Computer Communications 135 (2019) 16–27

Contents lists available at ScienceDirect

Computer Communications

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

Reversible spatio-temporal perturbation for protecting location privacy

Chao Li *, Balaji Palanisamy

School of Computing and Information, University of Pittsburgh, USA

ARTICLE INFO

Keywords:
Location privacy
Multilevel privacy
Reversible cloaking algorithm
Spatio-temporal cloaking algorithm
k-anonymity
Road network

ABSTRACT

Ubiquitous deployment of low-cost mobile positioning devices and widespread use of high-speed wireless networks have resulted in a rapid growth of location-based applications. While location-based services provide a wide range of life enhancing experiences to users, the exposure of location information poses significant privacy risks that can invade the users' location privacy. Location privacy risks can be mitigated using location anonymization techniques that perturb the raw location of users to make the location indistinguishable from that of a set of other users. A fundamental limitation of traditional location anonymization techniques is that they are developed as unidirectional techniques that fail to support multi-level control of access to location data when data users have different access privileges on the exposed location information. As a result, location information once perturbed cannot be reduced in terms of anonymity or degree of perturbation even when some data users have access to fine granular information in the exposed data. Recent techniques on reversible spatial cloaking techniques employ data anonymization keys to perturb a user's location in a pseudo-random manner such that the anonymized location information can be de-anonymized later using the anonymization keys. While reversible spatial cloaking provides support for multi-level location privacy, their performance is limited by their adopted spatial cloaking model in which the location perturbation occurs solely in the spatial domain without considering the temporal domain. Hence, reversible spatial location cloaking techniques obtain lower success rate and lower spatial resolution of the perturbed location leading to unreliable anonymization and lower service quality. This paper presents a new suite of reversible cloaking techniques that reversibly perturb location information of users using a spatio-temporal cloaking model, allowing data perturbation to occur along both spatial and temporal dimensions while still ensuring that the spatio-temporal expansion process is reversible when suitable access keys are provided. The proposed model achieves higher success rate and higher spatial resolution compared to reversible spatial cloaking. We compare our techniques through extensive experiments on real road networks. The results show that our techniques offer better QoS performance than the existing approaches and demonstrate strong attack resilience against adversarial attacks.

1. Introduction

Wide-spread availability of high-bandwidth wireless networks and the proliferation of GPS supported mobile devices have rapidly increased the demands for location-based service applications. In the big data era, the user experience of location-based service applications is widely enhanced through novel combinations of both location and contextual information of users from multiple data sources leading to more personalized and more customizable services than ever. Examples include personalized navigation *("it is not easy to drive through this road based on user's past driving behavior")*, weather forecast *("the rain at your current location will stop exactly after 11 min".)* and location-based social networking *("you have one friend currently in the same restaurant")*. According to recent surveys [1,2], nearly 68% of the US population own and use a smartphone and roughly 90% of them use location-based service applications, implying that on an average over six out of ten people use services that require location information. While location-based services find numerous potential benefits, they also open new doors for privacy threats. The exposure of private location information can have many undesirable effects ranging from receiving unwanted location-related advertisements and spam to experiencing even life-threatening events leading to physical attacks [3]. With the advent of big data and big data analytics, the risk of disclosing location information is further exacerbated as an adversary can correlate the exposed location with information from various other data sources to infer more accurate and fine-grained information about individuals [4].

The risks of disclosing private location information can be reduced by using location anonymization techniques. Location anonymization refers to the process of perturbing user location information such that the perturbed information becomes indistinguishable from that of a set of other users. A user is considered to be location *k*-anonymous if her

* Corresponding author.
E-mail addresses: chl205@pitt.edu (C. Li), bpalan@pitt.edu (B. Palanisamy).

https://doi.org/10.1016/j.comcom.2018.12.003
Received 5 July 2018; Received in revised form 20 November 2018; Accepted 14 December 2018
Available online 18 December 2018

location information is indistinguishable from the location information of at least $k-1$ other users in a spatial or spatio-temporal space. As an extension to the location k-anonymization model, location l-diversity [5] and segment l-diversity [6] constraints have been proposed to further strengthen the privacy offered by these solutions. Several location perturbation techniques based on anonymization and differential privacy [7–15] have been proposed in the literature to tackle the location privacy problem. However, a fundamental limitation of these existing location privacy protection schemes is that location information once perturbed to provide a certain anonymity level cannot be reversed to reduce anonymity or the degree of perturbation, which means different location data users have no choice but to get the perturbed location information with same privacy level. In such a scenario, the owners of the location information lose the multi-level privacy control of their data requiring even privileged data users to access information at lower accuracy and granularity than what they are entitled to. For instance if Alice is concerned about her location privacy, she might decide to expose her location with a certain privacy level at a location-based social network [16]. However, she may wish to give her friends access to a reduced anonymity level as she may trust them more than the others. Also, Alice may want to give access to her exact location information to some of her close friends who are most trustworthy.

Recent techniques [17] on reversible spatial cloaking techniques employ data anonymization keys to perturb a user's location in a pseudo-random manner such that the anonymized location information can be de-anonymized later using the anonymization keys. While reversible spatial cloaking provides support for multi-level location privacy, their performance is limited by their adopted spatial cloaking model in which the location perturbation occurs solely in the spatial domain without considering the temporal domain. Hence, reversible spatial location cloaking techniques obtain lower success rate and lower spatial resolution of the perturbed location leading to unreliable anonymization and lower service quality. This paper presents a new suite of reversible cloaking techniques that reversibly perturb location information of users using a spatio-temporal cloaking model, allowing data perturbation to occur along both spatial and temporal dimensions while still ensuring that the spatio-temporal expansion process is reversible when suitable access keys are provided. The proposed model achieves higher success rate and higher spatial resolution compared to reversible spatial cloaking. We compare our techniques through extensive experiments on real road networks that show that our techniques offer better QoS performance than the existing approaches and demonstrate strong attack resilience against adversarial attacks.

The rest of the paper is organized as follows: Section 2 provides a background and an overview of the multi-level reversible location anonymization problem. In Section 3, we discuss two reversible spatio-temporal cloaking schemes that support multi-level location privacy, namely time-first reversible spatio-temporal cloaking scheme and space-first reversible spatio-temporal cloaking scheme. In Section 4, we present the analysis of our experiments on realistic road network traces generated using GTMobiSim. We discuss related work in Section 5 and we conclude in Section 6.

2. Overview of concepts and models

In this section, we first present the location anonymity models used in our work and describe the composition of a user-defined privacy profile that captures customized privacy requirements of the users. We then introduce the spatio-temporal cloaking model used in our work that allows data perturbation to occur along both spatial and temporal dimensions. We discuss the proposed class of reversible cloaking schemes that can leverage the spatio-temporal cloaking model to support the multi-level location privacy requirements of users while ensuring high service quality. Finally, we discuss the attack models used for evaluating the attack resilience of the proposed schemes.

2.1. Location anonymity models

In this paper, We use two anonymity models, namely *location k-anonymity* and *segment l-diversity* for protecting the location privacy of users. They are defined as follows.

Definition 1 (*Location k-anonymization*)**.** The location information of a user is said to be k-anonymous if the location information is indistinguishable from the location information of at least k-1 other users.

Definition 2 (*Segment l-diversity*)**.** The location information of a user is said to be segment l-diverse if the exposed location contains at least l well-represented road segments.

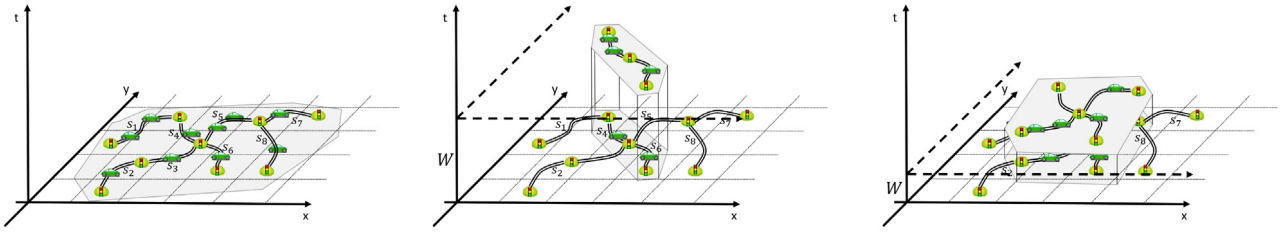
The *location k-anonymity* requirement ensures that the exposed location of a user is indistinguishable from a set of other users on the road network. However, satisfying *location k-anonymity* alone may not be sufficient to protect the location privacy of the user in cases when there are homogeneity attacks [5]. For instance, if all the k users contained in a *k-anonymized* spatial region are present in a single physical location, such as a hospital, then even though there are k users in the cloaked region, an adversary observing the region can still infer the actual location of the subject with high certainty. To protect against such scenarios, the notion of *location l-diversity* has been introduced [5,18]. A cloaked location satisfies *segment l-diversity* [6] if the cloaked region not only includes k distinct users but also contains l well represented road segments. Therefore, from an attacker's perspective, a cloaking area with more segments increases the difficulty to track a user and hence ensuring a larger l-diversity provides higher location privacy.

2.2. User-defined privacy profile

In practice, each anonymization request should include a user-defined privacy profile, which indicates customized anonymization requirements desired by the user. A user specifies a pair of values, denoted as (δ_k, δ_l), as her desired anonymization level, where δ_k and δ_l indicates the numbers of users and segments that she would like the cloaking region to contain. Besides δ_k and δ_l, the privacy profile also includes two QoS-related parameters, namely the spatial tolerance σ_s and temporal tolerance σ_t, which indicate the maximum acceptable cloaking spatial area and the maximum time delay to receive the response [6,10,14]. Therefore, a complete user-defined privacy profile contains four parameters, represented as $(\delta_k, \delta_l, \sigma_s, \sigma_t)$.

