.e.*, the obfuscated location information and the co-location information) of all the users who are connected to $u$ by a *chain* of co-location (*i.e.*, the connected component of $u$ in the co-location graph). Formally speaking, this means that the adversary cannot rely only on the *marginal* distributions of the users' location; instead he must consider the *joint* distributions. In other words, co-locations turn $N$ disjoint inference problems (*i.e.*, HMM problems solved by the forward-backward algorithm) into a joint inference problem.

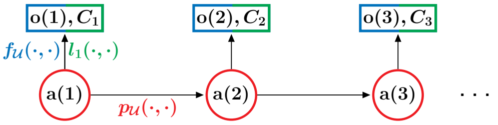


Fig. 2. Sample HMM for $T = 3$ time instants. *States* are represented by red circles, and *observations* by blue-green rectangles. State transition probabilities are specified by the joint user mobility profiles $p_{\mathcal{U}}$ and output probabilities are specified by a combination of $f_{\mathcal{U}}$ (for individual observations) and $l_t$ (for co-location observations).

To solve the localization problem, we consider the users jointly and show that it translates to an HMM problem as depicted in Figure 2. Note that more advanced learning techniques, such as neural networks, could also be used. We solve this problem by using the forward-backward algorithm [8], [9]. For a set $\mathcal{U}$ of users and time $t$, we define the following forward and backward variables:

$$\begin{aligned}\alpha_t^{\mathcal{U}}(\mathbf{r}) &\triangleq \Pr\left(\mathbf{o}(1)\dots\mathbf{o}(t), C_1\dots C_t, \mathbf{a}(t)=\mathbf{r}\,|\,\mathcal{K}\right)\\ \beta_t^{\mathcal{U}}(\mathbf{r}) &\triangleq \Pr\left(\mathbf{o}(t+1)\dots\mathbf{o}(T), C_{t+1}\dots C_T|\mathbf{a}(t)=\mathbf{r},\mathcal{K}\right) \quad (4)\end{aligned}$$

where $\mathbf{r}$ denotes a vector of size $N$, *i.e.*, $\mathbf{r} \in \mathcal{R}^N$, and represents the actual locations of all users at a single time instant. These variables can be defined recursively (over $t$) and, unlike in the case where no co-location observations are available, their expressions involve the co-location information. More specifically, we prove (in Appendix A) that for all $\mathbf{r} \in \mathcal{R}^N$, we have

$$\alpha_t^{\mathcal{U}}(\mathbf{r}) = \begin{cases} \pi_{\mathcal{U}}(\mathbf{r}) & \text{if } t = 0\\ l_t(\mathbf{r}, C)\cdot f_{\mathcal{U}}\left(\mathbf{r},\mathbf{o}(t)\right)\cdot & \\ \qquad \displaystyle\sum_{\boldsymbol{\rho}\in\mathcal{R}^N} \alpha_{t-1}^{\mathcal{U}}(\boldsymbol{\rho})\cdot p_{\mathcal{U}}\left(\boldsymbol{\rho},\mathbf{r}\right) & \text{if } t > 0\end{cases} \quad (5)$$

and

$$\beta_t^{\mathcal{U}}(\mathbf{r}) = \begin{cases} \displaystyle\sum_{\boldsymbol{\rho}\in\mathcal{R}^N} l_{t+1}(\boldsymbol{\rho}, C)\cdot\beta_{t+1}^{\mathcal{U}}(\boldsymbol{\rho})\cdot & \\ \qquad p_{\mathcal{U}}\left(\mathbf{r},\boldsymbol{\rho}\right)\cdot f_{\mathcal{U}}\left(\boldsymbol{\rho},\mathbf{o}(t+1)\right) & \text{if } t < T\\ 1 & \text{if } t = T\end{cases} \quad (6)$$

where $\mathbf{r} = (r_1,\dots,r_N) \in \mathcal{R}^N$, $\boldsymbol{\rho} = (\rho_1,\dots,\rho_N) \in \mathcal{R}^N$, $\mathbf{r}' = (r'_1,\dots,r'_N) \in \mathcal{R}'^N$, and

$$\pi_{\mathcal{U}}(\mathbf{r}) = \prod_{i=1}^{N}\pi_{u_i}(r_i),$$

$$f_{\mathcal{U}}(\mathbf{r},\mathbf{r}') = \prod_{i=1}^{N} f_{u_i}\left(r_i, r'_i\right),$$

$$p_{\mathcal{U}}\left(\boldsymbol{\rho},\mathbf{r}\right) = \prod_{i=1}^{N} p_{u_i}(\rho_i, r_i).$$

$l_t(\cdot,\cdot)$ denotes the joint probability that the users report the set of co-locations observed at time $t$, when the configuration of their actual locations at $t$ is given. That is, formally,

$$\begin{aligned}l_t(\mathbf{r}, C) &\triangleq \Pr\left(C_t|\mathbf{a}(t)=\mathbf{r}\right)\\ &= \prod_{u_i,u_j\in\mathcal{U}}\begin{cases} g_{u_i,u_j}(r_i,r_j) & \text{if } (u_i \leftrightarrow_t u_j) \in C_t\\ 1 - g_{u_i,u_j}(r_i,r_j) & \text{otherwise}\end{cases}\end{aligned} \quad (7)$$

More specifically, this is a likelihood function that captures the probability that precisely the co-locations in $C_t$ are reported, taking into account the individual co-location reporting function for every pair of users. As we assumed that co-locations are reported independently of one another, this likelihood can be expressed as a product of individual co-location reporting functions for all pairs of users.

The intuition behind Equation (5) is that the forward variable at time $t$ can be expressed recursively, with respect to time, by combining, for all possible locations of the users at time $t-1$: (1) the joint probability that the users were at location $\boldsymbol{\rho}$ at time $t-1$ and reported the obfuscated locations and co-locations observed by the adversary up to time $t-1$ (this is captured by $\alpha_{t-1}^{\mathcal{U}}$), (2) the joint probability that the users move from the locations $\boldsymbol{\rho}$ to the locations $\mathbf{r}$ (this is captured by $p_{\mathcal{U}}$), (3) the joint probability that the users obfuscate their locations $\mathbf{r}$ to those observed by the adversary $\mathbf{o}(t)$ (this is captured by $f_{\mathcal{U}}$) and (4) the joint probability that the users report co-locations $C_t$ observed by the adversary, assuming their locations $\mathbf{r}$ (this is captured by $l_t(\mathbf{r}, C)$). Because users obfuscate their locations independently from each other, the joint obfuscation probability is the product of the individual obfuscation probabilities (hence the expression of $f_{\mathcal{U}}$). The same applies to $p_{\mathcal{U}}$ and $l_t(\mathbf{r}, C)$. A similar line of reasoning applies to Equation (6).

The function $l_t(\cdot,\cdot)$ captures the likelihood of observing a set of co-location information (or not) given the actual users' locations. Schematically speaking (with a deterministic vision where only real co-locations are reported, for the sake of clarity), the set of possible locations for a user $u_i$ (at time $t$), co-located with a user $u_j$, consists of the locations that can be obfuscated into the location reported by $u_i$ at time $t$ and that can be reached (according to $u_i$'s mobility profile) from a possible location of $u_i$ at time $t-1$ **and** that can be obfuscated into the location reported by $u_j$ at time $t$ **and** that can be reached (according to $u_j$'s mobility profile) from a possible location of $u_j$ at time $t-1$.

Finally, the posterior probability distribution of the users' locations can be computed based on the forward and backward variables, by using the following formula, for $u_i \in \mathcal{U}$ and at time $t$:

$$\begin{aligned}h_t^{u_i}(r) &= \Pr\left(a_{u_i}(t) = r\,|\,\{\mathbf{o}(t)\}_{t=1..T}, C, \mathcal{K}\right)\\ &= \frac{\displaystyle\sum_{\mathbf{r}\in\mathcal{R}^N\,|\,r_i=r} \alpha_t^{\mathcal{U}}(\mathbf{r})\cdot\beta_t^{\mathcal{U}}(\mathbf{r})}{\displaystyle\sum_{\mathbf{r}\in\mathcal{R}^N}\alpha_t^{\mathcal{U}}(\mathbf{r})\cdot\beta_t^{\mathcal{U}}(\mathbf{r})}\end{aligned} \quad (8)$$

In short, the probability that the users are at given locations at time $t$ is computed based on all the observations before and at time $t$ ($\alpha_t$) and the observations after time $t$ ($\beta_t$). The denominator is a normalization factor.

