On the Impact of User Movement Simulations in the Evaluation of LBS Privacy-Preserving Techniques

Sergio Mascetti[1], Dario Freni[1], Claudio Bettini[1],
X. Sean Wang[2], and Sushil Jajodia[3]

[1] DICo, Università di Milano
[2] Dept. of CS, University of Vermont
[3] CSIS, George Mason University

Abstract. The evaluation of privacy-preserving techniques for LBS is often based on simulations of mostly random user movements that only partially capture real deployment scenarios. We claim that benchmarks tailored to specific scenarios are needed, and we report preliminary results on how they may be generated through an agent-based context-aware simulator. We consider privacy preserving algorithms based on spatial cloaking and compare the experimental results obtained on two benchmarks: the first based on mostly random movements, and the second obtained from the context-aware simulator. The specific deployment scenario is the provisioning of a friend-finder-like service on weekend nights in a big city. Our results show that, compared to the context-aware simulator, the random user movement simulator leads to significantly different results for a spatial-cloaking algorithm, under-protecting in some cases, and over-protecting in others.

1 Introduction

Location-based services (LBS) are often cited as killer applications for the latest GPS-equipped 3G phones. These phones are slated to be massively distributed in 70 countries. While car navigation and identification of nearest points of interest are already widely used services, more interest are generating the so-called *friend-finder* services as a class of LBS that will change once more our way to interact. A friend-finder service reveals to a participating user the presence of other close-by participants belonging to a particular group (friends is only one example), possibly showing their position on a map. From a technical point of view, in contrast to services that find nearest points of interests, this service is characterized by a sequence of LBS requests instead of single ones, since a user may want to periodically check, while moving or even while staying in the same place, for close-by participants.

Sociological studies have shown that a large number of users perceive the release of their precise location, as part of a LBS request, as a possible privacy threat [1]. Considering friend-finder services it is easy to identify two types of

privacy threats: a) the association of the identity of the user with the specific group of persons he is interested in may reveal her religious, sexual, or political orientation, and b) the association of the identity of the user with her precise location may reveal what kind of places she has been to, or that she has not been where she was supposed to be at that time.

As formally shown in [2] the likelihood of a privacy violation, and consequently the defense techniques to be applied, strongly depend on the knowledge that an adversary may have. In the friend-finder service the service provider (SP) may not be trusted or the communication channels may be insecure; then, the adversary's knowledge may include the precise identity and location information submitted with each request, and both the above privacy threats would become real privacy breaches. The substitution of identities with pseudonyms does not entirely solve the problem, if, for example, the adversary happens to know who is at the location at the time reported in the request (e.g., in the case the issuer of the request uses a fidelity card at a store). In some cases, the adversary may also be able to recognize sequences of requests as issued by the same anonymous user (e.g., by observing the same pseudonym or by spatio-temporal tracking) and use this information to re-identify the issuer.

Several defense techniques against both threats under different adversary models have been proposed, and may be applied to the friend-finder service; however, current proposals very rarely have formal assessments of the provided privacy preservation, and are generally supported by experimental results based either on real datasets of questionable significance for real LBS services (i.e., trucks or school bus traces) or on data simulations based on mostly random user movements that hardly match the specific deployment scenario of a LBS service.

In order to understand if the use of simulations based on mostly random user movements may be a real problem, or if it is actually useful and safe to use these simulations, we considered a typical deployment scenario for a friend-finder service: a large number of young people using the service on a weekend night in a large city like Milan, Italy. We performed a deep study, using different sources, including on-line surveys, of the parameters characterizing this scenario. We then used the Brinkhoff simulator [3], widely used in testing LBS privacy preservation, to generate, based on the parameters, a first dataset of user movements. A second dataset was created with a personalized version of the Siafu agent-based context-aware simulator [4] which is able to capture much more details of our scenario. Then, based on a common metric for privacy and quality of service evaluation, we run a large number of tests on both datasets, considering different abilities of re-identification by the adversary, as well as different privacy preserving techniques.

Our results consistently show that (i) in some cases the evaluation on random movement simulations leads to the definition of overprotective techniques and (ii) in other cases, the techniques that are shown to meet privacy requirements based on those simulations do not meet them when tested with more realistic context-aware simulations.

We focus our technical treatment on protecting the association of the user with the request he has issued (e.g., with the group of people he is interested in,

as in threat (a) described above), even if we believe that our arguments can be easily extended to techniques only aimed to protect the location.

Related work

We are not aware of related work in this area considering specifically the relevance of realistic simulations in LBS. There are however several studies on user movements with impact on many different application areas including epidemiology, transportation, computer networks, marketing, as well as LBS. A very interesting study supporting an argument against random movement simulations recently appeared [5]. In the following we briefly report the main techniques currently proposed for LBS privacy preservation, identifying the ones similar to those tested in our experiments, and the ones using simulations to generate the datasets for experiments.

Privacy preserving solutions based on cryptographic techniques that totally hide the location information in requests, even to the SP, have been recently proposed [6] for LBS based on 1-NN queries, and may be probably adapted for the service we consider. If proven to be correct, no simulation would be needed for these techniques since no information would leak from any request and the above privacy threats do not apply. However, this adaptation is still to be investigated, and there are some general concerns with these approaches regarding efficiency and flexibility.

A popular alternative technique is spatial cloaking, consisting in the generalization of the spatial information transmitted to the SP as part of a service request. By receiving generalized locations, the SP can only return approximate results on the presence of close-by group members and their positions; while it may be possible to have a trusted entity in the middle filtering the communication and improving the precision, the related overhead costs should be taken into account in evaluating the trade-off between generalization and quality of service. While in this paper we consider techniques based on spatial cloaking as in [7–9, 2, 10], other proposals have considered different techniques, including the generation of dummy requests, the use of incremental requests, or the substitution in the request of the position of the issuer with a region that does not include her (see among others [11–13]).

Most of the proposals for LBS privacy have only considered requests in isolation while a few have also addressed the cases in which sequences of requests can be exploited by the adversary ([14, 15] among others), as in the friend-finder service. A related problem is privacy-aware publication of trajectories [16, 17]; even if this has some aspects more similar to database publication than to service request privacy preservation, we believe that our results may be important for these studies as well.

Synthetic, mostly random, user movements obtained by the Brinkhoff simulator or other simulators have been used in most of the above cited papers as well as in our own previous work.