2.3. Spatio-temporal cloaking model

To satisfy the k-anonymity requirement δ_k, a cloaking algorithm can include mobile users by expanding the cloaking region geographically and/or extending the waiting time window, thus resulting in three cloaking models, namely *spatial cloaking*, *temporal cloaking* and their combination *spatio-temporal cloaking*. Fig. 1 explains the relationship and differences among the three models. Suppose a user sends an anonymization request with $(\delta_k = 10, \delta_l = 2)$ at t_0, then the cloaking region should contain at least ten mobile users and two segments. Fig. 1(a) shows the result of *spatial cloaking*, which expands the cloaking region as a two-dimensional spatial area to include nine other mobile users to satisfy $(\delta_k = 10, \delta_l = 2)$. In this example, the result includes eight segments from s_1 to s_8, which are restricted by the σ_s. In Fig. 1(b), unlike *spatial cloaking*, the *temporal cloaking* includes nine other mobile users by extending the waiting time window W along the time axis so that the other nine mobile users passing the minimum number of segment requested by $\delta_l = 2$ are included in the cloaking region to satisfy $\delta_k = 10$. In the example, the *temporal cloaking* results in a cloaking region containing only two segments s_4 and s_6 while the mobile users are accumulated through the waiting time window W restricted by the σ_t. Alternatively, as shown in Fig. 1(c), the *spatio-temporal cloaking* expands in all the x, y and t axes so that a three-dimensional box is formed



(a) Spatial cloaking (b) Temporal cloaking (c) Spatio-temporal cloaking

Fig. 1. Cloaking models.

to include the required number of mobile users. To achieve a certain δ_k, the cloaking spatial area generated by the *spatio-temporal cloaking* is smaller than the one generated by *spatial cloaking* while larger than the one generated by *temporal cloaking*. Meanwhile, the waiting time window W required by *spatio-temporal cloaking* is between the ones required by the other two models. Therefore, we can see that there is a tradeoff between the cloaking spatial area and the waiting time window W regarding δ_k and the *spatio-temporal cloaking* is a general model to handle this tradeoff. In addition, we may treat *spatial cloaking* and *temporal cloaking* as two special cases of *spatio-temporal cloaking*. In practice, by properly leveraging this tradeoff, *spatio-temporal cloaking* can usually achieve better results. Consider an anonymization request with $(\delta_k, \delta_l, \sigma_s, \sigma_t)$, to satisfy δ_k, *spatial cloaking* may easily lead to a cloaking region larger than σ_s while *temporal cloaking* may also lead to a response delay longer than σ_t. In contrast, *spatio-temporal cloaking* can make full use of available resources within the QoS requirements (σ_s, σ_t) to satisfy (δ_k, δ_l).

2.4. Multilevel location privacy management system

Our work aims at developing techniques for supporting a multilevel location privacy management system (Fig. 2) for location-based services (LBS), which allows a LBS user to expose location information with different granularity to other users. In this system, mobile users first submit their real location information to a reversible cloaking process. Such a process can be operated by trusted LBS providers as a functional module supporting reversible and fine-grained location cloaking for their users. In case of untrusted LBS providers, the reversible cloaking process can be implemented using a trusted third party anonymizer [6,7,17]. Upon receiving the real location information, the reversible cloaking process then generates multiple secret keys and uses these keys to pseudo-randomly create multiple levels of cloaked location information from user's real location information. After that, the reversible cloaking process sends the secret keys to the mobile user and exposes the cloaked location information corresponding to the highest privacy level. From then on, privileged data users can request secret keys from the mobile user to reduce the privacy level of the exposed cloaked location information to access the information with higher accuracy. Compared with conventional location cloaking systems [6,7,10,19], the system in Fig. 2 offers several advantages. First, instead of an *all-or-nothing* access control, this system enables a multi-level access control that allows different groups of data users to access the location data with different levels of utility and privacy. Second, this system allows the data owner to manage the access to the location data using secret keys. The data owner can determine a set of access rules and make the locally stored secret keys be automatically sent to privileged data users based on the rules. Last but not the least, with the reversible cloaking algorithms, the system allows the multiple levels of cloaked location information to be derived from a single cloaked location information exposed at the LBS provider through the access keys and thus minimizes the storage overhead for maintaining multiple versions of the cloaked location information.

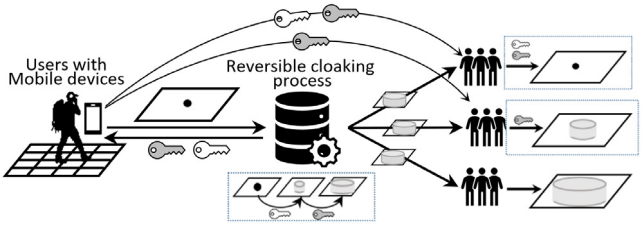


Fig. 2. Multilevel location privacy management system.

2.5. Reversible spatio-temporal cloaking

The performance of the multilevel location privacy management system highly depends on the adopted reversible cloaking techniques. In [17], a class of reversible cloaking techniques was designed based on the spatial cloaking model. However, as we discussed in Section 2.3, the performance of such reversible spatial cloaking techniques is limited by the spatial cloaking model. In this paper, we develop a new set of reversible cloaking techniques based on the more advanced spatio-temporal cloaking model, which can efficiently support the multi-level location privacy management system with higher success rate and higher spatial resolution while ensuring higher service quality.

An example of the reversible spatio-temporal cloaking process with N privacy levels is shown in Fig. 3. In the example, the user-defined privacy profile can be denoted as $UDPP = \{(\delta_k^i, \delta_l^i, \sigma_s^i, \sigma_t^i) | 1 \le i \le N-1\}$, with $UDPP^i = (\delta_k^i, \delta_l^i, \sigma_s^i, \sigma_t^i)$ representing the profile for a specific privacy level L^i. Specifically, we define privacy level L^0 as the cloaking box that is actually the snapshot of the segment of the actual user at t_0. In addition, each privacy level, L^i is associated with a secret key, Key^i, which is used to drive the anonymization process for that privacy level. Therefore, with access to the anonymization key of a particular privacy level, users of the cloaked location data can selectively de-anonymize the cloaking box to reduce privacy levels and obtain finer location information. A detailed example of a four level case is shown in Fig. 3. The segment s_4 contains the actual user, so its snapshot at query time t_0 forms the cloaking box of level, L^0. Using the key Key^1 to reach the privacy level, δ_k^1, δ_l^1 of L^1, the cloaking box is expanded along spatial axes by including s_6 while at the same along the time axis from t_0 to a time window W_1. Then, Key^2 is used further to expand the cloaking box to meet δ_k^2, δ_l^2 of level L^2 by adding segments $\{s_3, s_5\}$ and extending W_1 to W_2. Finally, $\{s_1, s_2, s_7, s_8\}$ are added and W_2 is extended to W_3 by using the key, Key^3 to reach the highest privacy level, L^3.

Later, when the cloaked location information needs to be reduced in privacy levels, it can be done using the secret keys. For instance, for accessing the information at the lower privilege level, L^2, Key^3 can be used to exactly identify and remove the segments $\{s_1, s_2, s_7, s_8\}$ from the spatial cloaking region and also shrink window W_3 to W_2 so that the cloaking box of level L^2 can be restored. Similarly, using both Key^3 and Key^2, the segments $\{s_1, s_2, s_7, s_8, s_3, s_5\}$ can be removed and W_2 can be reduced to W_1, which result in the cloaking box of level L^1. Therefore, by merely managing the secret keys among the location data users at

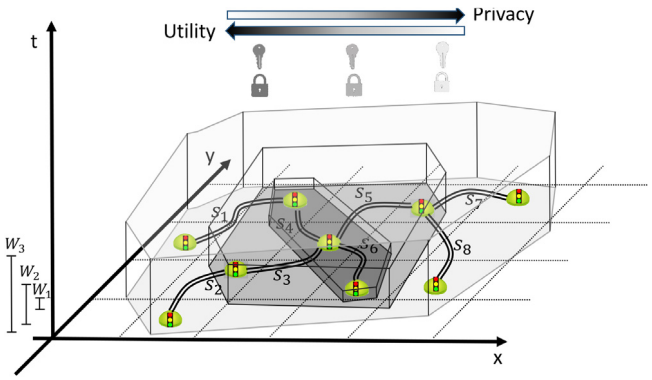


Fig. 3. Reversible spatio-temporal cloaking.

different privilege levels, the whole process protects location privacy under multiple discrete levels as customized in the user-defined privacy profile.

2.6. Attack model

An anonymization scheme is considered to be strong if it makes it harder for the attackers to infer the original location information from the anonymous cloaking area. However, when it is extended to a multilevel framework, the ability of the attackers to correctly reduce the privacy level should also be taken into account. Given that an adversary at one privilege level may attempt to access the finer information entitled to users of a higher privilege level, we need to ensure that even if the attacker has complete knowledge of the reversible spatio-temporal cloaking algorithm, no additional information can be inferred without the access to the secret anonymization keys. In this paper, we introduce two attack models: (i) replay attack and (ii) network-distance attack to evaluate the attack-resilience of the proposed multi-level spatio-temporal cloaking schemes. Similar to the adversary models in [6,7,14], we primarily focus on snapshot exposure of location information for supporting snapshot location-based queries. For continuous location-based queries, with the additional ability to combine and correlate information from the location exposure of multiple snapshot instances, the adversary's chances of inferring the true location can be increased [20,21]. While addressing such query-correlation attacks is a promising direction for future work, the scope of the replay attack model considered in our work is limited to snapshot queries.