We now evaluate the complexity of the joint localization attack. The first observation is that the size of the state space (*i.e.*, the locations of all users) is $M^N$. To attack a user at time $t$, the adversary needs to compute the values of $\alpha$ *up to* time $t$ and the values of beta *down to* time $t$ (using dynamic programming for optimal performance). At each time instant, the adversary needs to compute the values of these two variables for all possible values of their inputs $\mathbf{r} \in \mathcal{R}^N$ (there are $M^N$ possible values for $\mathbf{r}$). The computation of each of these values requires summing over the $M^N$ possible locations $\boldsymbol{\rho}$ at time $t-1$; for each of the possible locations, the computation of one element of the sum takes $\Theta(N^2)$ operations (the complexity of the computation of $l$ dominates for the computation of $\beta$). Therefore, the computation of the forward and backward variables, at all time instants, for all possible values of the localizations is $\Theta(N^2TM^{2N})$ operations. Note that the complexity is the same whether the adversary attacks one or all the users at one or all time instants. In fact, the adversary can precompute the $h_t^u$ for all $u$ and all $t$ with a complexity that is dominated by that of the computations of the forward and backward variables. In summary, the complexity of the localization attack on one or all of the users in $\mathcal{U}$ is

$$c_{\text{opt}}(N,T,M) = \Theta(N^2TM^{2N}) \ . \tag{9}$$

The complexity of the optimal localization attack is prohibitively high and prevents its use for the entire set of users of a mobile social network; the optimal localization attack is tractable only for small values of $N$, *i.e.*, 2-3. In the next section, we propose low-complexity alternatives for performing low-complexity approximate localization attacks.

# 4 Approximate Localization Attack

We propose two low-complexity alternatives for performing approximate localization attacks. Essentially, the first carefully selects a small set of users to consider when attacking a target user and performs an optimal joint localization attack on this small set of users (*i.e.*, considering only the co-locations between these users). The intuition behind this heuristic is that the locations of a user are significantly correlated with those of only a limited number of users (*e.g.*, a few co-workers during work hours, and her family and close friends the rest of the time). The second alternative makes use of all available location and co-location information (from all users) but only performs an approximate inference attack to localize users. We formulate the localization problem as a Bayesian network and apply a well-known inference algorithm, namely belief propagation.

## 4.1 Limited User-Set Heuristic

As discussed in Section 3, the optimal localization attack can be efficiently performed only on small sets of users. This is because the location of a target user $u$ depends on locations of *all* other users that are connected to $u$ in the co-location graph (where there is an edge between two users $u$ and $v$ if $u' \leftrightarrow_t v$ for some time $t$). The rationale of our first approximation is to limit the number of users, to whom the target user's location depends on, and to consider only those that have a high location correlation with $u$. Concretely, we choose the user(s) that have the largest number of reported co-locations with the targeted user, and we perform an optimal localization attack on the resulting set of users. We call these users the *co-targets* of the targeted user. Depending on his computational power, the adversary can choose one or two such users (*i.e.*, $N=2$ or $N=3$) to attack the target with. The co-targets of a user $u$ are chosen as follows:

$$\text{co-target}_1(u) \triangleq \underset{v \in \mathcal{U}\setminus\{u\}}{\text{argmax}} \left|\{t \in \{1,\dots,T\} \mid u \leftrightarrow_t v\}\right| \tag{10}$$

$$\text{co-target}_2(u) \triangleq \underset{v \in \mathcal{U}\setminus\{u,u'\}}{\text{argmax}} \Big[ \left|\{t \in \{1,\dots,T\} \mid u \leftrightarrow_t v\}\right| + \left|\{t \in \{1,\dots,T\} \mid u' \leftrightarrow_t v\}\right| \Big] \tag{11}$$

where $u' = \text{co-target}_1(u)$ and $|\cdot|$ denotes the cardinality of the set. More specifically, the first co-target of a user $u$ is the user with whom $u$ has the more reported co-locations during the time interval considered for the localization attack. The second co-target of $u$ is chosen so as to maximize the number of co-locations with $u$ **plus** the number of co-locations with $u$'s first co-target. Note that the set of considered users can be different for every targeted user; in particular $v = \text{co-target}_1(u) \not\Longrightarrow u = \text{co-target}_1(v)$. In practice, an adversary could also take into account the function $g$ when choosing $u$'s co-targets, to better reflect the trustworthiness of the reported co-locations. The complexity of this heuristic is $\Theta(TM^4)$ for $N=2$ and $\Theta(TM^6)$ for $N=3$ (obtained by replacing $N$ by its value in the generic expression (9) of the complexity of the optimal attack).

## 4.2 Bayesian Networks-Based Approximation

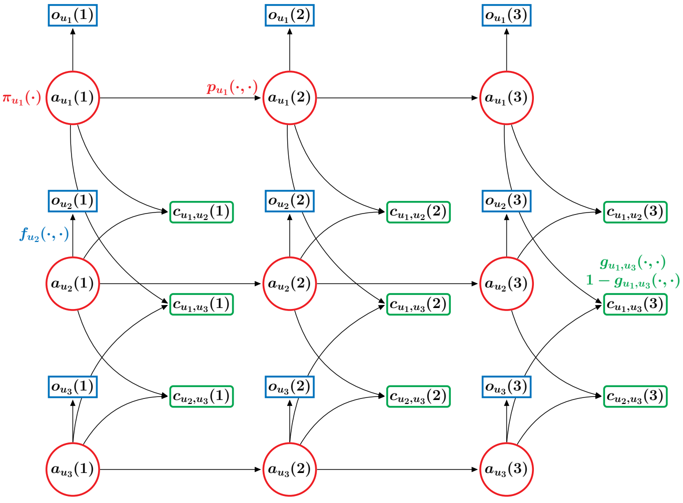


Fig. 3. Sample Bayesian network for $N=3$ users and $T=3$ time instants. *Actual location nodes* are represented by red circles, *observed location nodes* by blue rectangles and *observed co-location nodes* by green rectangles with rounded corners. Probabilistic dependencies are specified by edges and conditional probability distributions (CPD), e.g., a co-location observation depends only on the actual locations of the two involved users and the probabilistic dependency is captured by $g$.

We propose using approximation algorithms on Bayesian networks, as a low-complexity alternative solution to the localization problem. A Bayesian network is a graphical model that encodes the probabilistic dependencies between different random variables of interest [9], [10]. More specifically, a Bayesian network is a directed acyclic graph in which nodes represent random variables and the edges model conditional dependence between the variables corresponding to the nodes they connect. In addition to its (graph) structure, a Bayesian network is also specified by its parameters: Each node has an associated conditional probability distribution (CPD), which specifies the probability that the corresponding variable will take a certain value, given a combination of values of the variables associated with its predecessor nodes. Modeling our problem as a Bayesian network enables us to exploit existing approximate inference algorithms, such as the belief propagation (BP) algorithm [10], [11] (which we use in the evaluation). BP is an algorithm that converges to the optimal solution by iteratively updating the posterior of a random variable, based on that of its neighbors and on its CPD, using values of the observed variables. For Bayesian networks that do not contain undirected loops, which is *not* the case of our model, the BP algorithm converges to the optimal solution in only one iteration. Because of its iterative aspect, it balances (through the number of iterations) execution time and accuracy. Moreover, by running the BP-based solution, the adversary can obtain coarse-grained estimates of the users' locations after a few iterations and update them with more precise estimates as BP progresses. The heuristic presented in the previous sub-section makes the most out of a subset of the available information (i.e., optimal inference on the data of the target user and her co-targets), whereas the BP-based solution only approximates the optimal solution but exploits all the available information (approximate inference on the data of all the users).

We build a Bayesian network as illustrated in Figure 3 (for $N = 3$): For any user $u$ and any time instant $t$, a node is associated with the variable $a_u(t)$ and another with the variable $o_u(t)$. To represent the fact that the observed location depends only on her actual location at that time, an edge connects the corresponding nodes and the corresponding CPD is $f_u$. Additionally, an edge connects the node corresponding to a user $u$'s actual location at time $t$ to her actual location node at time $t+1$, with its CPD determined by her mobility profile $p_u$ (following from the Markov assumption). For any pair $u, v$ of users and any time instant, an observed co-location node is associated with variable $c_{u,v}(t)$, with its CPD specified by $g_{u,v}$ (it depends on the actual location of the two users involved). Our Bayesian network consists of $T \cdot N$ actual/observed location nodes and $T \cdot N(N-1)/2$ observed co-location nodes.[4] Location nodes have one incoming edge, and co-location nodes have two. Consequently, the complexity for one iteration of the belief propagation algorithm is $O(N^2 \cdot T \cdot M^2)$.