Organization

The rest of the paper is organized as follows. In Section 2 we formally define how we evaluate the privacy of LBS requests, or equivalently, how we measure the risk of a privacy violation upon issuing a request. In Section 3 we explain how the two datasets were obtained from the generators based on the parameters characterizing the deployment scenario. In Section 4 we briefly explain the privacy preservation algorithms being used and we report our experimental results. Section 5 concludes the paper.

2 Privacy metric of generalized requests

As mentioned in the introduction, we are concerned with privacy protection via location generalization (also called spatial cloaking). In this section, we formalize the adversary model we consider in this paper, and give a metric to measure the privacy provided by a set of generalized requests against the adversary.

2.1 Requests, original requests, and generalized requests

We first formally define requests and generalized request for LBS. A request issued by a user without alteration is called an *original* request, and a *generalized* request is one that is sent to the service provider and has been altered from the original one for the purpose of privacy protection. Both kinds of requests are called *requests* and denoted r. A convention in this paper is to use r' to denote generalized requests to emphasize its generalized nature, while use r to denote original requests, if not specified otherwise.

Either the client software or a trusted medium transforms (or *generalizes*) an original request to a generalized one. In this paper, we are not concerned about *how* the generalization has happened, but rather on the *resulting* generalized requests and their privacy properties. In the experimental section, we evaluate the performance of generalization algorithms based on the generalized requests they generate.

Each LBS *request* r, either original or generalized, is logically divided into three parts: $IDdata$, $STdata$, and $SSdata$, containing user identification data, location and time of the request, and other service parameters, respectively. In the sequel, the spatial and temporal components in $STdata$ are denoted with $Sdata$ and $Tdata$, respectively. In this paper, for the sake of simplicity, we consider space and time as discrete domains. However, our results can be easily extended to the case in which these two domains are continuous.

Each generalized request r' must correspond to an original request r such that the difference between r and r' is only in $SData$ and furthermore, $r.Sdata \subseteq r'.Sdata$, i.e., the spatial region of the generalized request must contain (or be equal to) the spatial region of the original request[4]. We use $issuer(r)$ to denote the actual issuer of the (original or generalized) request r.

[4] Here "region" can be a point.

2.2 Adversary model

The objective here is to provide an adversary model that captures a general class of adversary models. In a sense, our adversary model is an adversary "meta-model". This adversary meta-model concerns two aspects of knowledge that an adversary might have: (1) knowledge of users' whereabouts (i.e., their locations), and (2) correlation of (generalized) requests. These two aspects cover the (explicit or implicit) assumptions appeared in the relevant literature.

For users' locations, we assume that the adversary has the knowledge expressed as the following $Ident$ function:

$$Ident_t : the\ Areas \longrightarrow the\ User\ sets,$$

that is, given an area A, $Ident_t(A)$ is the set of users whom, through certain means, the adversary has identified to be located in area A at time t. In the following, when no confusion arises, we omit the time instant t. We further assume that this knowledge is *correct* in the sense that these identified users in reality are indeed in area A at the time.

For a given user i, if there exists an area A such that $i \in Ident(A)$, then we say i is *identified* by the adversary. Furthermore, we say that i is *identified in* A. Note that there may be users who are also in A but the adversary does not identify them. This may happen either because the adversary is not aware of the presence of users in A, or because the adversary cannot identify these users even if he is aware of their presence. We do not distinguish these two cases in our adversary model as we shall see later that the distinction of the two cases does not make any perceptible difference in the ability of the adversary when the total population is large.

Clearly, in reality, there are lots of different sources of external information that can lead the adversary to estimate the location of users. Some may lead the adversary to know that a user is in a certain area, but not the exact location. For example, an adversary may know that Bob is in a pub (due to his use of a fidelity card at the pub), but may not know which room he is in. Some statistical analysis may be done to derive the *probability* that Bob is in a particular room, but this is beyond the scope of this paper.

The most conservative assumption regarding this capability of the adversary is that $Ident(A)$ will give *exactly* all the users for each area A. It can be seen that if the privacy of the user is guaranteed in this most conservative assumption, then privacy is also guaranteed against any less precise $Ident$ function. However, this conservative assumption is unlikely true in reality, while some observed that this assumption degenerates the quality of service unnecessarily. It will be interesting to see how much privacy and quality of service change with more realistic $Ident$ functions. This is partly the goal of our paper.

As part of this adversary model regarding the location and users, we also assume another function:

$$Num_t : the\ Areas \longrightarrow [0, \infty),$$

that is, given an area A, $Num_t(A)$ gives an estimate of the number of users in the area at time t. This function can be derived from statistical information publicly available or through some kind of counting mechanism such as tickets to a theater. Again, when no confusion arises, we do not indicate the time instant t.

The second part of the adversary model is his ability to correlate requests. We formalize this with the following function L:

$$L : the\ Requests \longrightarrow the\ Request\ sets,$$

that is, given a (generalized) request r', $L(r')$ gives a set of requests such that the adversary has concluded, through certain means, are issued by the same user who issued the request r'. In other words, all the requests in $L(r')$ are *linked* to r', although the adversary may still not know who the user is.

Note that $L(r)$ may only give an (often small) subset of all the requests issued by the issuer of r. On the other hand, we assume that the function L is *correct* in the sense that each request in $L(r)$ is indeed issued by the same user in reality. A set of requests is called a *trace*, denoted τ, if from the link function L we understand that all requests are issued by the same user. The requests in τ are implicitly ordered along the time dimension.

As in the case for $Ident$ function, the most conservative assumption on correlation is that $L(r)$ gives exactly *all* the (generalized) requests that are issued by the issuer of r. This is a very strong assumption that may lead to severely decrease quality of service when accompanied with the most conservative assumption about the $Ident$ function. Again, a partial goal of this paper is to study the impact of a less conservative but more realistic assumption on L.

In [2], we proposed a formal framework to model LBS privacy attacks and defenses for the static case. The main idea is that the safety of a defense technique can be formally evaluated only if the *context*, i.e., the assumptions about the adversary's external knowledge, is explicitly stated. Following this methodology, in this paper, a context C_H is given by three functions $Ident$, Num, and L, that is

$$C_H = (Ident, Num, L).$$

In the next section, we formalize the attack on the generalized requests that an adversary can perform in a context C_H.