2.6.1. Replay attack

In the replay attack, each segment within the cloaking region is iteratively considered to be the segment of the actual user and the associativity for all the segments are calculated by simulating the cloaking algorithm from this assumed start segment. If the number of segments shared by the replayed cloaking region generated from a segment, s_i, and the real cloaking region generated from the real start segment is N_i. The associativity, A_i of s_i, can be calculated as $A_i = \frac{N_i}{\sum N_i}$. After obtaining A_i for all the segments within the cloaking region, the uncertainty of the attacker can be quantified by Entropy [22] measured as $E = -\sum A_i \log A_i$. The Entropy is a measure of the amount of information required to break the anonymity provided by the system. Therefore, the larger the entropy, the higher is the uncertainty of the attacker and the scheme is more attack-resilient. The purpose of the replay attack is to infer the location of the actual user in terms of the segment where the user is located. However, in a multi-level location privacy model, the purpose of the attacker may be just to infer finer location information corresponding to a lower privacy level. Therefore, even if the cloaking algorithm provides high resilience to replay attacks, it may not be safe under attacks that target at just reducing the privacy levels as opposed to exactly inferring the actual user's location. Next, we introduce the network-distance attack that aims at reducing the anonymity level of the exposed location based on network distance information in the exposed cloaked location.

2.6.2. Network distance attack

In the network-distance attack, the attacker's goal is to identify which privacy level each segment in the cloaking region belongs to. This attack can be effective because many road-network based cloaking algorithms expand the cloaking region by adding new segments adjacent to the current cloaking region. Therefore, the inference attack becomes very effective when the cloaking algorithm leaves the actual user location close to the center of the cloaking area. In other words, with higher confidence, the attacker can guess that the segments far away from the center of the cloaking region belong to the higher privacy levels as those segments are likely to be added near the end of the cloaking process.

Precisely, in the network-distance attack, given a cloaking region, the attacker first computes the distance between each segment of the cloaked region and the center of cloaking area. Let the set of segments within the cloaking region be represented as $C = \{cs_1, cs_2, \ldots, cs_n\}$. The distance, $d(cs_i, cs_j)$ between two segments cs_i, cs_j in the cloaked region is defined as the distance between their midpoints along their road segments and the network distance of cs_i indicates the distance between cs_i and the center of the cloaking area. It is computed as $nd_i = \frac{\sum_{j=1}^{n} d(cs_i, cs_j)}{n-1}$. The attacker then estimates the likelihoods of a segment belonging to a privacy level by assigning higher likelihoods to segments with higher network distance to the higher privacy levels. Based on this information, the attacker can guess the privacy level a particular segment belongs to. Therefore, in order for a location cloaking scheme to be resilient to this attack, every segment added to the cloaked region should be equally probable to be located within the cloaking area. In other words, the probability distribution of the network distance of any segment added to the cloaked region should follow a uniform distribution, thus maximizing the uncertainty of the attacker in this attack.

In the next section, we present our proposed reversible spatio-temporal location cloaking mechanisms that support multi-level location privacy over road networks.

3. Reversible spatio-temporal cloaking

A successful spatio-temporal cloaking should satisfy both σ_s and σ_t. In general, it is hard to determine which of the many possible cloaking boxes is the optimal one as it should be determined in a customized manner. For the type of LBS requests that users prefer more accurate response using a smaller cloaking area, the spatio-temporal cloaking box with the smallest bottom area while the highest height would be the best choice. In contrast, for the type of LBS requests that users prefer shorter response delay, the spatio-temporal cloaking box with the shortest height while the largest bottom area offers the best performance. In this section, we present two different multi-level reversible spatio-temporal cloaking techniques satisfying the two types of LBS requests, respectively. We first present time-first reversible spatio-temporal cloaking (TF-RSTC), which aims at generating the spatio-temporal cloaking boxes with the smallest spatial cloaking area by first expanding itself along the time axis. We then present space-first reversible spatio-temporal cloaking (SF-RSTC), which results in the spatio-temporal cloaking boxes with shortest response delay by first expanding along the spatial axes.

3.1. Time-first reversible spatio-temporal cloaking

Intuitively, to minimize the spatial cloaking area, the available time σ_t should be made full use of. Therefore, in time-first reversible spatio-temporal cloaking (TF-RSTC), the height of a cloaking box is directly set to the maximum allowable value, namely σ_t, to aggregate the maximum

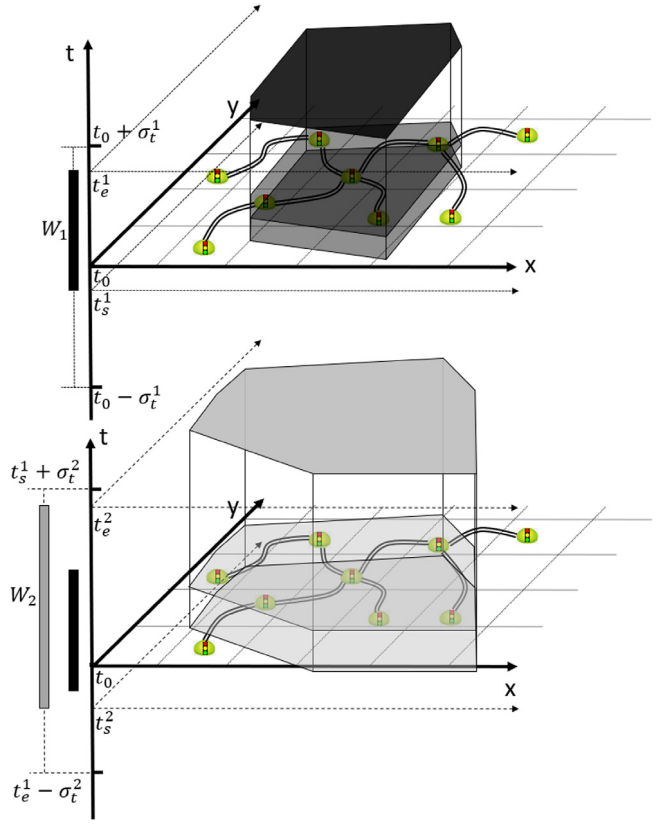


Fig. 4. Time-first reversible spatio-temporal cloaking.

number of mobile users for each segment, thus increasing the weight of each segment before a spatial cloaking algorithm is performed. Then, segments with boosted weights are gradually added to the spatial cloaking area to expand the cloaking box along x–y axes until (δ_k, δ_l) is satisfied.

There are two main challenges in this process. First, in the cloaking time window W that starts at t_s and ends at t_e, the position of location exposure time t_0 in the window should be chosen in a randomized manner, otherwise it may be exposed. For example, if we set $t_s = t_0$ and extend the window along one direction of time axis, the adversary who knows the information of W can easily infer t_0 because of the deterministic scheme and then leverage this vulnerability to locate the real user as one of the mobile users who were included into the cloaking box at t_0. Therefore, the location exposure time t_0 should be uniformly distributed in W to be perturbed. To achieve this, the cloaking time window W should be extended along two directions of the time axis. The positive extension corresponding to the future time should be bounded by temporal tolerance set by the service user, namely $|t_e - t_0| \leq \sigma_t$, to guarantee an acceptable delay of feedback. However, even though the negative extension pointing corresponding to the past has no influence on feedback delay, it should also be bounded. A very long negative extension may aggregate too many mobile users on each segment. In segment-based algorithms over road networks, each time one segment is added or removed, the change of δ_k will be very large in this case, which fails to control the anonymity level in a fine-grained manner. Therefore, we set $|W| = |t_e - t_s| = \sigma_t$ and $t_s \leq t_0 \leq t_e$ to bound both positive and negative extension by σ_t and also perturb t_0. In other words, the window W can be viewed as a sliding window with a fixed length σ_t, which is bounded by the range $[t_0 - \sigma_t, t_0 + \sigma_t]$. An example of such a window is the W_1 in Fig. 4. The real position of the window is pseudo-randomly determined by a secret key. We will discuss this later.

Algorithm 1: TF-RSTC

Input : Road network graph G, original segment s_u, original request time t_0, number of privacy levels N, secret keys $\{K_s^i | 1 \leq i \leq N-1\}$, user defined $\{(\delta_k^i, \delta_l^i, \sigma_s^i, \sigma_t^i) | 1 \leq i \leq N-1\}$.

Output: A cloaking area $CloakA^{N-1}$ and a set of users $CloakU^{N-1}$ for privacy level L^{N-1}.

1 Initially, $t_e^0 = t_0$, $\sigma_t^0 = 0$;
2 **for** $i = 1$ *to* $N-1$ **do**
3 $R = PseudoRandomNext(K_s^i)$;
4 $T = \sigma_t^i - \sigma_t^{i-1}$;
5 $t_s^i = t_e^{i-1} - \sigma_t^i + T * \frac{R \bmod 100}{100}$;
6 $t_e^i = t_s^i + \sigma_t^i$;
7 $W_i = [t_s^i, t_e^i]$;
8 **for** each segment s' satisfying $dist(s', s_u) \leq \sigma_s$ **do**
9 Weight s' based on W_i;
10 **end**
11 $\{CloakA^i, CloakU^i\} \leftarrow SEF$;
12 **end**

The second challenge arises due to the requirement of having multiple levels. In a multi-level scenario with N privacy levels, as discussed in Section 2.5, each privacy level L^i, except L^0, has a user-defined σ_t^i. Usually, a higher privacy level expecting larger δ_k is given longer σ_t for getting more mobile users. We now consider window $W_1 = [t_s^1, t_e^1]$ for a higher privacy level L^1 and window $W_2 = [t_s^2, t_e^2]$ for a lower privacy level L^2 and we assume $|W_1| < |W_2|$ based on the rule. Since both the two windows should include t_0, the position of the two windows may have two probabilities. In the first case, the smaller window W_1 is fully included by the larger window W_2, namely $t_s^2 \leq t_s^1 < t_e^1 \leq t_e^2$. One example of this fully included situation can be found in Fig. 4. In the second case, W_1 and W_2 are only intersected, which indicates either $t_s^2 \leq t_s^1 < t_e^2 \leq t_e^1$ or $t_s^1 \leq t_s^2 < t_e^1 \leq t_e^2$. However, we have to force the first probability to happen because the intersection case will make an adversary easily infer that t_0 locates in the intersection area, thus significantly compromising the cloaking result.