We compare the approximate localization attack to the optimal localization attack, and we measure its accuracy by the average Hellinger and statistical distance between their output region distributions (details in Appendix E).

# 5 COUNTERMEASURES

So far, we presented and analyzed a localization attack that exploits co-location information. In this section, we propose two countermeasures that mitigate the (negative) effect of co-locations on the users' location privacy. These countermeasures apply to the case where users explicitly report their co-locations, typically on a social network. For co-location information leaked by the underlying technologies, such as IP addresses and Bluetooth and Wi-Fi scans of neighboring devices, technology-dependent techniques should be used. For instance, a user can hide her IP address from the service provider by using a proxy, a VPN or a peer-to-peer anonymization network such as Tor. Note that countermeasures are not limited to those presented in this section. Altering the individual LPPM settings (the value of $\lambda$, $\mu$ and $\nu$, using obfuscation or cloaking) would also reduce, to some extent, the privacy risk. Unfortunately there is not much else a user can do to protect herself, other than hide or generalize co-location information or prevent it from being inferred. In practice, this would translate to hiding IP addresses, disabling Bluetooth, or blurring faces in pictures posted on online social networks, as proposed in [13]. Simply put, the proposed countermeasures operate as follows: The first consists in making co-located users report the same (obfuscated) location and the second consists in generalizing time and/or user information in the reported co-locations.

## 5.1 Coordinated LPPMs

In order to make the inference attacks we described in previous sections less effective, we propose a simple countermeasure: user *coordination*. This means that if users report being co-located at some time instant and also want to report obfuscated individual check-ins, they should coordinate them (*i.e.*, report the same obfuscated location). Such a mechanism requires collaboration between users, which can be challenging to achieve in practice. A possible solution, in the case of explicitly reported co-locations, is that a user who posts a co-location information embeds her obfuscated location so that all the co-located users report the same obfuscated location (if they do report their locations). Collaboration could also be achieved by means of short-range ad-hoc communication technologies such as Wi-Fi Direct or Bluetooth, as the co-located users are physically close. We emphasize that this does not mean that co-located users have to also report individual check-ins, but rather that if they want to report individual check-ins, they agree to make them the same. We argue this would bring no detriment to users' utility of individual check-ins, as the obfuscation mechanism selects a *random* neighboring location to the actual location, which users have no control or preference over. Intuitively, reporting single co-locations in a coordinated fashion should give an adversary less information, because it maximizes the set of possible locations co-located users could be in. As described in Figure 1, based on individual check-ins of co-located users, an adversary can infer that both users should be located in the intersection of possible locations of each of the co-located users. With coordination, the possible locations of users are the same, thus maximizing their intersection. Note that this countermeasure has an effect only if both users use obfuscation.

[4] Note that when the probability of two users reporting a co-location between them is null (e.g., non-friend users in a social network), the corresponding nodes can be removed. As suggested by Dunbar's number [12], a user has a limited number of friends. Therefore, in many contexts, the number of co-location nodes grows linearly with $N$.

### 5.2 Generalization of Co-locations

We propose another countermeasure for reducing the effectiveness of inference attacks that make use of co-location information. In the case of single location observations, a recommended privacy-protection technique is obfuscation by generalization (i.e., report a large area that contains the user's actual location). Similarly, we propose that users generalize co-location information, in a coarse-grained fashion; specifically, this implies generalizing the time component of a co-location, and/or the co-located user(s) component. Generalizing the time component in a co-location information means reporting a time range instead of the exact time (*e.g.,* use "morning" instead of "10am"). Generalizing the user component means excluding the names of the friends a user is with and only reporting the *number* of friends (*e.g.,* instead of reporting being with her friend Alice, a user would just report being with *a friend*). More generally, in the case where a user $u$ is co-located with $k$ friends $u'_1, u'_2, \ldots, u'_k$ he would no longer report $k$ pairwise co-locations with each of them ($u \leftrightarrow_t u'_1, \ldots, u \leftrightarrow_t u'_k$), but instead report one *generalized co-location* $u \leftrightarrow_t k$ friends. The user component of the co-location could also contain social ties information such as "with two colleagues", or "with some friends".

We analyze in more depth the case of generalizing the co-located user(s) component of a co-location. Intuitively, if this mechanism is employed by the users, it is harder for the adversary to exploit a co-location information because he has to explore all possible combinations of real users a user is with and assign a likelihood to each of them. This leads to $\binom{N_u^{\text{colleagues}}}{k}$ possible choices to explore for the generalized co-location "with $k$ colleagues", where $N_u^{\text{colleagues}}$ is the number of user $u$'s colleagues in the social networks, and $2^{N_u^{\text{friends}}+N_u^{\text{family}}}$ choices for the generalized co-location "with some friends and family". More specifically, the joint variables $\alpha^{\mathcal{U}}_{\cdot}(\cdot)$ and $\beta^{\mathcal{U}}_{\cdot}(\cdot)$ (Equation (5) and Equation (6)) would include a summation in the computation of the likelihood of observing the obfuscated co-locations ($l_{\cdot}(\cdot, C)$) for *all* possible instantiations of *all* reported co-locations at time $t$ (for $\alpha$) or $t+1$ (for $\beta$) by *all* users. This would drastically increase the complexity of the optimal inference attack. Note that generalizing the user component of the co-locations would also drastically increase the complexity of the BP-based solution; the current Bayesian network (Figure 3) could not be used anymore. We will investigate this as part of future work. In summary, this countermeasure protects the users' privacy by making the inference prohibitively computationally expensive for the adversary.

Obfuscating the time component of co-locations would also lead to a drastic increase in complexity because the adversary would have to consider all combinations of exact time instances when users are co-located (which makes the computation of the joint $\alpha, \beta$ variables nontrivial). Naturally, obfuscating both components of co-location information would result in the greatest complexity increase. We leave the design and in-depth analysis of inference algorithms when a combination of the proposed countermeasures is employed by users to future work. We intend to analytically evaluate the inference complexity and empirically evaluate the users' *privacy gain* and potential *utility loss* in different scenarios of employed countermeasures.

## 6 Differential-Privacy Perspective

In this section, we complement our inferential approach to privacy quantification, presented in the previous sections, with a brief analysis of the effect of co-locations on users' location privacy from a differential-privacy perspective. In the geo-indistinguishability framework [14], [15] (i.e., the application of differential privacy to geo-location), each observation has a privacy cost that depends on the level of noise added by the mechanism used (typically drawn from a planar Laplace distribution). For instance, in order to guarantee $\varepsilon$-differential privacy, one must introduce noise with an amplitude such that the expected distance between the actual location and the reported location is proportional to $1/\varepsilon$. Consider the case of a single time instant. If two co-located users each report one obfuscated version of their actual locations, the adversary has access to two observations of the same variable, i.e., the users' common location. Following the composability property of differential privacy, this means that, to guarantee $\varepsilon$-differential privacy for the users' location, each individual reported obfuscated location should satisfy $(\varepsilon/2)$–differential privacy (unless the two users agree on reporting the same obfuscated location, as discussed in Section 5 dedicated to countermeasures). This means that the expected distance between the users' actual locations and the obfuscated locations they report is doubled, thus causing a substantial utility loss. This reasoning can be generalized to an arbitrary number of co-located users: At every time instant, the level of noise a user must introduce (and thus the utility loss she faces), in order to retain the same privacy level in the presence of co-locations, is proportional to the number of co-located users.

A more complex analysis of the effect of co-locations, from a differential-privacy perspective, could be carried out by leveraging on the Putterfish framework [16] (or more recently [17]) which enables taking into account the correlation between entries in a differential-privacy analysis. We leave this research direction to future work.

## 7 Experimental Evaluation

Using a dataset of mobility traces, we evaluate the effect of co-locations on users' privacy, with respect to the various localization attacks presented in the previous sections.

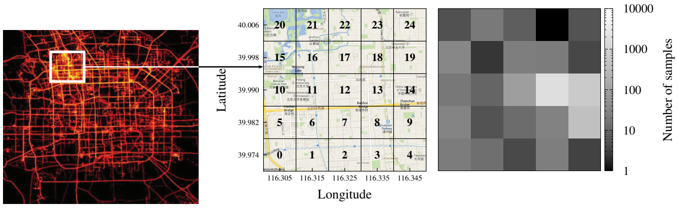


Fig. 4. Illustration of the dataset used in the evaluation. Most traces are located in the region of Beijing (left); we focus on a small active area that corresponds to the campus of Tsinghua University and we partition it by using a $5 \times 5$ square grid (middle). The heat-map (right) shows the number of samples in each region (logscale), for the users of interest.