A consequence of restricting to context C_H is that, analogously to the related work in this area, we focus our attention on using only $STdata$ as a *quasi-identifier*. Intuitively, a quasi-identifier in a request is a combination of values that can be used to provide more information on who the actual issuer of a request may be than without these values. For example, if the $Ident$ function is given, the $STData$ in the request is a quasi-identifier as it may provide information on the actual issuer, as shown in the next subsection. In principle, any information contained in a request should be carefully analyzed to see if it may serve as a quasi-identifier. For example, the IDdata part is an obvious target, and some service specific parameters may be used to link the request to users. However, these aspects are outside the scope of this paper.

2.3 Privacy Evaluation

The general question for this subsection is, given a set of generalized requests and a context C_H, how much privacy the users who issued these requests have.

We want to find the following function:

$$Att : the\ Request\ set \times the\ Users \longrightarrow [0, 1],$$

Intuitively, given a (generalized) request r' and a user i, $Att(r', i)$ gives the probability that the adversary can derive from C_H that i is the issuer of r' among all the users.

In the following of this section we show how to specify the attack function for context C_H. Once the attack function is specified, we can use the following formula to evaluate the privacy value of a request:

$$Privacy(r') = 1 - Att_{C_H}(r', issuer(r')) \tag{1}$$

Intuitively, this value is the probability that the attacker will not associate the issuer of request r' to r'.

In order to specify the Att function, we introduce the function $Inside(i, r')$ that indicates the probability of user i to be located in $r'.Sdata$ at the time of the request. Intuitively, $Inside(i, r') = 1$ if user i is identified by the adversary as one of the users that are located in $r'.Sdata$ at time $r'.Tdata$, i.e., $i \in Ident_t(r'.Sdata)$ when $t = r'.Tdata$. On the contrary, $Inside(i, r') = 0$ if i is recognized by the adversary as one of the users located outside $r'.Sdata$ at time $r'.Tdata$, i.e., there exists an area A with $A \cap r'.Sdata = \emptyset$ such that $i \in Ident(A)$. Finally, if neither of the above cases hold, then the adversary does not know where i is. There is still a probability that i is in $r'.Sdata$. Theoretically, this probability is the number of users in $r'.Sdata$ that are not recognized by the adversary (i.e., $Num(r'.Sdata) - |Ident(r'.Sdata)|$) divided by all the users who are not recognized by the adversary anywhere (i.e., $|I| - |Ident(\Omega)|$, where I is the set of all users, and Ω is the entire area for the application). Formally,

$$Inside(i, r') = \begin{cases} 1 & \text{if } i \in Ident(r'.Sdata) \\ 0 & \text{if } \exists A : A \cap r'.Sdata = \emptyset \text{ and } i \in Ident(A) \\ \frac{Num(r'.Sdata) - |Ident(r'.Sdata)|}{|I| - |Ident(\Omega)|} & \text{otherwise} \end{cases} \tag{2}$$

We can now define the Att function in context C_H. For the sake of presentation, let us first consider the attack in the snapshot context

$$C_{snap} = (Ident, Num, L_{snap}),$$

where for each generalized request r', $L_{snap}(r') = \{r'\}$. In this special case, the probability of a user i of being the issuer of r' is given by the probability of i being in $r'.Sdata$ at the time of the request, normalized among all the users in I. Formally, the attack can be defined as:

$$Att_{C_{snap}}(r', i) = \frac{Inside(i, r')}{\sum_{i' \in I} Inside(i', r')} \tag{3}$$

When the total population of users is large (relative to the number of users whose locations are known to the adversary), then the "otherwise" case in Formula 2 is very small, albeit nonzero. Intuitively, if a user i falls into this case, then the adversary cannot really distinguish this particular user from all other users who also fall into this case. For such a user i, we can simply give a value $1/|I|$ to $Att_{C_{snap}}(r', i)$. We could give $1/(|I| - Num(\Omega))$, but this does not make much impact in practice. Now it's easy to see that

$$Att_{C_{snap}}(r', i) \approx \begin{cases} 1/Num(r'.Sdata) & \text{if } Inside(i, r') = 1 \\ 0 & \text{if } Inside(i, r') = 0 \\ 1/|I| & \text{otherwise} \end{cases} \quad (4)$$

The above formula makes intuitively sense. Indeed, if i is recognized as inside $r'.Sdata$, without any other information, the adversary cannot distinguish him/her from any of the $Num(r'.Sdata)$ people in the area who might be the issuer. If i is recognized outside, then clearly i cannot be the issuer due to our definition of (generalized) requests. If i is not recognized anywhere (meaning he/she can be anywhere), then the attacker cannot distinguish him/her from any of the other people who are not recognized. Since we assume the total population is much greater than $Num(\Omega)$, the probability that i is the issuer is close to $1/|I|$.

Example 1. Consider the situation shown in Figure 1(a) in which there is the request r' such that, at time $r'.Tdata$, there are three users in $r'.Sdata$: one of them is identified as i_1, the other two are not identified. The adversary can also identify users i_2 and i_3 outside $r'.Sdata$ at time $r'.Tdata$. Assume that the set I contains 100 users.

(a) First request, r'. (b) Second request, r''.

Fig. 1. Example of attack

Clearly, i_2 and i_3 have zero probability of being the issuers, since they are identified outside $r'.Sdata$ and due to the assumption that the spatial region

of any generalized request must contain the spatial region of the original request. On the contrary, the adversary is sure about the fact that i_1 is located in $r'.Sdata$. By Equation 3, the attack associates i_1 to r' with likelihood $1/(\sum_{i' \in I} Inside(i', r'))$. By Formula 2, for each user i in $I \setminus \{i_1, i_2, i_3\}$, $Inside(i, r') = 2/100$. Therefore, $\sum_{i' \in I} Inside(i', r') = 97 * 2/100 + 1 \approx 3$. Consequently, the probability of i_1 to be the issuer of r' is approximately $1/3$. Moreover, each user $i \in I \setminus \{i_1, i_2, i_3\}$ has a probability to be the issuer of about $(2/100)/3 = 2/300$.

In the general case $L(r') \supseteq \{r'\}$, we can evaluate, analogously to the snapshot case, the probability that a user is located in the generalized region of all the requests in the trace $\tau = L(r')$. So, we can extend the $Inside$ function to traces where, given a trace τ and a user i, $Inside(i, \tau)$ is the probability that user i is located, for each request r' in τ, in $r'.STdata$. Then, the attack is

$$Att_{C_H}(r', i) = \frac{Inside(i, L(r'))}{\sum_{i' \in I} Inside(i', L(r'))} \tag{5}$$

We now turn to consider how to compute $Inside(i, \tau)$.