We propose the TF-RSTC algorithm that can properly handle both the challenges (Algorithm 1). The algorithm takes the road network graph data, user's original location information, multi-level number and a secret key and a profile $(\delta_k^i, \delta_l^i, \sigma_s^i, \sigma_t^i)$ for each level as inputs. For each privacy level (line 2–12), the algorithm takes three steps to pseudo-randomly expanding the spatio-temporal cloaking box in a reversible manner, namely time window expansion (line 3–7), segment weight increment (line 8–10) and finally spatial expansion (line 11). In both first and third step, the reversibility is supported by using the secret key as a seed of a pseudo-random number generator (line 3 and 11) and then leverage the created pseudo-random numbers to serve as providers of randomness during window expansion and segment selection. As a result, when later the same secret keys are used, the same randomness can be re-generated to narrow the window and remove segments, thus allowing the secret key holders to reduce the privacy level in a deterministic manner while preventing any party without the keys to do the same thing. During time window expansion, after generating a pseudo-random number, the algorithm first computes the expansion amount T, which indicates the amount of length increased from window W_{i-1} in the last round to window W_i in the current round. In TF-RSTC, as the goal is minimizing spatial cloaking area, the window length is always expanded to its maximum allowable amount, so T should be the difference of σ_t between two adjacent privacy levels. Once T has been set, a part of its amount is pseudo-randomly cut off through $T * \frac{R \bmod 100}{100}$ and then added to the lower bound of the range of the window in the current round, which then determines the start time t_s of the current window (line 5). After that, the end time t_e is simply the sum of t_s and σ_t (line 6) and the current window is finalized (line 7). To sum up, this time

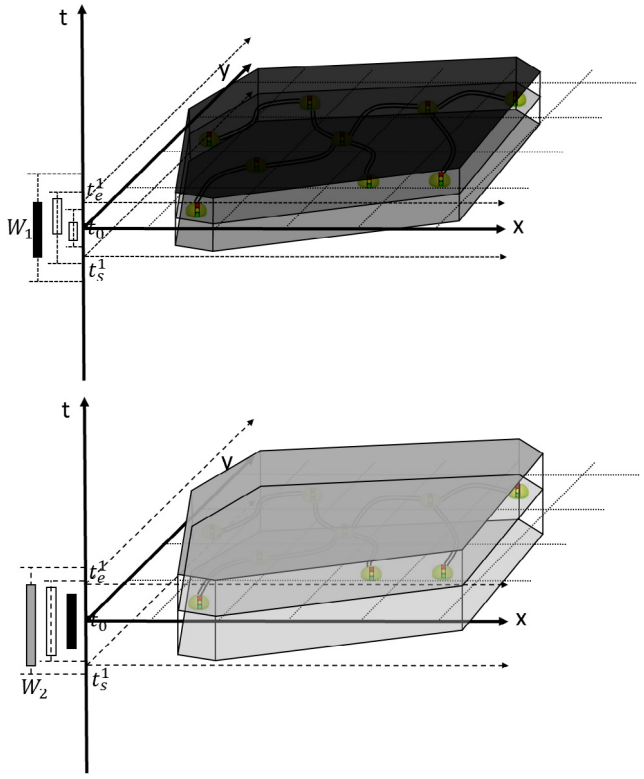


Fig. 5. Space-first reversible spatio-temporal cloaking.

window expansion process offers the following properties: (1) windows at all privacy levels are all with the length of their maximum allowable amounts; (2) a higher privacy level window is always longer than a lower privacy level window and it always covers all lower privacy level windows in full; (3) the multi-level window expansion is performed in a pseudo-random manner, which allows only the secret key holders to later narrow the window and de-anonymize the cloaking box along the time axis. After the time window of a particular privacy level has been decided, the mobile users passing the reachable segments from user's original location information within σ_s are accumulated to increase the weight of those segments. Finally, the spatial expansion function (SEF) can be used to expand the cloaking box along the spatial axes to form the cloaking box with $CloakA$ and $CloakU$ that satisfies the (δ_k, δ_l) of the current privacy level. It is worth noting that both RGE and RPLE proposed in [17] can be adapted to work as the spatial expansion function. We discuss more details about how RGE and RPLE can be adapted in Section 3.3.

3.2. Space-first reversible spatio-temporal cloaking

Unlike TF-RSTC, in the space-first reversible spatio-temporal cloaking (SF-RSTC), the spatio-temporal cloaking box first expands itself in the spatial domain along x–y axes to capture all available segments in the area bounded by σ_s. After that, the cloaking box gradually expands itself along the time axis by including more mobile users passing the same area at different timestamps until δ_k is satisfied. As a result, the SF-RSTC scheme enables shortest response delay at the price of the largest spatial cloaking area. Similar to TF-RSTC, the SF-RSTC algorithm should also solve the two challenges presented in Section 3.2. In addition, the two challenges should be handled in a way that the time window W gradually increases its length along the two directions of the time axis in a step-by-step manner. The reason for this requirement is to make the response delay as small as possible without violating the privacy protection.

To handle the challenges under the new circumstances, we propose the SF-RSTC algorithm as shown in Algorithm 2. In addition to the inputs taken by the TF-RSTC algorithm, the SF-RSTC algorithm requires a window extension rounds M, which indicates the number of times that a window spends to increase its length from zero to the maximum allowable amount. Intuitively, a larger M tends to results in a smaller window increment step and therefore a smaller window length is more likely to be found. In contrast, a smaller M may reduce the algorithm running time because of its coarse-grained search. In the algorithm, for each privacy level (line 2–23), the spatial expansion function is first run under the restriction that there is no room to expand the cloaking box along the time axis (line 3). In other words, the algorithm first gives it a try to see whether the maximum spatial cloaking area generated purely through the spatial expansion function can directly satisfy (δ_k, δ_l) (line 20–22). If not, the algorithm will start to gradually increase the time window W (line 4–19). Similar to the TF-RSTC algorithm, the secret key is used to provide pseudo-randomness (line 5). The algorithm will try M rounds (line 7–18) of window length increment, with each round increasing the window length by an amount $\triangle t$ (line 6). For example, in Fig. 5, the algorithm spent three rounds to set W_1 and two rounds to set W_2. During the jth round at the ith privacy level, the window with its current length $\sigma_t^i = \sigma_t^{i-1} + j * \triangle t$ is pseudo-randomly determined within the range $[t_e^{i-1} - \sigma_t^i, t_s^{i-1} + \sigma_t^i]$, thus guaranteeing that the current window can fully cover all lower privacy level windows (line 8–12). After the window has been set, the mobile users passing the spatial cloaking region generated through the first try (line 3) during the extended time window are accumulated (line 13). If the updated result can satisfy (δ_k, δ_l) for this privacy level, the algorithm will go to the next privacy level (line 14–17), otherwise the algorithm will further increase the time window with another $\triangle t$ in the next round until either (δ_k, δ_l) is satisfied or the window length has been increased for M rounds.

Algorithm 2: SF-RSTC

Input : Road network graph G, original segment s_u, original request time t_0, number of privacy levels N, secret keys $\{K_s^i | 1 \le i \le N-1\}$, user defined $\{(\delta_k^i, \delta_l^i, \sigma_s^i, \sigma_t^i) | 1 \le i \le N-1\}$, window extension rounds M.

Output: A cloaking area $CloakA^{N-1}$ and a set of users $CloakU^{N-1}$ for privacy level L^{N-1}.

1 Initially, $t_e^0 = t_0$, $\sigma_t^0 = 0$;
2 **for** $i = 1$ *to* $N-1$ **do**
3 $result \leftarrow SEF$;
4 **if** $result == FAIL$ **then**
5 $R = PseudoRandomNext(K_s^i)$;
6 $\triangle t = \frac{\sigma_t^i - \sigma_t^{i-1}}{M}$;
7 **for** $j = 1$ *to* M **do**
8 $\sigma_t^i = \sigma_t^{i-1} + j * \triangle t$;
9 $T = \sigma_t^i - \sigma_t^{i-1}$;
10 $t_s^i = t_e^{i-1} - \sigma_t^i + T * \frac{R \bmod 100}{100}$;
11 $t_e^i = t_s^i + \sigma_t^i$;
12 $W_i = [t_s^i, t_e^i]$;
13 Update $result$ based on W^i;
14 **if** $result == SUCCESS$ **then**
15 $\{CloakA^i, CloakU^i\} \leftarrow result$;
16 *break*;
17 **end**
18 **end**
19 **end**
20 **else**
21 $\{CloakA^i, CloakU^i\} \leftarrow result$;
22 **end**
23 **end**

3.3. Reversible spatial expansion

In both the TF-RSTC and SF-RSTC algorithms presented in the previous two subsections, we abstract the spatial expansion of the

cloaking box as an external function called at line 11 of Algorithm 1 and line 3 of Algorithm 2, respectively. To make the two algorithms fully reversible, similar to the way that window expansion can be pseudo-randomly controlled by secret keys, we also need to use the secret keys to pseudo-randomly select segments to expand the spatial cloaking region so that later the same keys can be applied by privileged data users to identify and remove these segments. In the rest of this subsection, we first present the high level idea of using keys to pseudo-randomly select segments to form spatial cloaking regions in a reversible manner. We then review the two approaches proposed in [17] for implementing reversible spatial cloaking, namely reversible global expansion (RGE) and reversible pre-assignment-based local expansion (RPLE). Finally, we discuss how RGE and RPLE techniques can be adapted and used as the spatial expansion function in TF-RSTC and SF-RSTC algorithms.