### 7.1 Dataset, Methodology and Experimental Setup

The dataset was collected by Microsoft Research Asia, in the framework of the GeoLife project [18]. It comprises the GPS traces (*i.e.,* sequences of time-stamped latitude-longitude couples, sampled at a rate of one point every 1-5 seconds) of 182 users, collected over a period of over three years. The GPS traces are scattered all over the world; but most of them are located in the region of Beijing, China. We processed the data as follows, in order to fit in our formalism.

**Space discretization.** We select the area of $\sim 4.4$ km × $4.4$ km, within Beijing, that contains the largest number of GPS samples, and we filter out GPS samples that are outside of this area. This geographic area corresponds to the campus of Tsinghua University (longitude ranging from 116.3 to 116.35 and latitude ranging from 39.97 to 40.01, see Figure 4). We partition the selected area into 25 regions by using a 5×5 square grid. The GPS coordinates of each sample are translated into the region (*i.e.,* the grid cell) they fall into.

**Time discretization.** We split the continuous time interval into one-hour time sub-intervals that correspond to time instants in our formalism. For each time sub-interval $t$ and for each user $u$, we set the user's actual location in that time interval (*i.e.,* $a_u(t)$) to the region corresponding to the sample that is the closest to the midpoint of the considered time sub-interval. If a user's trace does not contain any samples in a given time sub-interval, the user's actual location is set to a dummy region $r_\perp$, leaving us with partial user traces.

**Co-location generation.** As the dataset does not contain explicit co-location information reported by the users, we use synthetic co-locations that we generate as follows: At each time instant, we generate a co-location between two users according to the probabilistic co-location reporting function $g_{\cdot,\cdot}(\cdot,\cdot)$, based on their discretized actual locations (if they are different from $r_\perp$). We consider a special case of the co-location reporting function (Equation (1)) as follows:

$$g_{\cdot,\cdot}(r_u, r_v) = \begin{cases} \nu & \text{if } r_u = r_v \\ \mu & \text{if } r_u \neq r_v \end{cases} \tag{12}$$

As stated in the model, the adversary is assumed to know the values of $\mu$ and $\nu$. Intuitively, $\mu$ represents the probability a *fake* co-location is reported, and $\nu$ represents the probability a *true* co-location is reported. This model assumes that for any user, reporting a co-location does not depend on the actual location where she and her friend are and that the user chooses to report their co-location with a fixed probability. In order to simplify the evaluation, we assume that the co-location reporting function is the same among any pair of users, as in the case of a Bluetooth scenario. We could relax this assumption and make $\nu$ and $\mu$ functions of the particular pair of users; for example, if a social graph of relationships between users were available, we could consider $\nu, \mu > 0$ only for pairs of users for which a social relationship exists, and 0 for all other user pairs, as users typically report co-locations on social networks only with their friends. Regarding the values of $\nu$ and $\mu$, several cases can also be considered: $\nu = 1$ and $\mu = 0$ would correspond, for example, to an ideal Bluetooth scenario, in which devices discover each other automatically and report co-locations with all neighboring devices; $\nu < 1$ and $\mu = 0$, could also correspond to a Bluetooth scenario, where co-locations are reported with only some of the neighboring devices. In our evaluation, we will consider both cases.

For each user, we compute the number of *real* co-locations[5] she has with every other user in the dataset, across the full user traces. We keep only the users for which there exists another user with whom they have at least 200 co-locations. For these users, we consider their *common* time interval (*i.e.,* the longest time interval during which all these users have at least one sample); we obtained an interval of ~6000 hours. Within this interval, we sample 10 short traces of 300 continuous hours such that (1) all users have at least 10% of valid samples (*i.e.,* , different from $r_\perp$) and (2) all users have at least 20 co-locations with their co-target$_1$ (as defined in Eq. (11)). This leaves us with a total of 5 users.

**User mobility profiles construction.** We build the mobility profiles $\{p_u\}_{u \in \mathcal{U}}$ of the users based on their entire discretized traces by counting the transitions from any region to any region (in $\mathcal{R}$) in one time instant.

**Obfuscation.** We consider that users report a single (or none), potentially obfuscated, location at each time instant. This means that the set $\mathcal{R}'$ in which the obfuscated location $o_u(\cdot)$ takes values is $\mathcal{R} \cup \{r_\perp\}$. We consider, for each user $u$, that two location-privacy protection mechanisms are used together: First, the location is hidden (*i.e.,* obfuscated to $r_\perp$) with a probability $\lambda_u$ and then, if the location has not been hidden, it is replaced by a region (chosen uniformly at random) at a distance of at most $d_u$ from the user's actual discretized location (*i.e.,* a region). If the actual location of a user is not known (*i.e.,* set to $r_\perp$), the LPPM returns $r_\perp$ with probability 1. In our evaluation, we vary $\lambda_u$ from 0 to 1 and we set $d_u$ to the size of one grid cell; this means that, if it is not hidden, a user's location is obfuscated either to its actual value (with probability 0.2) or to one of the four adjacent regions (*e.g.,* 2, 6, 8 and 12 for region 7 in Figure 4), each with probability 0.2.

**Privacy Evaluation.** We evaluate the location privacy of the users based on the metric defined in (3). For each user and for each short trace, we generate 20 random obfuscated traces (remember that obfuscation is a random process), and we perform a localization attack on each of them. We compute the average location privacy of each user across the different obfuscated traces and across the different time instants. Time instants for which the location of a user is not known (*i.e.,* set to $r_\perp$) are not taken into account in the computation of their average over time.

**Limitations.** Unfortunately, we could not obtain real datasets from online social networks containing both (coarse-grained) location and co-location data. Due to the synthetic nature of the reported location and co-location information in our data source, our experimental setup does not perfectly reflect a real usage case. Deciding whether to report locations and co-locations is a complicated process, which involves many factors, such as the users' contexts, their privacy preferences and the shared information itself. Therefore, the results presented in this section should be taken with a pinch of salt as they cannot directly be interpreted as the magnitude of the threat in real life. Yet, we

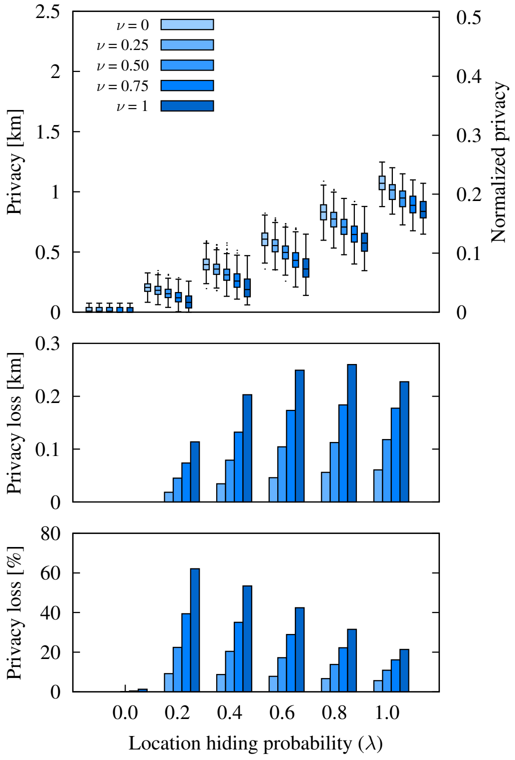
[5] Note that by real co-locations, we mean that the users are at the same location (i.e., their actual locations at a given time instant are the same), regardless of the fact that the co-location is reported or not.

believe that they are significant enough for understanding the effect of co-locations on location privacy, the sources of privacy loss, and the relative performance of the proposed heuristics. Also, the number of users considered in most of our evaluations (*i.e.,* 5), as well as the active area considered are relatively small. In order to overcome the aforementioned shortcomings, we intend to collect a large-scale dataset from an existing social network. We also intend to run experiments on large grids (*i.e.,* larger than 5×5).