First consider some easy cases. If $i \in Ident(r')$ for all requests $r' \in \tau$, then $Inside(i, \tau) = 1$. If $i \in Ident(A)$ and $A \cap r'.Sdata = \emptyset$ for an area A and at least one requests $r' \in \tau$, then $Inside(i, \tau) = 0$.

The rest of cases are difficult ones. To calculate $Inside(i, \tau)$, we need to consider the likelihood of someone moving from one location to/from another in the specific times. In this paper, we advocate the following as a reasonable approach. We assume for each pair of locations A and B and two times t_1 and t_2, we know the probability of a user i being in B at time t_2 conditioned on the fact that the user is in A at time t_1. In formalism, consider two random variables X: "i is inside A at time t_1" and Y: "i is inside B at time t_2", where A and B are two areas and t_1 and t_2 are two different times. We assume the adversary knows the value $P(Y|X)$.

We note that $P(Y|X)$ in general is not the same as $P(Y)$. Indeed, how likely user i is in B can depend on how likely the same user is in A. Take two extreme examples: if A and B are very far away and t_1 and t_2 are close to each other, then i cannot be in B at t_2 if i is in A at t_1, i.e., $P(Y|X) \approx 0$. On the other hand, if A and B are just two locations along a one-way road and the difference between times t_1 and t_2 matches the time needed to move from A to B with a normal moving speed, then $P(Y|X) \approx 1$. In practice, this value can be derived from historical observations and experiences.

Now, assume τ consists of the requests $r'_1, \ldots, r'_k$. We form a Bayesian network for each user i with X_1, ..., X_k as the nodes, where each X_j corresponds to the random variable: "user i is in $r_j.Sdata$ at time $r_j.Tdata$". In this network, for each node X_h that satisfies the condition (denoted c) $i \in Ident_t(r'_h.Sdata)$ with $t = r'_h.Tdata$, we draw an arc towards each other node X'_h which does not satisfy condition C. In addition, for each pair of nodes r'_h and r''_h such that neither satisfy condition c, we draw an arc from X'_h to X''_h if the $r'_{h'}.Tdata < r'_{h''}.Tdata$. (The resulting network is acyclic.) As we have assumed, we know the

value $P(X'_h|X_h)$ for each arc X_h to X'_h. Denote by E the conjunctive fact that $P(X_h) = 1$ for each $r_h \in \tau$ that satisfies condition c. What we want to find is $P(X_1, \ldots, X_k|E) = Inside(i, \tau)$. This is a well-studied belief revision problem, and many computation and approximation methods exist. (Note that if we apply this method to the easier cases mentioned earlier, we would arrive at the correct values.)

Example 2. Continue from Example 1 and assume a second request r'' (see Figure 1(b)) is issued after r' and that r'' is linked with r', so τ consists of these two requests. At time $r''.Tdata$, there are 4 users inside $r''.Sdata$, two of which are identified as i_1 and i_2. No user is identified outside $r''.Sdata$. From the above discussion, it follows that $Inside(i_2, \tau) = Inside(i_3, \tau) = 0$ since i_2 and i_3 are identified outside the first generalized request r'. All the other users have a non-zero probability of being inside the generalized region of each request in the trace. In particular, $Inside(i_1, \tau) = 1$ since i_1 is recognized in both requests. Consider a user $i \in I \setminus \{i_1, i_2, i_3\}$, and denote X and Y being the assertion that "i is in $r'.Sdata$ at time $r'.Tdata$" and "i is in $r''.Sdata$ at time $r''.Tdata$". In this case, the Bayesian network for i has two nodes $X_{r'}$ and $X_{r''}$, and there is an arc from $X_{r'}$ to $X_{r''}$ since r'' is issued after r' is. Now let us assume $P(X_{r''}|X_{r'}) = 0.75$, i.e., there is a 75% likelihood that someone in $r'.Sdata$ will move to $r''.Sdata$ at the specified times. Now compute $Inside(i, \tau) = P(X_{r'}, X_{r''}) = P(X_{r'})P(X_{r''}|X_{r'}) = 2/97 * 0.75$. Now the sum of all the $Inside(j, \tau)$ value is $1 + 0 + 0 + 97 * 2/97 * 0.75 = 2.5$. The attack value under these assumptions then is as follows: For $Att_{C_H}(r'', i_1) = 1/2.5 = 40\%$, $Att_{C_H}(r'', i_2) = Att_{C_H}(r'', i_3) = 0$, and $Att_{C_H}(r'', i) = (2/97) * .75/2.5 \approx 0.6\%$ for all other 97 users i.

To make the situation more interesting, let us remove the fact that i_2 was recognized outside at time $r'.Tdata$, and we want to figure out the value $Inside(i_2, \tau)$. In this case, let us assume $P(X_{r'}|X_{r''}) = 0.75$, namely people who are in $r''.Sdata$ have a 75% likelihood to be from $r'.Sdata$. Under the fact E that i_2 is in $r''.Sdata$, then we know $Inside(i_2, \tau) = P(X_{r'}, X_{r''}|E) = 0.75$. Then the sum of $Inside$ values is $1+0.75+0+97*2/97*0.75 = 3.25$. Hence, $Att_{C_H}(r'', i_1) \approx 31\%$, $Att_{C_H}(r'', i_2) \approx 23\%$, $Att_{C_H}(r'', i_3) = 0$, and $Att_{C_H}(r'', i) = (2/97) * 0.75/3.25 \approx 0.47\%$ for each other 97 users i. This is an interesting exercise as it reveals that if we add i_2 to be possibly in $r'.Sdata$ (with 75% probability), then the likelihood that i_1 is the issuer decreases, which is intuitively correct.

It is worth noting that the definition of attack in context C_H is a proper extension of the attack that can be defined in the conservative context in which the adversary knows the location and the identity of each user in each time instant. The historical attack in this context was first proposed in [14]. The idea is that the only users that have non-zero probability of being the issuer of a trace of requests are those whose spatio-temporal location is contained in the generalized region of every request in the trace. It can be easily seen that, if each user can be identified at each time instant, then the $Inside()$ function returns either 0 or 1 and hence the attack we specified for context C_H assigns a zero

probability to each user that is located outside the generalized region of any request in the trace.