In reversible spatial cloaking, the anonymization and de-anonymization processes are considered as a continuous selection and removal of road segments on the geographic road map respectively. To ensure that the process is reversible, the segments are selected in a pseudo-random manner. Each road segment on the map is linked to several other segments, which are located close to it. Once a road segment S is selected during anonymization, the next selected road segment is from one of its linked segments. With a certain access key, a fixed segment S' among them is deterministically selected. However, without the access key, all its linked segments would have the same probability to be selected, thus making the selection process pseudo-random and making it impossible to reverse without possessing the access key. Then, during the de-anonymization process, the newly selected segment S' maps to the previous road segment S using the access key. The algorithms checks which road segment is linked with S' to narrow down the options and whether segment S' can be deterministically selected with the access key if we assume a segment is S. A key challenge here is the 'collision' issue that could happen in the de-anonymization process. That is, we may find multiple road segments that meet the conditions to be the candidate of the previously chosen road segment. To address this issue, in RGE, for each road segment selection during anonymization, the links of previously selected segments are rebuilt on the fly to avoid collisions and optimize the selection based on the current state. In RPLE, prior to the anonymization process, all the road segments in the map are pre-assigned their links in a collision-free manner. As a result, RGE has larger anonymization runtime to build collision-free links on the fly but smaller memory requirement while RPLE has smaller anonymization runtime but requires larger memory space to store the collision-free links. Next, we review the process of RGE and RPLE with Figs. 6 and 7, respectively.

In both Figs. 6 and 7, the current cloaking region is $\{s_8, s_9, s_{11}\}$, where s_8 is the last selected segment, and the algorithms are selecting the next segment to be added into the cloaking region. In RGE (Fig. 6), the three selected segments $\{s_8, s_9, s_{11}\}$ and the same number of non-selected nearby segments $\{s_6, s_{10}, s_{14}\}$ are taken to form a 3×3 square matrix, where the cells are filled with 0–2 in a way that each row/column has no repeated value. Assume that the pseudo-random number R_i generated through the access key gives $R_i \ mod \ 3 = 2$, then s_{14} will be the next selected segment because only the cell $[s_8][s_{14}]$ has value 2 at row s_8. Later in de-anonymization, after removing s_{14}, the same matrix can be formed and the same access key can give $R_i \ mod \ 3 = 2$. By looking at column s_{14}, since only the cell $[s_8][s_{14}]$ has value 2, the algorithm understands that s_8 should be the next removed segment. In this way, the reversibility can be established in a collision-free manner. Unlike RGE, in RPLE (Fig. 7), prior to the anonymization process, the algorithm has generated one forward list and one backward list for each segment in the map. All the lists have the same length, which is six in the example. Assume that the pseudo-random number R_i gives $R_i \ mod \ 6 = 3$, then s_{14} will be the next selected segment because it is the third element in the forward list of s_8. Later in de-anonymization, since s_8 is also the third element in the backward list of s_{14}, with the same access key giving $R_i \ mod \ 6 = 3$, the algorithm is able to remove s_8 after s_{14}. As can be seen,

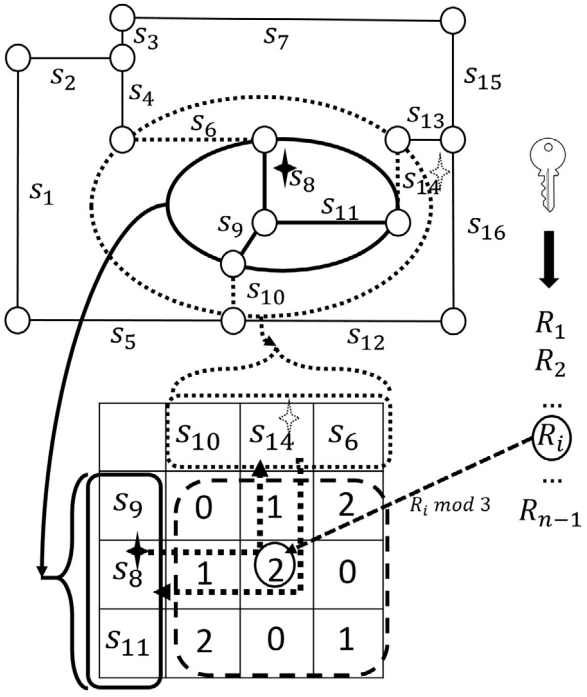


Fig. 6. Reversible global expansion.

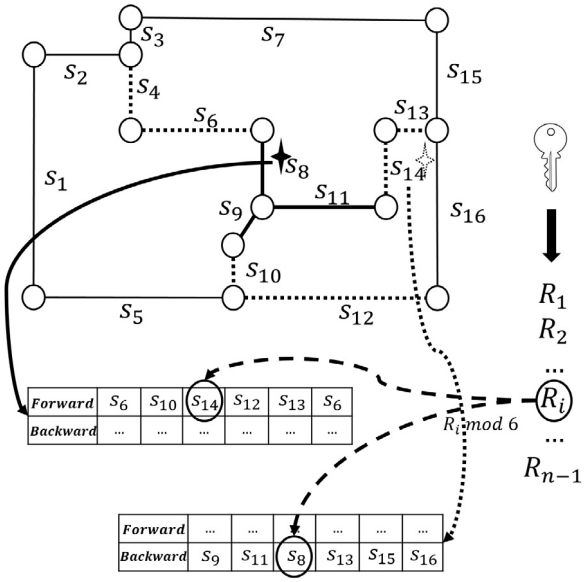


Fig. 7. Reversible pre-assignment-based local expansion.

to establish reversibility in RPLE, s_{14} should be at the same position in the forward list of s_8 where s_8 is located in the backward list of s_{14}. With this objective, in RPLE, the two lists for all the road segments can be generated in a greedy manner.

Both RGE and RPLE can be adopted as the spatial expansion function used in TF-RSTC and SF-RSTC algorithms. However, they must be adjusted in two aspects. First, in a spatio-temporal cloaking box, the same mobile user may appear at different segment within the box at different timestamps. To avoid counting such a mobile user for multiple times, when each new segment is selected to be added into the cloaking region, such collision should be first detected and the repeated mobile users should be removed. Second, in the original RGE and RPLE algorithms, the algorithm will stop when only δ_k is satisfied. However,

in the new context, the algorithm should stop only when both δ_k and δ_l are satisfied.

4. Experimental evaluation

In this section, we first present the experimental setup, and then evaluate the performance of proposed reversible spatio-temporal cloaking schemes.

4.1. Experimental setup

To simulate and compare different anonymization approaches, we use GTMobiSim mobile trace generator for road network [23]. Our experiments were designed based on a real road network map of northwest part of Atlanta, involving 6979 junctions and 9187 segments, obtained from maps of National Mapping Division of the USGS. There are 10,000 cars randomly generated along the roads based on Gaussian distribution. Once a car is generated, the associated destination is also randomly chosen and the route selection is based on shortest path routing. All the cloaking schemes are implemented in Java with the help of GTMobiSim. For all experiments, we repeated 100 times and took the average as results.

Our experimental evaluation consists of three parts. In the first part, we compare the performance of reversible spatio-temporal cloaking schemes proposed in this work with the existing reversible spatial cloaking schemes [17], namely reversible global expansion (RGE) and reversible pre-assignment-based local expansion (RPLE). In the rest of this section, when a spatial cloaking algorithm XYZ is applied as the spatial expansion function in time-first reversible spatio-temporal cloaking (TF-RSTC) (line 11 of Algorithm 1) or space-first reversible spatio-temporal cloaking (SF-RSTC) (line 3 of Algorithm 2), we refer to the corresponding spatio-temporal cloaking algorithms as TF-XYZ and SF-XYZ respectively. Since both RGE and RPLE can be applied in both TF-RSTC and SF-RSTC, we implement and compare all the possible combinations, namely TF-RGE, TF-RPLE, SF-RGE and SF-RPLE, with RGE and RPLE. Our results show that the reversible spatio-temporal cloaking schemes achieve higher success rate and higher spatial resolution compared to reversible spatial cloaking.

In the second set of experiments, we implement and compare the proposed reversible spatio-temporal cloaking schemes (TF-RGE, TF-RPLE, SF-RGE, SF-RPLE) with a set of irreversible spatio-temporal cloaking schemes. The irreversible spatio-temporal cloaking schemes are implemented by adopting two conventional spatial cloaking algorithm, namely Random Sampling (RS) and Star-based road network expansion (SE) [6] in TF-RSTC and SF-RSTC, which results in TF-RS, TF-SE, SF-RS and TF-SE schemes. The comparison of the eight algorithms shows that although the reversible spatio-temporal algorithms can offer the reversibility feature, their performance evaluated with different evaluation metrics are still as good as the irreversible ones without the reversibility feature, which demonstrates that the reversibility feature does not come at the cost of a performance drop.

Finally, in the third part of experiments, we compare and evaluate the attack resilience of the four reversible spatio-temporal cloaking schemes (TF-RGE, TF-RPLE, SF-RGE, SF-RPLE) against replay attack and network distance attack.