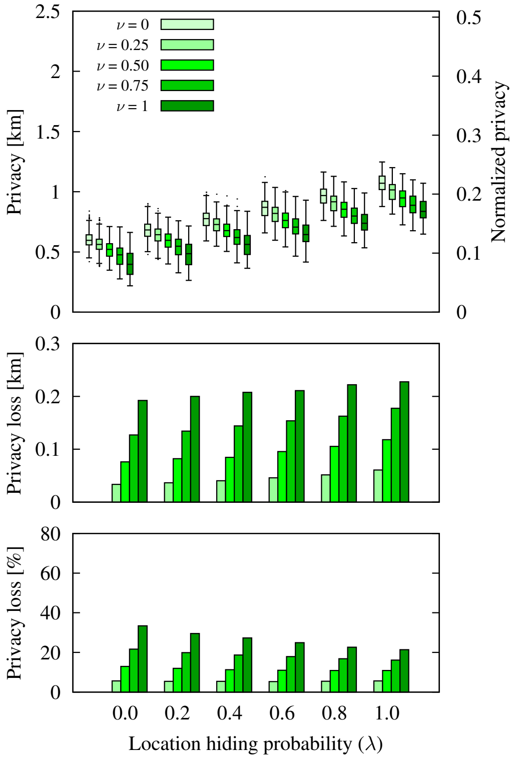
### 7.2 Experimental Results

We experimentally evaluate the algorithms, presented in Section 4, in different scenarios, with different settings. For the solution based on belief propagation, we relied on the implementation provided in the Bayes Net Toolbox for Matlab (https://code.google.com/p/bnt/); for the optimal inference algorithm, we used our own JAVA implementation. The purpose of our evaluation is to assess the raw performance of our algorithms, but also to compare their results. In addition, we analyze the effect of the different parameters of the model (including the individual LPPM settings of the users and the *differences* between the individual LPPM settings of the users) and of the set of co-locations considered in the localization attack.

**Effects of true co-locations and LPPM settings.** We begin our evaluation by analyzing the effect of (1) the amount of reported true co-locations and (2) the LPPM settings (*i.e.,* w/ or w/o obfuscation and the location hiding probability $\lambda$, assumed to be the same across users) in the case of two users, *i.e.,* the target user and her first co-target are considered jointly in an optimal localization attack, namely the limited user set approximation with $N = 2$. For this evaluation, we consider the case where no fake co-locations are reported. The results are depicted in Figure 5. The top sub-figure shows the case where no obfuscation is used (*i.e.,* the users disclose their *actual* locations with probability $1 - \lambda$ and hide them otherwise), and the bottom sub-figure shows the case where obfuscation is used (*i.e.,* the users disclose their *obfuscated* locations, specifically a region chosen uniformly at random among the actual location and the four immediate neighboring regions, with probability $1 - \lambda$ and hide them otherwise). The top graphs show a box-plot representation (*i.e.,* first quartile, median, third quartile and outliers) of the users' location privacy expressed in terms of the expected error of the adversary, in kilometers (left axis) and in proportion of the size of the considered geographic area (right axis). For each couple of values $(\lambda, \nu)$, we draw one box-plot to aggregate the data-points obtained for all users and for all the 20 randomly generated obfuscated versions of each of the considered actual traces. Note that without obfuscation, the case $\lambda = 0$ leads to zero privacy as users *always* disclose their *actual* locations. It can be observed that the proportion of reported true co-locations consistently decreases the location privacy of the users. To quantify this decrease, we plot (middle and bottom graphs) the privacy loss caused by the use of co-location information, with respect to the case where no true co-locations are reported, *i.e.,* $\nu = 0$. We show both the median absolute privacy loss (in kilometers, middle graph) and the median relative privacy loss (in percentage of the privacy in the case $\nu = 0$, bottom graph). Note that the median privacy loss is equal to the median of the differences (w.r.t. the case $\nu = 0$) and **not** to the difference of the median privacy.



(a) Without obfuscation



(b) With obfuscation

Fig. 5. Privacy (top), absolute privacy loss (middle) and relative privacy loss (bottom) for the limited user set attack with $N = 2$ users, when users do not report fake co-locations ($\mu = 0$). The privacy *loss* is expressed w.r.t. the case where no co-locations are available ($\nu = 0$, $\mu = 0$); the histograms show median values.

Consider for example, the case $\lambda = 0.4$ and $\nu = 0.5$: In the case without obfuscation the median privacy loss is approximately 80m, which corresponds to a decrease of approximately 21%. The median absolute privacy loss can go up to 260m ($\lambda = 0.8$, $\nu = 1$) and the median relative privacy loss up to 62% ($\lambda = 0.2$ and $\nu = 1$). We observe the same trend, with a more modest loss, in the case where obfuscation is used. We emphasize that when there is an obfuscated location observation, the adversary has only 5 choices of cells to locate the user: the cell of her actual location and 4 neighboring cells of the actual location. Hence, an upper bound for privacy in this case is given by the inter-cell distance (0.87km). It can be observed in Figure 5(b) that when all observations are available ($\lambda = 0$), this upper bound is indeed respected.

As a complementary experiment, we also studied the effect of co-location information when users employ spatial cloaking instead of obfuscation. The same trend is apparent in this case. Detailed results are presented in Appendix B.

In the next sections, we focus on the case where users obfuscate their locations, report true co-locations with probability $\nu = 0.5$ and do not report fake co-locations ($\mu = 0$).

**Effects of the differences of individual LPPM settings.** We provide an analysis of the effect of the differences, in the users' LPPM settings, on the users' location privacy in Appendix C of the supplemental material (see also [1]).

**Comparison of the proposed low-complexity alternatives.** Here, we compare, through experimentation, the proposed inference algorithms for the localization attack, by taking into account different scenarios, as depicted in Figure 6. We assume all users use the same LPPM settings, *i.e.*, same value for $\lambda$ and disclose only their obfuscated locations. In Scenario (a), we consider, in turn, all target users in our set and perform an individual localization attack on each of them, using only their own reported locations and no co-locations. This corresponds to the baseline case $\nu = 0$, which was presented in detail in Figure 5b. We then consider the case of an adversary that exploits co-locations. We assume users report only a limited proportion of their true co-locations, with probability $\nu = 0.5$, and no fake co-locations ($\mu = 0$). Scenario (b) corresponds to the case of an adversary that, in order to attack a target user, performs an optimal joint inference attack on the target and her co-target, as described in Section 3. This scenario corresponds to the case $\nu = 0.5$ in Figure 5b. Scenarios (c) and (d) correspond to the case of an adversary that performs an optimal joint attack on the target and her **two co-targets**. We distinguish two cases, (c) – in which the co-locations between the co-targets are ignored and (d) – in which all co-locations between any of the three users are considered. We make this distinction solely to quantify the privacy loss that stems from the use of co-locations that do not directly involve the target. In practice, an adversary would always consider Scenario (d) because it takes into account more information at no extra cost. Finally we consider Scenario (e), that corresponds to an adversary that uses *all* reported co-locations but solves an *approximate* joint inference problem, as described in Section 4.2. We set the maximum number of iterations of the BP algorithm to 20.

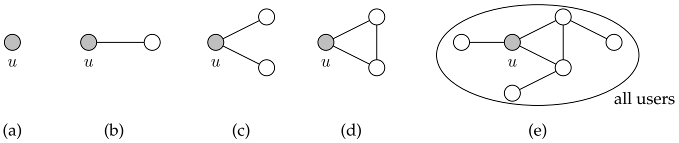


Fig. 6. Co-locations considered in the evaluation: (a) no co-locations (also referred to as *No co-target*), (b) only co-locations between the target and co-target$_1$ (heuristic, $N = 2$), (c) only co-locations between the target and co-target$_1$ and between the target and co-target$_2$ (heuristic, $N = 3$), (d) all co-locations between the target, co-target$_1$ and co-target$_2$ (heuristic, $N = 3$), (e) all co-locations (belief propagation in our proposed Bayesian network formalization depicted in Figure 3).

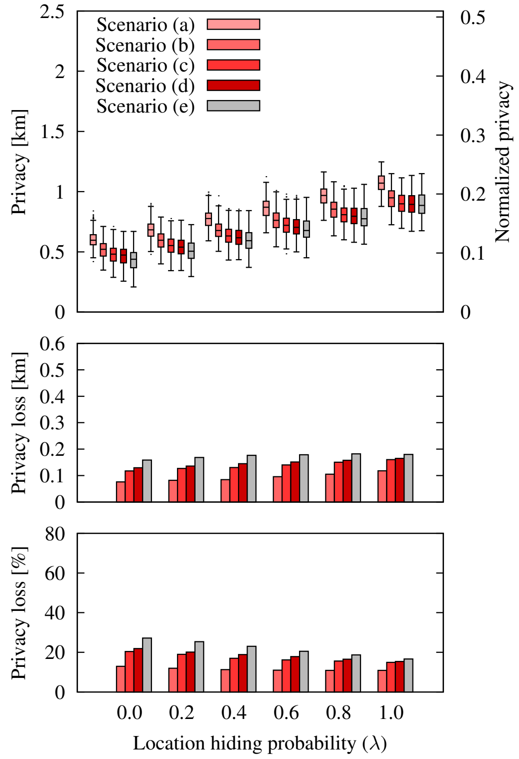


Fig. 7. Comparison of the different localization attacks for the scenarios (a)-(e) depicted in Figure 6, with obfuscation. The privacy loss (middle and bottom) is evaluated w.r.t. Scenario (a). In scenarios (b)-(e), we consider users report true co-locations with probability $\nu = 0.5$ and that they do not report fake co-locations ($\mu = 0$).