3 The *MilanoByNight* simulation

In order to evaluate privacy-preserving techniques applied to LBS, a dataset of users' movements is needed. In our experiments, we want to focus on privacy threats that arise when using a friend finder service, as described in Section 1. We suppose that this kind of service is primarily used by people during entertainment hours, especially at night. Therefore, the ideal dataset for our experiments should represent movements of people on a typical Friday or Saturday night in a big city, when users tend to move to entertainment places. To our knowledge, currently there are no datasets like this publicly available, specially considering that we want to have large scale, individual, and precise location data (i.e., with the same approximation of current consumer GPS technology). In this section we describe how we generated this user movement dataset.

3.1 Relevant Parameters

For our experiments we want to artificially generate movements for $100,000$ users on the road network of Milan[5]. The total area of the map is 324 km^2, and the resulting average density is 308 users/km^2. Very detailed digital vector maps of the city have been generously provided by the municipality of Milan. The simulation includes a total of $30,000$ home buildings and $1,000$ entertainment places; the first value is strictly related to the considered number of users, while the second is based on real data from public sources which also provide the geographical distribution of the places. Our simulation starts at 7 pm and ends at 1 am. During these hours, each user moves from house to an entertainment place, spends some time in that place, and possibly moves to another entertainment place or go back home.

All probabilities related to users' choices are modeled with a probability distributions. For this specific data generation, some of the important parameters of the simulation are:

- **Source and destination**. These are the locations essential to define movements. They may be homes or entertainment places. Some places in some districts are more popular than others.
- **StartingTime**. The time at which a user leaves her home to go to the first entertainment place.
- **Permanence**. How long will a user stay at one entertainment place?
- **NumPlaces**. How many entertainment places will a user visit on one night?

In order to have a realistic model of these distributions, we prepared a survey to collect real users data. We are still collecting data, but the current parameters are based on interviews of more than 300 people in our target category.

[5] $100,000$ is an estimation of the number of people participating in the service we consider.

3.2 Weaknesses of mostly random movement simulations

Many papers in the field of privacy preservation in LBS use artificial data generated by moving object simulators to evaluate their techniques. However, most of the simulators are usually not able to reproduce a realistic behavior of users. For example, objects generated by the Brinkhoff generator [3] cannot be aggregated in certain places (e.g., entertainment places). Indeed, once an object is instantiated, the generator chooses a random destination point on the map; after reaching the destination, the object disappears from the dataset. For the same reason, it is not possible to reproduce simple movement patterns (e.g.: a user going out from her home to another place and then coming back home), nor to simulate that a user remains for a certain time in a place.

Despite these strong limitations, we made our best effort to use the Brinkhoff simulator to generate a set of user movements with characteristics as close as possible to those explained in Section 3.1. For example, in order to simulate entertainment places, some random points on the map, among those points on the trajectories of users, were picked. The simulation has the main purpose of understanding if testing privacy preservation over random movement simulations gives significantly different results with respect to more realistic simulations.

3.3 Generation of user movements with a context simulator

In order to obtain a dataset consistent with the parameters specified in Section 3.1, we need a more sophisticated simulator. For our experiments, we have chosen to customize the Siafu context simulator [4]. With a context simulator it is possible to design models for agents, places and context. Therefore, it is possible to define particular places of aggregation and make users dynamically choose which place to reach and how long to stay in that place. In our simulation homes are distributed almost uniformly on the map, with a minor concentration on the central zones of the city. Entertainment places are mostly concentrated in 5 zones of the city.

The distributions for *StartingTime*, *Permanence* and *NumPlaces* parameters introduced in Section 3.1 were modeled with the results of the survey. For example, the time of permanence in an entertainment place was modeled according to the following percentages derived from the survey: 9.17% of the users stays less than 1 hour, 34.20% stays between 1 and 2 hours, 32.92% stays between 2 and 3 hours, 16.04% stays between 3 and 4 hours, and 7.68% stays more than 4 hours.

Following these parameters, in our dataset users spend 50.87% of the time at home, 7.28% of the time moving from one place to another and 41.85% of the time in entertainment places. When a user moves from one place to another, she decides whether to go on foot or by car. In general, if an entertainment place is farther than 500 meters, people tend to move by car, and this is reflected in the simulation. The average speed of movements by car is 20 km/h, while the average speed on foot is 3.6 km/h. With our parameters 10.64% of movements are done on foot, while all the others are done by car.

4 Experimental results

In this section we show the results of our experimental evaluation. We first define how we evaluate the quality of service in Section 4.1, then we describe the experimental setting in Section 4.2 and the generalization algorithms we used in Section 4.3. Finally, in Sections 4.4 and 4.5 we show the impact of the simulation parameters and of the user movements, respectively, in the evaluation of the generalization algorithms.

4.1 Evaluation of the Quality of Service

Different metrics can be defined to measure QoS for different kind of services. For instance, for the friend-finder service we are considering, it would be possible to measure how many times the generalization leads the SP to return an incorrect result i.e., the issuer is not notified of a close-by friend or, vice versa, the issuer is notified for a friend that is not close-by. While this metric is useful for this specific application, we want to measure the QoS independently from the specific kind of service. For this reason, in this paper we evaluate how QoS degrades in terms of the perimeter of the generalized region. If the generalized region is too large, the service becomes useless. For this purpose, we introduce a new parameter, called $maxP$, that indicates this threshold in terms of the maximum perimeter. We assume that no request is sent to the SP with a perimeter larger than $maxP$.

4.2 Experimental settings

In our experiments we used two datasets of users movements. The dataset AB (Agent-Based) was generated with the customized Siafu simulator as described in Section 3.3, while the dataset MRM (Mostly Random Movement) was created with the Brinkhoff simulator as described in Section 3.2. In both cases, we simulate LBS requests for the friend-finder service by choosing random users in the simulation, we compute for each request the generalization according to a given algorithm, we evaluate QoS as explained in Section 4.1 and privacy according to formula (1) presented in Section 2.