4.2. Reversible spatio-temporal cloaking and reversible spatial cloaking comparison

In the first part, we compare four reversible spatio-temporal cloaking schemes (TF-RGE, TF-RPLE, SF-RGE, SF-RPLE) with two reversible spatial cloaking schemes (RGE, RPLE) regarding two evaluation metrics. The first metric is relative spatial resolution (RSR), which is defined as the ratio of the size of the maximum allowable spatial area size specified by the spatial tolerance σ_s to the size of obtained cloaking area from algorithms. A larger RSR refers to a smaller cloaking area,

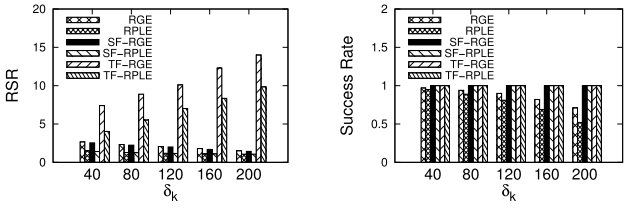


Fig. 8. Reversible spatio-temporal cloaking and reversible spatial cloaking comparison.

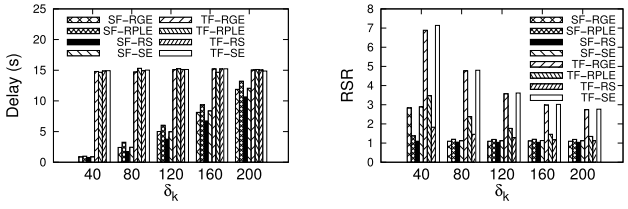


Fig. 9. Space-first and Time-first reversible spatio-temporal cloaking comparison.

which has a higher probability to provide more accurate LBS responses. The second metric is success rate, which simply refers to the ratio of the number of cloaking requests receiving successful responses that satisfy corresponding user-defined privacy profiles to the number of total cloaking requests.

The performance of the six algorithms are evaluated by varying the anonymity level δ_k as

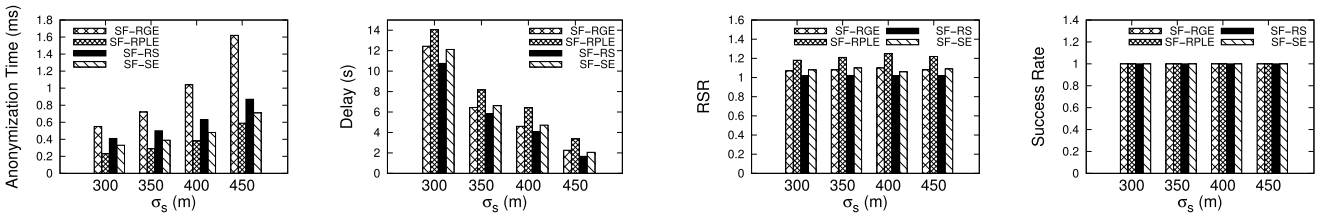
$$\delta_k = 10i \ for \ i = 1, 2...20$$

Also, the spatial tolerance, σ_s, is set as a function of the anonymity level, δ_k such that

$$\sigma_s = 400\sqrt{i} = \frac{40\delta_k}{\sqrt{i}} \ for \ i = 1, 2...20$$

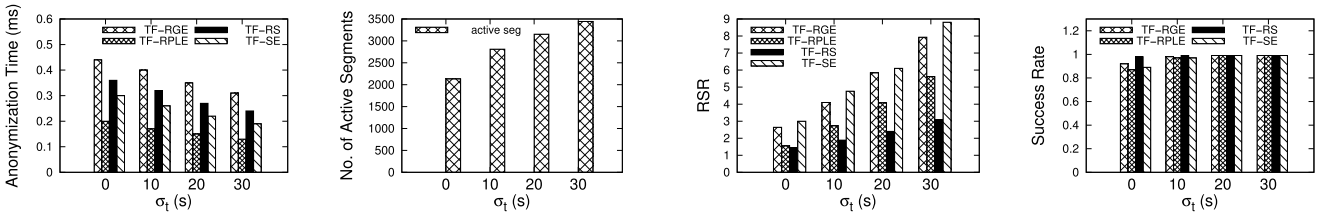
where the unit is meter(m). Therefore, the maximum allowable special region is a circular region with the user's actual location as the center and the spatial tolerance, σ_s as the radius. We also set 5% standard deviation for each σ_s and the segment diversity level δ_l is fixed to be 10. In addition, the temporal tolerance σ_t is fixed to 30 s for spatio-temporal algorithms.

The results of RSR with varying δ_k are shown in Fig. 8(a), where we observe three points. First, compared with RGE and RPLE, their implementation in space-first reversible spatio-temporal cloaking (SF-RSTC), namely SF-RGE and SF-RPLE, perform lower RSR. In contrast, the implementation of RGE and RPLE in time-first reversible spatio-temporal cloaking (TF-RSTC), namely TF-RGE and TF-RPLE results in much higher RSR. The reason is that SF-RSTC schemes first expand cloaking boxes in the spatial domain, which may quickly make RSR close to 1, namely the situation that the obtained cloaking area is close to the maximum allowable area. Theoretically, RSR offered by SF-RGE and SF-RPLE should be similar to that offered by RGE and RPLE. However, when cloaking region has been expanded to the maximum allowable area while σ_k is still not satisfied, SF-RGE and SF-RPLE can further extend cloaking boxes along the time axis while RGE and RPLE have to stop and response a 'FAIL'. This difference makes SF-RGE and SF-RPLE provide even lower RSR. In contrast, TF-RSTC schemes first expand cloaking boxes along the time axis, thus resulting in much smaller spatial cloaking area and much higher RSR. Second, in all three scenarios, cloaking schemes with RGE always perform higher RSR than cloaking schemes with RPLE. The reason is that RGE generates lists for segments in a dynamic on-the-fly manner, which helps make segments within



(a) Anonymization Time (b) Delay (c) Relative Spatial Resolution (d) Success Rate

Fig. 10. Space-first spatio-temporal cloaking.



(a) Anonymization Time (b) Active Segments (c) Relative Spatial Resolution (d) Success Rate

Fig. 11. Time-first spatio-temporal cloaking.

the cloaking region tighter and the cloaking area smaller. Finally, we observed that when δ_k increases, only RSR offered by TF-RSTC schemes also increases. The main reason is that the accumulation of mobile users along the time dimension in TF-RSTC schemes makes the expansion of cloaking boxes in spatial domain fall behind the expansion of maximum allowable area due to the increment of δ_k.

In Fig. 8(b), we show the results of success rate with varying δ_k. When δ_k increases from 10 to 200, the success rates offered by RGE and RPLE eventually decrease from higher values close to 100% to very low values, 71% for RGE and 52% for RPLE. In contrast, the success rates offered by all the reversible spatio-temporal cloaking schemes are always at 100%. This shows that reversible spatio-temporal cloaking schemes can significantly improve request success rate by making full use of both σ_t and σ_s.

To sum up, the results of the first part of experiments prove that the reversible spatio-temporal cloaking algorithms proposed in this work offer higher relative spatial resolution and higher query success rate than the reversible spatial cloaking algorithms in [17].

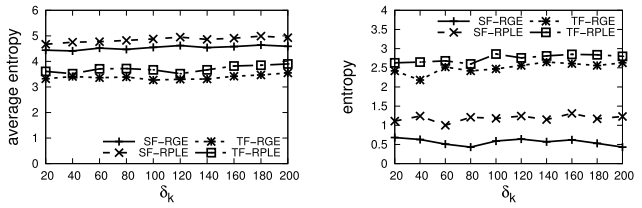
4.3. Reversible and irreversible spatio-temporal cloaking comparison

In the second part of designed experiments, we compare four reversible spatio-temporal cloaking algorithms (TF-RGE, TF-RPLE, SF-RGE, SF-RPLE) with four irreversible spatio-temporal cloaking algorithms (TF-RS, TF-SE, SF-RS, SF-SE). For short, the random sampling (RS) scheme first chooses the road segment containing the actual user. It then randomly adds segments within the bounded area restricted by σ_s into the cloaking region until the requirements of δ_k and δ_l are met. In the star-based road-network expansion(SE) [6], instead of randomly choosing segments from the bounded region in a discrete manner, the segments are chosen continuously based on an expansion scheme. The expansion begins from the segment containing the actual user and randomly expands such that each newly added segment is adjacent to at least one other segment in the currently formed cloaking region.

We first evaluate the eight algorithms in terms of response delay and RSR when δ_k is changed from 10 to 200. The results are shown in Fig. 9. Here, we fix δ_l to 10, σ_s to 300 m and σ_t to 30 s. As can be seen in Fig. 9(a), all the TF-RSTC algorithms have their response delay close to 15 s, namely a half of σ_t. This is because all TF-RSTC algorithms directly extend the time window length to σ_t and the window end is randomly taken within the range. In contrast, all SF-RSTC algorithms perform much smaller response delay, especially when δ_k is small. The reason is that a smaller δ_k requires fewer mobile users to be collected along the time axis, thus resulting in a smaller window length. In Fig. 9(b), the RSR of all the TF-RSTC algorithms is much larger, indicating that smaller spatial cloaking area can be provided. Therefore, to sum up, in cases when a shorter delay is the primary objective, SF-RSTC algorithms work much better. We can conclude that TF-RSTC algorithms should be preferred when more accurate feedback is expected.