Figure 7 shows the results of our comparison. The top graph shows a box-plot representation of users' privacy, for each of scenarios (a)-(e). To quantify the different effects on the users' privacy of the set of considered co-locations and of the inference algorithm used, we show (bottom) the absolute and relative privacy loss, with respect to Scenario (a), for each of the scenarios (b)-(e). It can be observed by comparing scenarios (a)-(d) that, unsurprisingly, the users' privacy decreases with the amount of considered co-locations. The comparison between scenarios (c) and (d) shows that co-locations between the target's co-targets improve the performance of the localization attack, but not as much as co-locations that directly involve the target user (Scenario (b) and Scenario (c)). Finally, we observe that the approximation based on belief propagation (Scenario (e)), which takes into account all co-locations and the location information of all the users, outperforms the first heuristic ($N \leq 3$), at a low computational cost. In this scenario, the median absolute privacy loss can go up to 182m ($\lambda = 0.8$) and the median relative privacy loss up to 27% ($\lambda = 0$), when $\nu = 0.5$ and $\mu = 0$. We can thus conclude that, when using belief propagation instead of joint optimal inference, the loss in inference accuracy is far smaller than the gain that stems from using all of the available co-location information and the location information of all the users.

In order to assess the performance of the belief propagation algorithm, we also compared it with the optimal inference algorithm, for all scenarios (a)-(d). For each of these scenarios, we computed the Hellinger distance between the BP algorithm and the optimal inference. We obtained the following distances: 3.79E-4 for Scenario (a), 5.18E-3 for Scenario (b), 1.79E-2 for Scenario (c) and 3.31E-2 for Scenario (d). Similarly, we computed the statistical distances and obtained the following: 1.86E-4 for Scenario (a), 3.79E-3 for Scenario (b), 1.84E-2 for Scenario (c) and 3.10E-2 for Scenario (d). These very small values for both the Hellinger and statistical distance, for all scenarios, show that the BP algorithm converges in about 20 iterations. In fact, we observe that the approximation provided by the BP algorithm is already quite close to the optimal after a very small number of iterations (i.e., 2-3) which suggests that the attack can be carried out efficiently by the adversary.

To further analyze and compare the performance of the different inference algorithms, we measured their execution times in a typical setting ($\lambda = 0.2$, $\nu = 0.5$ and $\mu = 0$, for a single user) on an 8-core Intel(R) Xeon(R) CPU E3-1270 V2 @ 3.50GHz with 16GB of RAM. We obtained the following results: Scenario (a): 1.82±0.0471s; Scenario (b): 3.87±0.0498s; Scenario (d): 2,555±73.6s; Scenario (e): 2.66±0.0151s. These results demonstrate the practicality of the BP-based attack.

**Effects of users' behavior of reporting fake co-locations.** Here, we analyze the effect of reporting fake co-locations. We focus on the case of two users, a target and her first co-target, who both obfuscate their locations and use the same location hiding probability. We present the case where $\lambda = 0.2$, but we mention that we observe the same trend for all values of $\lambda$. We vary $\nu$ – the probability to report true co-locations, as well as $\mu$ – the probability to report fake co-locations. We perform a joint optimal localization attack. The results are depicted in Figure 8. The top graph shows a box-plot representation of users' privacy, and the middle and bottom graphs show the median absolute and relative privacy loss, with respect to Scenario (a) (where no co-location information is considered). We observe that when all the true co-location information between the target and her co-target is reported ($\nu = 1$), the users' privacy increases as there are more fake co-locations reported (as $\mu$ increases). However, when none of the true co-locations are reported ($\nu = 0$), we observe that the users' privacy *decreases* with the increase of available fake co-location information. In other words, an adversary can exploit the absence of a reported fake co-location at some time instant to infer that the users must, in fact, be co-located (for large values of $\mu$).[6] This is an interesting observation, that shows an adversary can learn not only from available co-location information but also from the *absence* of co-location information. Finally, in the case where only some of the true co-location information is reported ($\nu = 0.5$), we observe the largest users' privacy for values of $\mu$ which lead to a high uncertainty for the adversary (these are middle values of $\mu$). We emphasize that users' privacy in the case where $\nu = \mu = 0$ (users *never* report co-location regardless of whether they are co-located or not) is the same as that where $\nu = \mu = 1$ (users *always* report co-location regardless of whether they are co-located or not). Finally, an important observation is that regardless of the amount of available co-location information (true or fake), users' privacy is never larger than that in the case where no co-locations are considered. This means that an adversary cannot be significantly confused by misleading co-location information, so reporting such fake co-locations would not be an effective privacy protection practice.

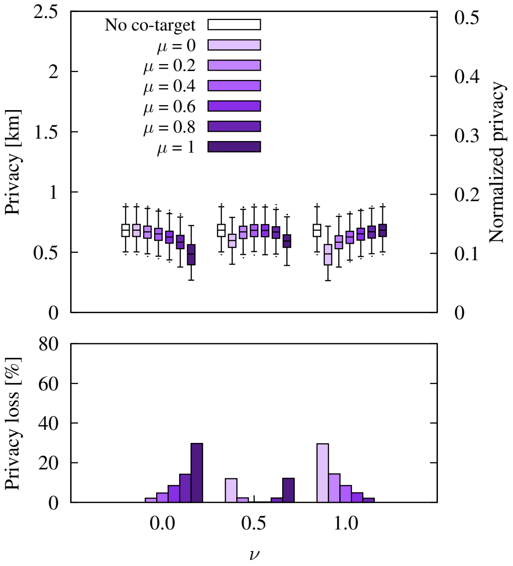


Fig. 8. Privacy (top), absolute privacy loss (middle) and relative privacy loss (bottom) for the limited user set attack with $N = 2$ users, with obfuscation, $\lambda = 0.2$ and $\nu \in \{0, 0.5, 1\}$. The privacy *loss* is expressed w.r.t. the case where no co-locations are available; the histograms show medians. We observed similar results for other values of $\lambda$ (not shown).

**Effects of user coordination.** We present the effect of using coordination for Scenario (e), where all available co-location is used. We infer the user location by using the BP algorithm for Scenario (e) and optimal inference for Scenario (a). We focus on the case where all users use obfuscation and have the same location hiding probability, $\lambda$. We assume users report true co-location information with probability $\nu = 0.5$ and no fake co-location information. We consider both the case where all users use coordination and the case where no users coordinate. We compare these with Scenario (a), where no co-location information is observed. Figure 9 shows the results of our experiment–a box plot representation of user privacy in the top graph, and median privacy loss with respect to Scenario (a) in the bottom graphs. We observe that when users coordinate, their privacy can still decrease compared to the case where no co-locations are used, but there is a *privacy gain* with respect to the case where co-locations are reported in an uncoordinated fashion. This privacy gain

[6] This is similar to the case of the entropy of a binary variable that is flipped with a given probability. The entropy increases with the flipping probability if it is lower than 0.5 and decreases beyond 0.5.

is higher, as $\lambda$ decreases. For instance, when $\lambda = 1$, users always hide their individual location and there is nothing to coordinate, so coordination has no effect on users' location privacy. However, as users report more individual check-ins ($\lambda$ decreases), the privacy gain stemming from coordination increases, with a peak for $\lambda = 0$ (where users' privacy loss is reduced by half when coordinating). We can conclude that, by coordinating their individual check-ins with their friends at times where users also report being co-located, users can limit the privacy loss caused by the co-location information.

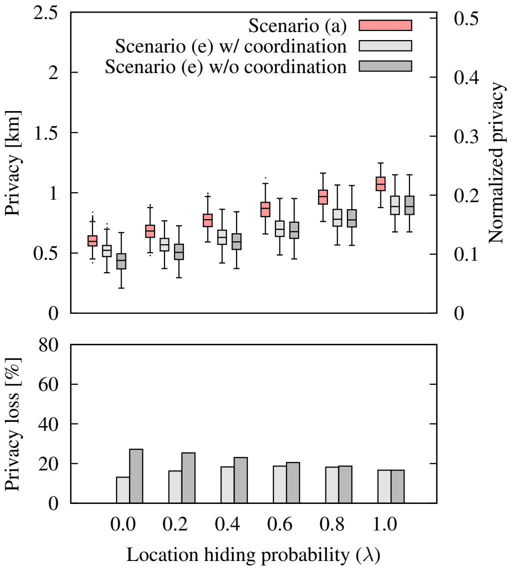


Fig. 9. Localization attack with and without coordination for scenario (e) depicted in Figure 6, with obfuscation, $\nu = 0.5$ and $\mu = 0$. The privacy loss (middle and bottom) is evaluated w.r.t. scenario (a).