The most important parameters that characterize the simulations are reported in Table 1, with the values in bold denoting default values. The *number of users* indicates how many users are in the simulation, and the simulations are designed so that this number remains almost constant at each time instant. In every two minutes, each user has a probability P_{req} of issuing a request. For technical reasons, the reported tests are based on a time frame of three hours over the total six hours of the MilanoByNight scenario. This implies that in the default case we consider a total of 45 requests (one every four minutes of the considered time frame). The parameter P_{id-in} indicates the probability that a user is identified when she is located in a entertainment place while P_{id-out} is the probability that a user is identified in any other location (e.g., while moving from home to a entertainment place). While we also perform experiments where

the two probabilities are the same, our scenario suggests as much more realistic a higher value for P_{id-in} (it is considered ten times higher than P_{id-out}). This is due to the fact that restaurants, pubs, movie theaters, and similar places are likely to have different ways to identify people (fidelity or membership cards, wifi hotspots, cameras, credit card payments, etc.) and in several cases more than one place is owned by the same company that may have an interest in collecting data about its customers.

Finally, P_{link} indicates the probability that two consecutive requests can be identified as issued by the same user.[6] While we perform our tests considering a full range of values, the specific default value reported in the table is due to a recent study on the ability of linking positions based on spatio-temporal correlation [18].

Table 1. Parameter values

Parameter	Values
dataset	***AB***, *MRM*
number of users	10k, 20k, 30k, 40k, 50k, 60k, 70k, 80k, 90k, **100k**
P_{req}	0.1, 0.2, 0.3, 0.4, **0.5**, 0.6, 0.7, 0.8, 0.9, 1.0
P_{id-in}	0.1, **0.2**, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0
P_{id-out}	0.01, **0.02**, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08, 0.09, 0.1
P_{link}	0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, **0.87**, 0.9, 1.0

The experimental results we show in this section are obtained by running the simulation for 100 issuers and then computing the average values.

4.3 The Generalization Algorithms Used in the Experiments

In our experiments we evaluate the privacy and the QoS of requests generalized by using two algorithms previously proposed in the literature. The first one, called *Grid*, was presented in [19], and it is used as a representative of several algorithms aimed at guaranteeing k-anonymity in the snapshot case, i.e., these algorithms do not take into account link ability of the adversary. Intuitively, this particular algorithm partitions all users into blocks, each one having at least cardinality k. Then, it computes the generalized region as the minimum bounding rectangle (MBR) that covers the location of the users in the same block as the issuer.

The second algorithm, *Greedy*, was first proposed in [14] and a similar idea was also described in [15]. The use of Greedy is intended to represent algorithms aimed at preserving privacy in the historical case, i.e., the general C_H context,

[6] The limitation to consecutive requests is because in our specific scenario we assume linking is performed mainly through spatio-temporal correlation.

assuming that the attacker may actually obtain and recognize traces of requests from the same issuer. This algorithm computes the generalization of the first request r in a trace using an algorithm for the snapshot case. While doing this, the set A of users located in the generalized region is stored. The generalized regions of the successive request r' linked with r is then computed as the MBR of the location of the users in A at the time of r'. In our implementation we use *Grid* as the snapshot algorithm to compute the generalization of the first request.

For the purpose of our tests, we modified the two algorithms above so that each generalized region has a perimeter always smaller than $maxP$. To achieve this, if the perimeter of the generalized region is larger than $maxP$, then the region is iteratively shrunk, until its perimeter is below $maxP$, by excluding from the MBR the user that is farther from the issuer. In the *Greedy* algorithm, when a user is excluded from the generalized region, then it is also excluded from the set A of users, and hence he is not used in the generalization of the successive requests. Eventually, when the set A contains the issuer only, a snapshot generalization is executed again and A is reinitialized.

In addition to the input request r, and the location of all the users in the system, the considered algorithms require two additional parameters: the value k, and the threshold $maxP$. In our tests, we used values for k between 10 and 60 (default is 10) and values for $maxP$ between 1000 to 4000 meters (default is 1000 meters).

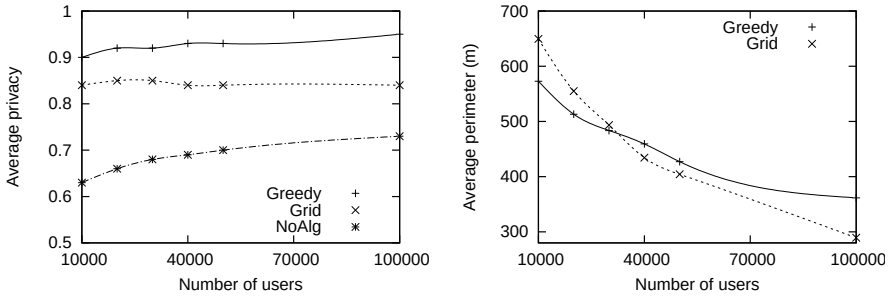
In our experimental results we also evaluated the privacy threat when no privacy preserving algorithm is applied. The label *NoAlg* is used in the figures to identify results in this particular case.

4.4 Impact of Simulation Parameters in the Evaluation of the Generalization Algorithms

The objective of the first set of experimental results we present is to show which parameters of the simulation affect most the evaluation of the generalization algorithms. In these tests we used the *AB* dataset only.

Figure 2(a) shows that the average privacy obtained with *Greedy* and *Grid* is not significantly affected by the size of the total population. Indeed, both algorithms, independently from the total number of users, try to have generalized regions that cover the location of k users, so the privacy of the requests is not affected. However, when the density is high, the two algorithms can generalize to a small area, while when the density is low, a larger area is necessary to cover the location of k users (see Figure 2(b)). On the contrary, the privacy obtained when no generalization is performed is significantly affected by the total population. Indeed, a higher density increases the probability of different users to be in the same location and hence it increases privacy also if the requests are not generalized.

A parameter that significantly affects the average privacy is the probability of identification of a user in a certain place. In Figure 3 we show the experimental results for different values of P_{id-in} when, in each test, P_{id-out} is set to



(a) Average privacy. (b) Average perimeter.

Fig. 2. Performance evaluation for different values of the total population.

$P_{id-in}/10$. As expected, considering a trace of requests, the higher is the probability of identifying users in one or more of the regions from which the requests in the trace were performed, the smaller is the level of privacy.