Next, we separately evaluate SF-RSTC algorithms and TF-RSTC algorithms using additional metrics, which are anonymization time, response delay, relative spatial resolution (RSR) and success rate. In Fig. 10, we evaluate the performance of the four SF-RSTC algorithms with varying $\sigma_s = 300\text{ m}, 350\text{ m}, 400\text{ m}, 450\text{ m}$ and we fix δ_k to 200, δ_l to 10 and σ_t to 30 s. In SF-RSTC algorithms, the cloaking box first extends along x–y axes before σ_s is reached and then turns to the time axis to capture more mobile users to satisfy δ_k. In Fig. 10(a), following the growth of σ_s, the anonymization times of all the algorithms increase because larger σ_s indicates more segments to be included into the cloaking area. In this case, for all the tested spatial tolerances, the boundary of the bounded area is reached first, so the RSR of all algorithms is always minimum (Fig. 10(c)). Once all the available segments have been captured, the extension of cloaking box turns to the time axis. For a smaller σ_s, a longer time window is required as shown in Fig. 10(b). In Fig. 10(d), the success rates of all algorithms in all four cases are very close to 1 indicating that the techniques are highly reliable.

Finally, in Fig. 11, we evaluate the performance of the four TF-RSTC algorithms with varying $\sigma_t = 0$ s, 10 s, 20 s, 30 s and we fix δ_k to 200, δ_l to 10 and σ_s to 1000 m. In TF-RSTC algorithms, the cloaking box first extends along the time axis to reach σ_t. In Fig. 11(a), the anonymization time of all the four algorithms decreases when σ_t is larger. Larger time window means smaller spatial cloaking area, thus fewer required segments. Fig. 11(b) measures the number of active segments with different σ_t. A segment is said to be active if it can fill the k-user requirement of δ_k by at least one user once it is added to the cloaking area. For a larger σ_t, a mobile user has more chance to pass more segments, which also means a segment may be passed by more mobile users and have a higher chance to become active. When σ_t is close to 0 s, which shows a snapshot record, only 2133 segments among the 9187 segments are active. This value increases to 3443 when σ_t rises to 30 s. In Fig. 11(c), the relative spatial resolution (RSR) is measured.



(a) Replay attack (b) Network distance attack

Fig. 12. Attack resilience.

For all the four algorithms, the RSR increases significantly, especially for TF-RGE and TF-SE. The reason for that is the increment of the number of active segments, which makes the algorithms satisfy δ_k with fewer selected segments, thus smaller cloaking area and larger RSR. Fig. 11(d) shows the results of success rate. The success rates of all the algorithms rise towards 100% with increasing σ_t. For the case when σ_t is 30 s, the success rates of the four algorithms are almost the same, which is very close to 1.

As can be seen from the results, the reversible spatio-temporal algorithms maintain similar performance as irreversible techniques in terms of anonymization time, response delay, relative spatial resolution (RSR) and success rate. Thus the reversibility feature of the proposed schemes does not come at the cost of any reduction in performance.

4.4. Attack resilience evaluation

This set of experiments evaluate the effectiveness of the four reversible spatio-temporal cloaking algorithms (TF-RGE, TF-RPLE, SF-RGE, SF-RPLE) in terms of their resilience to replay attack and network distance attack.

For replay attack, average information entropy is used as the metric to evaluate the uncertainty of the attacker: $entropy = -\sum A_i \log A_i$, where A_i is the associativity for each segment. Here, higher entropy means higher randomness and higher uncertainty for the attacker in inferring the true location of the user, thus leaking out less information and providing better privacy protection. Fig. 12(a) shows average entropy of the replay attack with varying δ_k. It can be seen that both SF-RGE and SF-RPLE offers higher average entropy than TF-RGE and TF-RPLE, which is the results of larger spatial cloaking area provided by SF-RSTC algorithms. In addition, RPLE performs higher average entropy in both TF-RSTC and SF-RSTC than RGE, which indicates that the looser cloaking region offered by RSTC has higher randomness than the tighter cloaking region offered by RGE.

In order to measure the resilience of the schemes against network distance attack, we measure entropy that captures the uncertainty of the attacker in identifying which privacy level a cloaking segment belongs to. In this experiment, we consider eight privacy levels beyond L^0 and therefore, the highest possible entropy of a network-distance attack in this case would be 3. This highest entropy represents the highest possible uncertainty of the adversary representing the scenario when the associativity of all the segments for all the eight levels follow a uniform distribution. Fig. 12(b) shows the results for varying δ_k. As can be seen, both TF-RGE and TF-RPLE offer higher entropy than SF-RGE and SF-RPLE. In SF-RSTC algorithms, since the spatial cloaking area is usually expanded close to the maximum allowable area, the position of the real user is usually closer to the center of the maximum allowable area. As a result, the probability to find the segment containing the real user through network distance attack becomes higher. In contrast, TF-RSTC algorithms can perform much better resilience regarding network distance attack. In addition, we can find that RPLE performs higher entropy in both TF-RSTC and SF-RSTC than RGE, which is also the results of looser cloaking region offered by RSTC.

Thus, this set of experiments shows that even though all algorithms offer significant resilience against the adversarial attacks, the TF-RSTC algorithms offer relatively higher resilience against network distance attack while the SF-RSTC algorithms offer higher resilience against the replay attack.

5. Related work

Location privacy has been an active area of research in the past. Broadly, location privacy protection mechanism can be classified into policy-based protection techniques and inference prevention-based techniques. Policy-based schemes give users permission to define privacy rules according to the service request, thus getting users' active participation. The inference-prevention schemes are more focused on prevention by protectively processing and perturbing the location information prior to disclosure. The latter can be further broken down into location data perturbation techniques represented by [10,14,15,17,24,25] and trajectory inference prevention techniques represented by [21,26–29].

Location data perturbation schemes consists of perturbation through dummies [30,31], information-theoretic approaches [24,25], spatial location cloaking [7,10,11,13–15,32] and differential privacy [8,9,12,33–35]. The goal of location data perturbation is to perturb users' real location information so that the injected uncertainty can resist potential attacks made by adversaries. In dummy-based approaches [30,31], when a user sends a query to one LBS provider, some dummy locations are also sent with the query. The dummy trajectories should have similar properties of the real trajectories so that adversaries with map information cannot distinguish the real trajectories from the dummy trajectories. In information-theoretic approaches [24,25], to reduce privacy leakage, the location data of a user consists of public data that is safe to be released and private data that should be well protected. However, information about private data may be revealed from the public data. To solve this, the public data of a user can be obfuscated through a carefully designed probabilistic obfuscation function so that information of private data is hard to be inferred from the obfuscated public data [24].

In the past, there have been many works related to spatial location cloaking. To proactively protect user's location privacy, k-anonymity, which was proposed for sensitive data protection [36], was applied to protect location privacy in the context of location-aware systems [37]. Since then, the techniques related to spatial cloaking has been developing rapidly. *CliqueCloak* algorithm proposed in 2004 considered the individual user's personalized privacy requirement for the first time [10]. A grid-based cloaking framework, *Casper* further extended this model with a privacy-aware query processor [14,38]. Subsequently, a directed-graph based cloaking algorithm was proposed to improve the success rate of anonymization [15] and the Hilbert Cloak algorithm uses a Hilbert curve to fill the whole area and track users [11].

These traditional cloaking schemes have some limitations. Most traditional cloaking techniques were designed for mobile users traveling on Euclidean space, recent work has considered the location cloaking problem under a constrained road network model [6,39,40]. In [41], location labels are introduced to distinguish locations of mobile users to sensitive and ordinary locations, which can be viewed as an enhanced cloaking technique in the IoT scenario. In [42], the fully trusted Anonymizer, which is usually required by most traditional cloaking schemes to perform the cloaking algorithms, is replaced by a function generator distributing the spatial transformation parameters periodically. In [43], an information-theoretic approach was introduced to define the notion of perfect location privacy, which indicated how to ensure users' perfect location privacy through anonymization methods. In [44,45], game theory models were applied to further enhancing cloaking schemes. Specifically, the privacy-utility tradeoff was modeled as a Stackelberg Bayesian game in [44] while a hide-and-seek game-theoretic model was used in [45] to prevent the rational trusted third party from colluding with rational adversaries.

Another dimension of recent work has studied the location privacy problem by perturbing the location information based on differential privacy constraints prior to disclosure [33–35]. Differential privacy [46, 47] provides rigorous protection against adversaries with background knowledge and quantifies the privacy in a mathematically provable manner. By carefully applying differential privacy mechanisms [46–48] to the trajectory data, the personal location information in the disclosed statistical output can be protected. Usually, the raw location dataset is first transferred to a special data structure, such as Prefix tree [9,12] or N-gram [8]. Then, the differential privacy protection mechanisms (e.g. Laplace Mechanism [46], Exponential Mechanism [48]) inject noises to the data structures before releasing them for further processing. While differential privacy provides a more formal and rigorous privacy guarantee against background knowledge attacks, it can result in a higher perturbation and may provide a lower data utility compared to anonymization techniques. Thus in cases where there is a lack of background knowledge and when the risks of such attacks are minimal, anonymization techniques are likely to provide a higher data utility compared to differential privacy.

As we can observe, most existing location privacy protection mechanisms have focused on developing unidirectional location perturbation approaches that do not allow fine granular information to be inferred even when some users have the privileges to access it. The reversible spatial cloaking algorithms proposed in [17] use access keys to control the pseudo-randomness required for generating an attack-resilient spatial cloaking region, thus allowing data owners to control the utility and privacy levels of their data. These algorithms are spatial cloaking algorithms that expand the cloaking region only along the spatial dimension. As a result, reversible spatial location cloaking techniques obtain lower success rate and lower spatial resolution of the perturbed location leading to lower reliability and reduced service quality. In contrast, the work presented in this paper leverages the more sophisticated spatial–temporal cloaking model [19] that perturbs the location data along both spatial and temporal dimensions while still ensuring that the spatio-temporal expansion process is reversible when suitable access keys are provided. Our experimental results show that, compared with the two reversible spatial cloaking algorithms in [17], the reversible spatio-temporal cloaking schemes proposed in this paper have a significant performance improvement in terms of spatial resolution and query success rate.