**Co-location Information on a Larger Scale.** We evaluate the Bayesian network-based approximation on a set of 38 users. We compare it with the optimal individual localization attack (where no co-location information is used) and observe the same trend that co-location information further reduces location privacy (results are available in Appendix D).

## 8 Related Work

Location is identity. Even if the set of locations shared by a user is anonymized, and her true identity is hidden from the location-based service provider, the observed trajectories can be re-identified [19]–[22]. This attack is made by linking available information about users' mobility in the past with their observed traces. To protect against such attacks, many location obfuscation mechanisms have been proposed in the literature; they suggest users hide their locations at certain locations, or reduce the accuracy or granularity of their reported locations [23]–[25]. These techniques increase users' privacy by making it more difficult for an adversary to de-anonymize users and localize or track them over time. The location privacy of users in such settings can be computed using the expected error of an adversary in estimating their locations [4]. In such an inference framework, an adversary has some background knowledge on users' mobility models.

The adversary's information, however, is not limited to mobility models. With most users being members of social networks, an adversary can de-anonymize location traces by matching the graph of co-traveler users with their social network graph [26]. Co-travelers are those who have been in each others' physical proximity for a considerable number of times. Researchers have extensively studied the problem of inferring social ties between users based on their physical proximity [27], [28]. Recent revelations about NSA surveillance programs also show that this type of information is of great use for tracking and identifying individuals [29]. The dual problem, i.e., inferring location from social ties, has also been studied by the research community [30]–[32]. In [33], the authors exploit proximity information detected via Bluetooth, which is similar to co-location, to build an opportunistic ad-hoc localization algorithm by using intersection techniques similar to what we use in our attack. In [34], the authors modeled the influence of social relationships on human movement and used this to predict future locations of the users. Location privacy risks have also been studied in the context of proximity detection (*e.g.*, finding nearby friends in OSNs) [35], [36].

The correlation between different users' information also opens the door to a new type of privacy threat. Even if a user does not reveal much information about herself, her privacy can be compromised by others. In [37], the authors study how information revealed, from pictures, by a user's friends in social networks can be used to infer private information about her location. Private information about, for example, user profile and her age can also be inferred from shared information on online social networks [38], [39]. A users' home address can also be inferred from those of her Facebook friends [40]. Mobile users, connecting to location-based services from a same IP address, can also compromise the privacy of those who want to keep their location private [41]. The loss in privacy, due to other users, has also been shown in other contexts such as genomics [42], [43]. Finally, interdependent privacy risks have been studied by using game-theoretic models for predicting the optimal behavior of rational users, in the context of OSNs [44], [45] and genomics [46]. Other game-theoretic interdependence models for security and privacy have been surveyed in [47].

Extracting co-location information about users, i.e., who is with whom, is becoming increasingly easier. More specifically, with the proliferation of mobile social networks, where users can check-in with others at various locations, the threat of available co-location information on users' location privacy is clear (as pointed out in [3]). Despite the mentioned works on quantifying the location privacy and the privacy of users in social networks, as well as the extensive research on privacy loss due to others, there has not been a study on evaluating location privacy where co-location information is considered. We bridge the gap between studies on location privacy and social networks, and we propose the first analytical framework for quantifying the effects of co-location information on location privacy, where users can also make use of obfuscation mechanisms.

## 9 Conclusion

In this paper, we have studied the effect on users' location privacy when co-location information is available, in addition to individual (obfuscated) location information. To the best of our knowledge, this is the first paper to quantify the effects of co-location information that stems from social relationships between users on location privacy; as such it constitutes a first step towards bridging the gap between studies on location privacy and social networks. Indeed, most studies on geo-location and social networks look at how social ties can be inferred from co-locations between individuals and how social ties can be used to de-anonymize mobility traces. We have shown that, by considering the users' locations jointly, an adversary can exploit co-location information to better localize users, hence decreasing their individual privacy. Although the optimal joint localization attack has a prohibitively high computational complexity, the polynomial-time approximate inference algorithms that we propose provide good localization performance. An important observation from our work is that a user's location privacy is no longer entirely in her control, as the co-locations and the individual location information disclosed by other users significantly affect her own location privacy.

The message of this work is that protection mechanisms must not ignore the social aspects of location information. Because it is not desirable to report dummy lists of co-located users (as this information is displayed on the users' profiles on social networks), a location-privacy preserving mechanism needs instead to generalize information about co-located users or to generalize the time of a social gathering, as well as the locations of users at other locations, in order to reduce the effectiveness of the attacks we suggested in this paper. As a first attempt to mitigate the privacy risks stemming from co-location information, we proposed a simple countermeasure that relies on cooperation between users and have demonstrated its effectiveness. We intend to address the design of social-aware location-privacy protection mechanisms (running on the users' mobile devices) to help the users assess and protect their location privacy when co-location information is available. An important aspect of generalization techniques is the tension between utility and privacy: For a user, reporting to be with "some friends" might not be sufficiently informative, and the generalized co-location information would fail to serve the user's purpose. Usability is also a crucial aspect for the adoption of technical protection mechanisms. We plan to investigate both the utility and usability aspects of such protection mechanisms through targeted user surveys.

As part of future work, we also plan to investigate the case where co-locations are not explicitly reported by the users, instead the adversary has access to the social ties between the users (e.g., friends, family, colleagues). Such ties can be associated with probabilistic co-location patterns, for instance, the fact that the locations of work-colleagues are often correlated during office hours.

## Acknowledgments

The authors are thankful to Stefan Mihaila for his help in collecting useful statistics about the use of co-location on Foursquare and to Konstantinos Chatzikokolakis and Marco Stronati for their help with the analysis of the effect of co-locations from a differential privacy perspective. Parts of this work were carried out while K. Huguenin and M. Humbert were with EPFL and R. Shokri was with ETH Zurich. This work was partially funded by the Swiss National Science Foundation with grant 200021-138089.

## References

[1] A.-M. Olteanu, K. Huguenin, R. Shokri, and J.-P. Hubaux, "Quantifying the Effect of Co-locations on Location Privacy," in *PETS*, 2014, pp. 184–203.

[2] "Facebook Messenger adds fast photo sharing using face recognition," The Verge, http://www.theverge.com/2015/11/9/9696760/facebook-messenger-photo-sharing-face-recognition, nov 2015, last visited: Nov. 2015.

[3] C. Vicente, D. Freni, C. Bettini, and C. S. Jensen, "Location-related privacy in geo-social networks," *IEEE Internet Computing*, vol. 15, no. 3, pp. 20–27, 2011.

[4] R. Shokri, G. Theodorakopoulos, J.-Y. Le Boudec, and J.-P. Hubaux, "Quantifying location privacy," in *S&P*, 2011, pp. 247–262.

[5] L. E. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite state markov chains," *The Annals of Mathematical Statistics*, vol. 37, no. 6, pp. 1554–1563, 1966.

[6] A. Narayanan and V. Shmatikov, "De-anonymizing social networks," in *S&P'09: Proc. of the 30th IEEE Symp. on Security and Privacy*, 2009, pp. 173–187.

[7] G. Ghinita, P. Kalnis, A. Khoshgozaran, C. Shahabi, and K.-L. Tan, "Private queries in location based services: Anonymizers are not necessary," in *SIGMOD*, 2008, pp. 121–132.

[8] R. L. Stratonovich, "Conditional Markov Processes," *Theory of Probability & its Applications*, vol. 5, no. 2, pp. 156–178, 1960.

[9] D. Koller and N. Friedman, *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[10] J. Pearl, *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, 2014.

[11] K. P. Murphy, Y. Weiss, and M. I. Jordan, "Loopy belief propagation for approximate inference: An empirical study," in *UAI*. Morgan Kaufmann Publishers Inc., 1999, pp. 467–475.

[12] R. I. M. Dunbar, "Neocortex size as a constraint on group size in primates," *Journal of Human Evolution*, vol. 22, no. 6, pp. 469–493, 1992.

[13] P. Ilia, I. Polakis, E. Athanasopoulos, F. Maggi, and S. Ioannidis, "Face/off: Preventing privacy leakage from photos in social networks," in *CCS*, 2015.