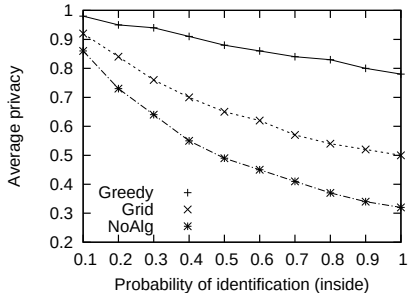
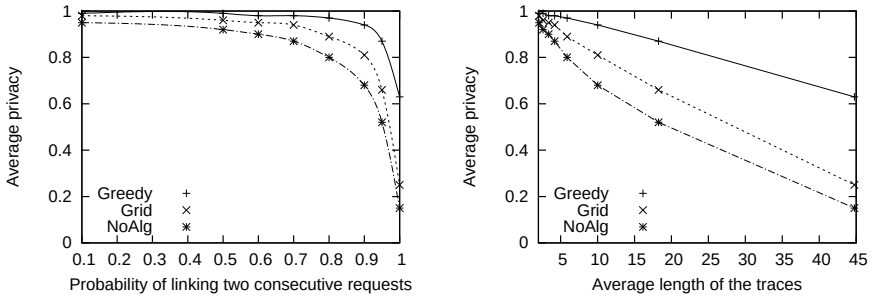


Fig. 3. Average privacy for different values of P_{id-in} ($P_{id-out} = P_{id-in}/10$).

Figure 4(a) shows the impact of P_{link} on the average privacy. As expected, high values of P_{link} lead to small values of privacy. Our results show that the relation between the P_{link} and privacy is not linear. Indeed, privacy depends almost linearly on the average length of the traces identified by the adversary (Figure 4(b)). However, the average length of the traces grows almost exponentially with the value of P_{link} (Figure 5).

To summarize the first set of experiments, our findings show that many parameters of the simulation significantly affect the evaluation of the generalization algorithms. This implies that when a generalization algorithm is evaluated it is necessary to carefully estimate realistic values for the parameters of the simulation. Indeed, an error in the estimation may lead to misleading results.



(a) Average privacy as a function of P_{link}. (b) Average privacy as a function of the average trace length.

Fig. 4. Performance evaluation for different values of P_{link}.

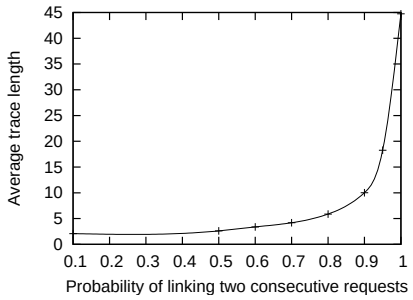


Fig. 5. Average trace length as a function of P_{link}.

4.5 Impact of the User Movements on the Evaluation of the Generalization Algorithms

The objective of the second set of experiments is to answer an important question posed in this paper: what is the impact of the different simulated user movements on the evaluation of the Generalization Algorithms? We answer to this question with a set of tests performed on the two different datasets we obtained as described above.

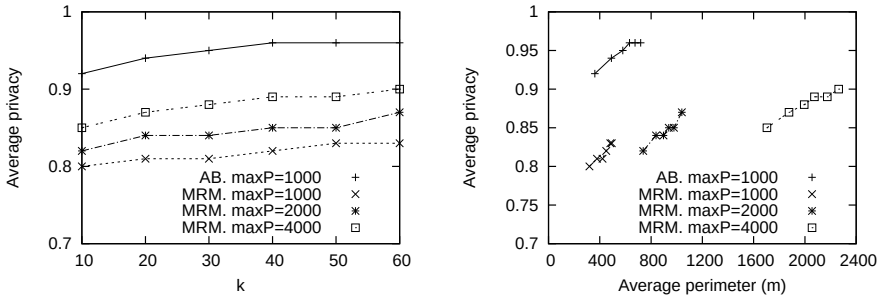
The first set of tests, reported in in Figure 6, compares the privacy achieved by the Greedy algorithm on the two datasets for different values of k and for different values of QoS. The experiments on MRM were repeated trying also larger values for the QoS threshold ($maxP = 2000$ and $maxP = 4000$), so three different versions of MRM appear in the figures. In order to focus on these parameters only, in these tests the probability of identification was set to the same value for any place ($P_{id-in} = P_{id-out} = 0.1$), and for the MRM dataset the issuer of the requests was randomly chosen only among those that stay in the simulation for 3 hours, ignoring the ones staying for much shorter time that inevitably are part of this dataset. This setting allowed us to compare the

results on the two datasets using the same average length of traces identified by the adversary.

Figure 6(a) shows that the average privacy of the algorithm evaluated on the AB dataset is much higher than on the MRM dataset. This is mainly motivated by the fact that in AB users tend to concentrate in a few locations (the entertainment places) and this enhances privacy. This is also confirmed by a similar test performed without using any generalization of locations; we obtained values constantly higher for the AB dataset (the average privacy is 0.67 in AB and 0.55 in MRM).

In Figure 6(b) we show the QoS achieved by the algorithm in the two datasets with respect to the average privacy achieved. This result confirms that the level of privacy evaluated on the AB dataset using small values of k and $maxP$ for the algorithm cannot be observed on the MRM dataset even with much higher values for these parameters.

From the experiments shown in Figure 6 we can conclude that if the MRM dataset is used as a benchmark to estimate the values of k and $maxP$ that are necessary to provide a desired average level of privacy, then the results will suggest the use of values that are over-protective. As a consequence, it is possible that the service will exhibit a much lower QoS than the one that could be achieved with the same algorithm.



(a) Average privacy as a function of k.

(b) Average privacy as a function of the average perimeter.

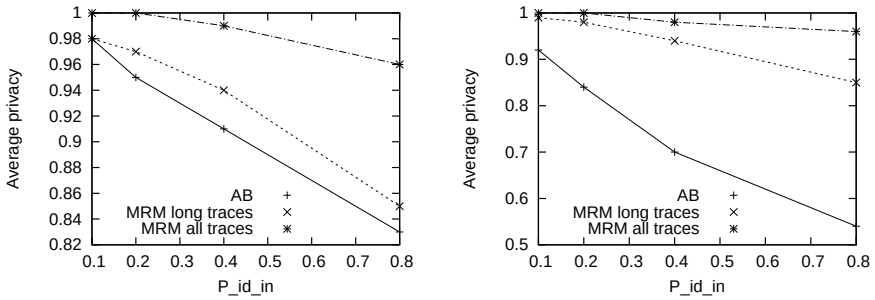
Fig. 6. Evaluation of the *Greedy* algorithm using AB and MRM data sets. $P_{id-in} = P_{id-out} = 0.1$

The above results may still support the safety of using MRM, since according to what we have seen above a technique achieving a certain level of privacy may only do better in a real scenario. However, our second set of experiments shows that this is not the case.