6. Conclusion

In this paper, we presented a new class of reversible spatio-temporal cloaking mechanisms for supporting multi-level privacy requirements in access controlled environments. We argue that conventional location perturbation techniques are irreversible and are not inherently designed to support multi-level privacy of users. While recent techniques on reversible spatial cloaking techniques employ data anonymization keys to perturb a users location in a pseudo-random manner, the performance of these schemes in terms of success rate and service quality is limited by their adopted spatial cloaking model in which the location perturbation occurs solely in the spatial domain. In this work, we have developed two reversible spatio-temporal cloaking mechanisms namely (i) time-first reversible spatio-temporal cloaking and (ii) space-first reversible spatio-temporal cloaking scheme that effectively support multi-level privacy, allowing users with higher privileges to obtain finer location information through reduced anonymity levels. The proposed techniques allow data perturbation to occur along both spatial and temporal dimensions while still ensuring that the spatio-temporal expansion process is reversible when suitable access keys are provided. We evaluate the proposed techniques through extensive experiments on real road networks that show that the proposed model achieves higher success rate and higher spatial resolution compared to the reversible spatial cloaking and offers better QoS performance and strong attack resilience against adversarial attacks.

References

[1] Most Smartphone Owners Use Location-Based Services. eMarketer, 2016.
[2] Technology Device Ownership: 2015. Pwe Research Center, 2016.
[3] K.O. Stalker, Victims Should Check For GPS, Associated Press, 2003.
[4] J. Krumm, A survey of computational location privacy, Pers. Ubiquitous Comput. 13 (6) (2009) 391–399.
[5] A. Machanavajjhala, D. Kifer, J. Gehrke, et al., l-diversity: Privacy beyond k-anonymity, CM Trans. Knowl. Discov. Data (TKDD) 1 (1) (2007) 3.
[6] T. Wang, L. Liu, P. Pesti, Privacy-aware mobile services over road networks, VLDB Endowment 2 (1) (2009) 1042–1053.
[7] B. Bamba, L. Liu, P. Pesti, et al., Supporting anonymous location queries in mobile environments with privacygrid, in: 17th international conference on World Wide Web (WWW 2008), pp. 237-246.
[8] Chen Rui, Gergely Acs, Claude Castelluccia, Differentially private sequential data publication via variable-length n-grams, in: Proceedings of 2012 the ACM conference on Computer and communications security, ACM, 2012.
[9] Chen Rui, et al., Differentially private transit data publication: a case study on the montreal transportation system, in: Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2012.
[10] B. Gedik, L. Liu, A customizable k-anonymity model for protecting location privacy, 2004.
[11] G. Ghinita, P. Kalnis, S. Skiadopoulos, PRIVE: anonymous location-based queries in distributed mobile systems, in: 16th international conference on World Wide Web (WWW 2007), pp. 371-380.
[12] He Xi, et al., Dpt: Differentially private trajectory synthesis using hierarchical reference systems, Proc. VLDB Endowment 8 (11) (2015) 1154–1165.
[13] P. Kalnis, G. Ghinita, K. Mouratidis, et al., Preventing location-based identity inference in anonymous spatial queries, IEEE Trans. Knowl. Data Eng. 19 (12) (2007) 1719–1733.
[14] M.F. Mokbel, C.Y. Chow, W.G. Aref, The new Casper:query processing for location services without compromising privacy, VLDB Endowment (2006) 763–774.
[15] Z. Xiao, X. Meng, J. Xu, Quality aware privacy protection for location-based services, in: Advances in Databases: Concepts, Systems and Applications, Springer Berlin Heidelberg, 2007, pp. 434–446.
[16] S. Salvatore, et al., Socio-spatial properties of online location-based social networks, in: ICWSM 11, 2011, pp. 329–336.
[17] C. Li, B. Palanisamy, ReverseCloak: Protecting Multi-level Location Privacy over Road Networks, in: Proc. of 24th ACM International Conference on Information and Knowledge Management (CIKM), 2015.
[18] N. Li, T. Li, S. Venkatasubramanian, t-closeness: Privacy beyond k-anonymity and l-diversity, in: Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on, IEEE, 2007, pp. 106–115.
[19] B. Gedik, L. Liu, Protecting location privacy with personalized k-anonymity: Architecture and algorithms, IEEE Trans. Mob. Comput. 7 (1) (2008) 1–18.
[20] C.F. Chow, M.F. Mokbel, Enabling private continuous queries for revealed user locations, Ad. Spatial Temporal Databases (2007) 258–275.
[21] B. Palanisamy, L. Liu, K. Lee, et al., Anonymizing continuous queries with delay-tolerant mix-zones on road networks, Distrib. Parallel Databases DAPD 32 (1) (2014) 91–118.
[22] R. Shokri, G. Theodorakopoulos, J.Y. Le Boudec, et al., Quantifying location privacy, in: 32nd IEEE Symposium on Security and Privacy, 2011, pp. 247–262.
[23] GTMobiSim. https://code.google.com/p/gt-mobisim/.
[24] D. Yang, D. Zhang, B. Qu, P. Cudré-Mauroux, PrivCheck: privacy-preserving check-in data publishing for personalized location based services, in: Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, ACM, 2016.
[25] D. Yang, B. Qu, P. Cudré-Mauroux, Privacy-Preserving Social Media Data Publishing for Personalized Ranking-Based Recommendatio, IEEE Trans. Knowl. Data Eng. (2018).
[26] A. Beresford, F. Stajano, Location Privacy in Pervasive Computing , in: Pervasive Computing, 2003, pp. 46–55.
[27] A. Beresford, S. Frank, Mix zones: User privacy in location-aware services, 2004.
[28] B. Palanisamy, L. Liu, Attack-resilient mix-zones over road networks: architecture and algorithms, IEEE Trans. Mob. Comput. TMC 14 (3) (2015) 495–508.
[29] B. Palanisamy, L. Liu, Mobimix: Protecting location privacy with mix-zones over road networks, in: 27th International Conference on Data Engineering (ICDE 2011), pp. 494-505.
[30] H. Kido, Y. Yanagisawa, T. Satoh, Protection of location privacy using dummies for location-based services, in: 25th International Conference on Distributed Computing Systems(ICPS 2005), pp. 1248-1248.
[31] S. Hayashida, et al., Dummy Generation Based on User-Movement Estimation for Location Privacy Protection, IEEE Access 6 (2018) 22958–22969.
[32] R. Cheng, Y. Zhang, E. Bertino, et al., Preserving user location privacy in mobile data management infrastructures, in: Privacy Enhancing Technologies, 2006, pp. 393–412.
[33] M.E. Andres, N.E. Bordenabe, K. Chatzikokolakis, et al., Geo-indistinguishability: differential privacy for location-based systems, in: 20th ACM SIGSAC conference on Computer and communications security (CCS2013), pp. 901-914.

[34] N.E. Bordenabe, K. Chatzikokolakis, C. Palamidessi, Optimal geo-indistinguishable mechanisms for location privacy, in: 21th ACM SIGSAC Conference on Computer and Communications Security (CCS 2014), pp. 251-262.
[35] Hua Jingyu, et al., A Geo-Indistinguishable Location Perturbation Mechanism for Location-Based Services Supporting Frequent Queries, IEEE Trans. Inf. Forensics Secur. (2017).
[36] L. Sweeney, A model for protecting privacy. International Journal of Uncertainty, Fuzzin. Knowl.-Based Syst. (2002) 557–570.
[37] M. Gruteser, D. Grunwald, X. Liu, Anonymous usage of location-based services through spatial and temporal cloaking, in: 1st international conference on Mobile systems, applications and services, 2003, pp. 31–42.
[38] M.F. Mokbel, C.Y. Chow, W.G. Aref, The new casper: A privacy-aware location-based database server, in: ICDE, 2017.
[39] H.J. Cho, S.J. Kwon, R. Jin, et al., A privacy-aware monitoring algorithm for moving k-nearest neighbor queries in road networks, Distrib. Parallel Databases (2014) 1–34.
[40] B. Ying, D. Makrakis, Protecting location privacy with clustering anonymization in vehicular networks, in: Computer Communications Workshops (INFOCOM WKSHPS 2014), pp. 305-310.
[41] G. Sun, et al., L2P2: A location-label based approach for privacy preserving in LBS, Future Gener. Comput. Syst. 74 (2017) 375–384.
[42] P. Tao, Q. Liu, G. Wang, Enhanced location privacy preserving scheme in location-based services, IEEE Syst. J. 11 (1) (2017) 219–230.
[43] Z. Montazeri, et al., Achieving perfect location privacy in wireless devices using anonymization, IEEE Trans. Inf. Forensics Secur. 12 (11) (2017) 2683–2698.
[44] R. Shokri, et al., Privacy games along location traces: A game-theoretic framework for optimizing location privacy, ACM Trans. Priv. Secur. (TOPS) 19 (4) (2017) 11, 74.
[45] D. Ying, N. Amiya, Location privacy-protection based on p-destination in mobile social networks: A game theory analysis, in: IEEE Conference on Dependable and Secure Computing, 2017.
[46] Dwork Cynthia, et al., Calibrating noise to sensitivity in private data analysis, in: TCC, Vol. 3876, 2006.
[47] C. Dwork, A. Roth, The algorithmic foundations of differential privacy, Theoret. Comput. Sci. 9 (3–4) (2013) 211–407.
[48] McSherry Frank, Kunal Talwar, alwar Mechanism design via differential privacy, in: FOCS'07, 48th Annual IEEE Symposium on, IEEE, 2007.