[14] M. E. AndrÃ©s, N. E. Bordenabe, K. Chatzikokolakis, and C. Palamidessi, "Geo-indistinguishability: differential privacy for location-based systems," in *CCS*, 2013, pp. 901–914.

[15] K. Chatzikokolakis, C. Palamidessi, and M. Stronati, "A Predictive Differentially-Private Mechanism for Mobility Traces," in *PETS*, 2014, pp. 21–41.

[16] D. Kifer and A. Machanavajjhala, "A rigorous and customizable framework for privacy," in *PODS*, 2012, pp. 77–88.

[17] C. Liu, S. Chakraborty, and P. Mittal, "Dependence Makes You Vulnerable: Differential Privacy Under Dependent Tuples," in *NDSS*, 2016, to appear.

[18] Y. Zheng, L. Liu, L. Wang, and X. Xie, "Learning transportation mode from raw GPS data for geographic applications on the web," in *WWW*, 2008, pp. 247–256.

[19] Y. De Mulder, G. Danezis, L. Batina, and B. Preneel, "Identification via location-profiling in GSM networks," in *WPES*, 2008.

[20] P. Golle and K. Partridge, "On the anonymity of home/work location pairs," in *Pervasive*, 2009, pp. 390–397.

[21] B. Hoh, M. Gruteser, H. Xiong, and A. Alrabady, "Enhancing security and privacy in traffic-monitoring systems," *IEEE Pervasive Computing*, vol. 5, no. 4, pp. 38–46, 2006.

[22] J. Krumm, "Inference attacks on location tracks," in *Pervasive*, 2007, pp. 127–143.

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

[24] M. L. Damiani, E. Bertino, and C. Silvestri, "The PROBE framework for the personalized cloaking of private locations," *Transactions on Data Privacy*, vol. 3, pp. 123–148, 2010.

[25] L. Huang, H. Yamane, K. Matsuura, and K. Sezaki, "Silent cascade: Enhancing location privacy without communication QoS degradation," in *SPC*, 2006, pp. 165–180.

[26] M. Srivatsa and M. Hicks, "Deanonymizing mobility traces: Using social network as a side-channel," in *CCS*, 2012, pp. 628–637.

[27] D. J. Crandall, L. Backstrom, D. Cosley, S. Suri, D. Huttenlocher, and J. Kleinberg, "Inferring social ties from geographic coincidences," *Proc. of the National Academy of Sciences*, pp. 1–6, 2010.

[28] N. Eagle, A. Pentland, and D. Lazer, "Inferring Friendship Network Structure by Using Mobile Phone Data," *Proc. of the National Academy of Sciences*, vol. 106, pp. 15 274–15 278, 2009.

[29] "How the NSA is tracking people right now," http://apps.washingtonpost.com/g/page/national/how-the-nsa-is-tracking-people-right-now/634/, 2013, last visited: Feb. 2014.

[30] C. A. Davis Jr, G. L. Pappa, D. R. R. de Oliveira, and F. de L Arcanjo, "Inferring the location of twitter messages based on user relationships," *Transactions in GIS*, vol. 15, no. 6, pp. 735–751, 2011.

[31] D. Xu, P. Cui, W. Zhu, and S. Yang, "Graph-based residence location inference for social media users," *IEEE MultiMedia*, vol. 21, no. 4, pp. 76–83, 2014.

[32] Y. Jia, Y. Wang, X. Jin, and X. Cheng, "TSBM: The temporal-spatial Bayesian model for location prediction in social networks," in *WI-IAT*, vol. 2, 2014, pp. 194–201.

[33] A. Uchiyama, S. Fujii, K. Maeda, T. Umedu, H. Yamaguchi, and T. Higashino, "UPL: Opportunistic localization in urban districts," *IEEE Transactions on Mobile Computing*, vol. 12, no. 5, pp. 1009–1022, 2013.

[34] E. Cho, S. A. Myers, and J. Leskovec, "Friendship and mobility: user movement in location-based social networks," in *KDD*, 2011.

[35] I. Polakis, G. Argyros, T. Petsios, S. Sivakorn, and A. D. Keromytis, "Where's wally?: Precise user discovery attacks in location proximity services," in *CCS*, 2015, pp. 817–828.

[36] M. Li, H. Zhu, Z. Gao, S. Chen, L. Yu, S. Hu, and K. Ren, "All your location are belong to us: Breaking mobile social networks for automated user location tracking," in *MobiHoc*, 2014.

[37] B. Henne, C. Szongott, and M. Smith, "Snapme if you can: Privacy threats of other peoples' geo-tagged media and what we can do about it," in *WiSec*, 2013, pp. 95–106.

[38] R. Dey, C. Tang, K. Ross, and N. Saxena, "Estimating age privacy leakage in online social networks," in *INFOCOM*, 2012.

[39] A. Mislove, B. Viswanath, K. P. Gummadi, and P. Druschel, "You are who you know: Inferring user profiles in online social networks," in *WSDM*, 2010, pp. 251–260.

[40] L. Backstrom, E. Sun, and C. Marlow, "Find me if you can: improving geographical prediction with social and spatial proximity," in *WWW*, 2010, p. 61.

[41] N. Vratonjic, K. Huguenin, V. Bindschaedler, and J.-P. Hubaux, "A location-privacy threat stemming from the use of shared public ip addresses," *IEEE Transactions on Mobile Computing*, vol. 13, no. 11, pp. 2445–2457, 2014.

[42] M. Humbert, E. Ayday, J.-P. Hubaux, and A. Telenti, "Addressing the concerns of the lacks family: Quantification of kin genomic privacy," in *CCS*, 2013, pp. 1141–1152.

[43] M. Gymrek, A. L. McGuire, D. Golan, E. Halperin, and Y. Erlich, "Identifying personal genomes by surname inference," *Science*, vol. 339, no. 6117, pp. 321–324, 2013.

[44] G. Biczók and P. H. Chia, "Interdependent privacy: Let me share your data," in *FC*, 2013, pp. 338–353.

[45] Y. Pu and J. Grossklags, "An economic model and simulation results of app adoption decisions on networks with interdependent privacy consequences," in *GameSec*, 2014, pp. 246–265.

[46] M. Humbert, E. Ayday, J.-P. Hubaux, and A. Telenti, "On non-cooperative genomic privacy," in *FC*, 2015.

[47] A. Laszka, M. Felegyhazi, and L. Buttyan, "A survey of interdependent information security games," *ACM Comput. Surv.*, vol. 47, no. 2, pp. 23:1–23:38, Aug. 2014.

**Alexandra-Mihaela Olteanu** is a Ph.D. candidate at EPFL. She earned her M.Sc. in computer science with a specialization in internet computing, from EPFL, in 2013. She worked as an intern at NUS University of Singapore and is a Google Anita Borg scholar. Prior to that, she earned her B.Sc. in Mathematics and Computer Science from Babes-Bolyai University, Romania, in 2008. Her research focuses on interdependent privacy risks.

**Kévin Huguenin** is a permanent researcher at LAAS-CNRS, France, which he joined in 2014. Prior to that, he worked as a post-doctoral researcher at EPFL and at McGill University. He earned his Ph.D. in computer science from the Université de Rennes, France, in 2010 and his M.Sc. degree from École Normale Supérieure de Cachan and the Université de Nice, France, in 2007. His research interests include security and privacy in networks and distributed systems.

**Reza Shokri** is a post-doctoral research fellow at UT Austin, TX, USA, which he joined in 2014. He is currently visiting Cornell Tech in New York, NY, USA. Prior to that, he was a post-doctoral researcher at ETH Zurich. He earned his Ph.D. on the quantification and protection of location privacy from EPFL in 2013 and his M.Sc. in software computer engineering from the University of Tehran, Iran, in 2007. His research focuses on computational privacy.

**Mathias Humbert** is a post-doctoral researcher in the Center for IT-Security, Privacy, and Accountability (CISPA) at Saarland University. He completed his Ph.D thesis on interdependent privacy in 2015, under the supervision of Jean-Pierre Hubaux, at EPFL. Prior to this, he received his M.Sc. (2009) and B.Sc. (2007) degrees from EPFL, and studied for one year (2007-2008) at UC Berkeley. His research interests include genomic privacy, privacy in online social networks, location privacy.

**Jean-Pierre Hubaux** is a full professor at EPFL. His current research activity is focused on privacy, notably in pervasive communication systems. In 2011, he started research activity in genomic privacy, in close collaboration with geneticists. In 2008, he completed a graduate textbook, entitled Security and Cooperation in Wireless Networks, with Levente Buttyan. He held visiting positions at the IBM T.J. Watson Research Center and at UC Berkeley. He is a fellow of both the ACM and IEEE.