In Figure 7 we show the results we obtained by varying the probability of identification. For this test, we considered two sets of issuers in the MRM data set. One set is composed by users that stay in the simulation for 3 hours, (MRM *long traces*, in Figure 7), while the other contains issuers randomly chosen in the

entire set of users (*MRM all traces*, in Figure 7), hence including users staying in the simulation for a much shorter time.

In Figure 7(a) and 7(b) we can observe that the execution on the MRM dataset leads to evaluate a privacy level that is higher than the one obtained on the AB dataset. In particular, the evaluation of the *Grid* algorithm using the MRM dataset (Figure 7(b)), would suggest that the algorithm is able to provide a high privacy protection. However, when evaluating the same algorithm using the more realistic dataset AB, this conclusion seems to be incorrect. In this case, the evaluation on the MRM dataset may lead to underestimate the privacy risk, and hence to deploy services based on generalization algorithms that may not provide the minimum required level of privacy.



(a) *Greedy* algorithm. (b) *Grid* algorithm.

Fig. 7. Average privacy using AB and MRM data sets. $P_{id-out} = P_{id-in}/10$.

5 Conclusions and open issues

In this paper we claim that the experimental evaluation of LBS privacy preserving techniques should be based on user movement datasets obtained through simulations tailored to the specific deployment scenario of the target services. Our results support our thesis for the class of LBS known as friend-finder services, for techniques based on spatial cloaking, and for adversary models that include the possibility for the adversary to occasionally recognize people in certain locations. We believe that these results can be generalized to other LBS, techniques and adversary models. For example, as a future work, it would be interesting to also evaluate some defense techniques that generalize the issuer's location to an area that does not necessarily contain the issuer's location. Moreover, in our experiments we only considered the first of the two privacy threats presented in the introduction. We do have some ideas on how to extend them to consider the second, location privacy, as well. Finally, we believe a significant effort should be devoted to the development of new flexible and efficient context-aware user movement simulators, as well as to the collection of real

data, possibly even in an aggregated form, to properly tune the simulations. In our opinion this is a necessary step to have significant common benchmarks to evaluate LBS privacy preserving techniques.

Acknowledgments

The authors would like to thank Stefano Varesi for his contribution in writing the code that was used for our simulations. This work was partially supported by National Science Foundation (NSF) under grant N. CNS-0716567, and by Italian MIUR under grant InterLink II04C0EC1D.

References

1. Barkhuus, L., Dey, A.: Location-based services for mobile telephony: a study of users privacy concerns. In: Proc. of the 9th International Conference on Human-Computer Interaction, IOS Press (2003) 709–712
2. Bettini, C., Mascetti, S., Wang, X.S., Jajodia, S.: Anonymity in location-based services: towards a general framework. In: Proc. of the 8th International Conference on Mobile Data Management, IEEE Computer Society (2007)
3. Brinkhoff, T.: A framework for generating network-based moving objects. GeoInformatica **6**(2) (2002) 153–180
4. Martin, M., Nurmi, P.: A generic large scale simulator for ubiquitous computing. In: 3rd Annual International Conference on Mobile and Ubiquitous Systems: Networking & Services, IEEE Computer Society (July 2006)
5. Gonzalez, M.C., Hidalgo, C.A., Barabasi, A.L.: Understanding individual human mobility patterns. Nature **453** (June 2008) 779–782
6. Ghinita, G., Kalnis, P., Khoshgozaran, A., Shahabi, C., Tan, K.L.: Private queries in location based services: Anonymizers are not necessary. In: Proc. of SIGMOD, ACM Press (2008)
7. Gruteser, M., Grunwald, D.: Anonymous usage of location-based services through spatial and temporal cloaking. In: Proc. of the 1st International Conference on Mobile Systems, Applications and Services (MobiSys), The USENIX Association (2003)
8. Mokbel, M.F., Chow, C.Y., Aref, W.G.: The new casper: query processing for location services without compromising privacy. In: Proc. of the 32nd International Conference on Very Large Data Bases, VLDB Endowment (2006) 763–774
9. Gedik, B., Liu, L.: Protecting location privacy with personalized k-anonymity: Architecture and algorithms. IEEE Transactions on Mobile Computing **7**(1) (2008) 1–18
10. Kalnis, P., Ghinita, G., Mouratidis, K., Papadias, D.: Preventing location-based identity inference in anonymous spatial queries. IEEE Transactions on Knowledge and Data Engineering **19**(12) (2007) 1719–1733
11. Kido, H., Yanagisawa, Y., Satoh, T.: Protection of location privacy using dummies for location-based services. In: Proc. of the 21st International Conference on Data Engineering Workshops, IEEE Computer Society (2005)
12. Yiu, M.L., Jensen, C.S., Huang, X., Lu, H.: Spacetwist: Managing the trade-offs among location privacy, query performance, and query accuracy in mobile services. In: Proc. of the 24th International Conference on Data Engineering, IEEE Computer Society (2008)

13. Ardagna, C.A., Cremonini, M., Damiani, E., di Vimercati, S.D.C., Samarati, P.: Location privacy protection through obfuscation-based techniques. In: Proc. of the 21st Conference on Data and Applications Security. Volume 4602 of Lecture Notes in Computer Science., Springer (2007) 47–60
14. Bettini, C., Wang, X.S., Jajodia, S.: Protecting privacy against location-based personal identification. In: Proc. of the 2nd workshop on Secure Data Management. Volume 3674 of LNCS., Springer (2005) 185–199
15. Xu, T., Cai, Y.: Location anonymity in continuous location-based services. In: Proc. of ACM International Symposium on Advances in Geographic Information Systems, ACM Press (2007)
16. Abul, O., Bonchi, F., Nanni, M.: Never walk alone: Uncertainty for anonymity in moving objects databases. In: Proc. of the 24th International Conference on Data Engineering, IEEE Computer Society (2008)
17. Terrovitis, M., Mamoulis, N.: Privacy preservation in the publication of trajectories. In: Proc. of the 9th International Conference on Mobile Data Management, IEEE Computer Society (2008)
18. Vyahhi, N., Bakiras, S., Kalnis, P., Ghinita, G.: Tracking moving objects in anonymized trajectories. In: Proc. of 19th International Conference on Database and Expert Systems Applications, Springer (2008, to Appear)
19. Mascetti, S., Bettini, C., Freni, D., Wang, X.S.: Spatial generalization algorithms for lbs privacy preservation. Journal of Location Based Services **2**(1) (2008)