Location Privacy Preservation Based on Continuous Queries for Location-Based Services

Li Zhang*, Yuwen Qian*, Ming Ding+, Chuan Ma*, Jun Li*, and Sina Shaham**

*Nanjing University of Science and Technology, Nanjing, China
+Data61, CSIRO, Sydney, Australia
**The University of Sydney, Sydney, Australia
[1]Email:{emmazh, admon, jun.li, chuan.ma}@njust.edu.cn, ming.ding@data61.csiro.au, sina.shaham@sydney.edu.au

***Abstract*—In recent years, location-based services (LBSs) have become increasingly popular, as they can greatly facilitate people's daily life. However, LBSs also raise serious concerns that the collected big data can be utilized by adversaries to track users' movements, and infer their specific locations and living habits. To address this privacy issue, we propose an algorithm based on the k-anonymity criterion, to generate dummy locations to protect users' privacy. We also develop a new metric, the trajectory entropy, to measure the performance of anonymity. Different from the existing technologies, which focus on the stationary location privacy, we further consider the correlation between adjacent locations in the continuous queried process. We assume it as a Markov process and exploit such Markov process to achieve privacy protection. Our simulation results on the real-life dataset show that compared with other algorithms, the proposed algorithm exhibits superior performance in providing location anonymity by resisting most attacks.**

I. Introduction

Under the explosive increase of big data, the rapid development of mobile devices [1], [2] and the Global Positioning System (GPS) technology, location-based services (LBSs) have changed the world. However, providing such services with private location information may raise some serious concerns, which is due to the fact that a certain amount of users' location information can be collected by LBSs servers, in order to provide personalized services by analyzing daily trajectories. If the location information is not protected adequately or the LBSs provider is hacked, the adversary can easily infer users' personal information by accessing the big data. Therefore, there is an urgent demand to protect personal location privacy in LBSs.

To address the issue, many approaches have been proposed in the past few years [3], [4]. Besides the cryptography primitive-based approaches in [4], existing works mainly focused on the obfuscation technique and anonymity-based technique [5], [6]. The obfuscation techniques aimed to protect the personal information by reducing the accuracy of position information sent to LBSs provider deliberately. Anonymity-based techniques are designed to protect the link between users' sensitive information and identities. Among anonymity-based techniques, k-anonymity is a widely accepted criterion to protect the location privacy, which was initially proposed to solve the sensitive information problem of relational databases [7]. Then it was introduced into the location privacy protection by Gruteser [8]. The main idea of k-anonymity is that the user transmits a set of k objects containing one real position and $k-1$ fake positions to the LBSs provider, and then the true location can be hidden in k locations. As mentioned in [9], the k-anonymity heavily relies on the third trusted anonymizer, so there is a potential risk if the LBSs provider is compromised by the adversary. To avoid this risk, an algorithm based on pseudo-location swapping scheme was proposed to achieve the anonymity [10]. However, the fake positions can be easily inferred as they are geographically closed. To solve the problem, Lu et al. proposed a dummy selection solution based on virtual grid or circle model [11]. While, the side information mentioned in [11]–[13] was considered to be a newly discovered risk if utilized by the adversary, especially in the big data era. For example, if the adversary acknowledges the side information, such as the query probability, it would be easier to infer the true location by excluding impossible fake ones. To lower this risk, Niu et al. [14] designed a Dummy-Location Selection (DLS) algorithm based on side information exploited by the adversary to achieve anonymity.

However, all the mentioned works only focused on stable location protection. If users travel across several locations, the side information, such as the transition probability between adjacent locations, can be utilized by the adversary to extract users' private location information. Then the privacy would be compromised. Recently, Shaham et al. proposed a method named robust dummy selection (RDG) in [15], which took advantage of transition entropy to solve private information leakage problem in continuous queries. The transition entropy is to calculate each location's posterior probability, which indicates the likelihood of the location being the real one. However, the RDG algorithm only measures the likelihood of positions but overlooks the likelihood of path. This would ignore the correlation between consecutive dummy locations, which might lead to the disclosure of the true location.

Motivated by [15], in this paper, we introduce a new metric named the trajectory entropy to measure the degree of anonymity. In more details, we first choose dummy locations based on the DLS algorithm, and then we further propose

an algorithm named path-based dummy selection (PBDS) to achieve the k-anonymity requirement. The major contributions of this paper are shown as follows,

- We propose a novel metric based on k-anonymity named trajectory entropy, which measures the uncertainty of distinguishing the real location from the dummy locations.
- We design an algorithm in terms of path protection named PBDS to protect privacy against the adversary, which utilizes the probabilities between adjacent locations, and guarantee the privacy.
- We analyze the performance of the proposed algorithm, and show its superiority by comparison with other existing algorithms based on a real-life dataset.

The rest of paper is organized as follows. Firstly, we introduce our system model in Section II. Next, in Section III we describe the proposed scheme and show the proposed algorithm in details. In Section IV, the attack model is presented and in Section V, simulation results and discussions are provided. Finally, we draw the conclusion in Section VI.

II. SYSTEM MODEL

In this section, we present the system model that is used in our study.

A. System Architecture

This section describes the system architecture. In general, the system model consists of three components: mobile users, LBSs providers and clients. Clients always query the LBSs provider for users' positions to execute a certain service based on different locations. In our model, we assume that the mobile users can directly contact with the LBSs provider without a trust third party (TTP), and the LBSs provider is not always incredible.

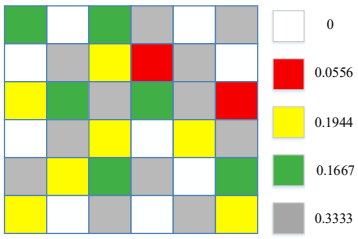

Fig. 1. Different query probability of different location

For ease of calculation, we divide the location map into an $n \times n$ cells. As shown in Fig. 1, different shades of locations have unique query probabilities. At time t, a user needs to send a query to the LBSs provider. Meanwhile, the user generates $k-1$ dummy locations and sends them together with the real location to the LBS provider to achieve k-anonymity. As a result, the adversary is not able to identify the real location with probability higher than $1/k^t$, and the user can hide the true location from the adversary. We denote that the set containing k positions at time t is

$$S^t = \{s_1^t, s_2^t, s_3^t, ..., s_k^t\}, \tag{1}$$

and the real location is r^t, where $r^t \in S^t$. The probability of location s_x^t being the real location can be written as:

$$P[s_x^t = r^t] = \frac{number\ of\ queries\ in\ s_x^t}{number\ of\ queries\ in\ the\ whole\ map}. \tag{2}$$

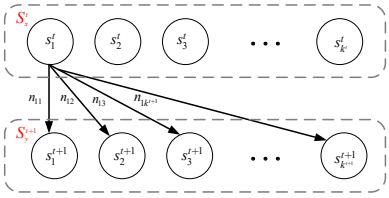

Fig. 2. The continuous queried locations of a user

We also consider the situation in which the user releases the location information continuously. The detail of this movement can be depicted in Fig. 2. As shown in this figure, at time t, a user u sends the dummy set S_x^t containing a real location and $k-1$ fake locations to the LBSs provider, and at next time $t+1$, this user moves to a new position and makes new query with location set S_y^{t+1}. We denote the number of times the location s_x^{t+1} being queried follows the location s_y^{t+1} is n_{xy}, which can be calculated from the historical data. Then the transition probability of s_y^{t+1} being queried after s_x^t consecutively can be written as

$$P[s_y^{t+1}|s_x^t] = \frac{n_{xy}}{\sum_{y=1}^{k^{t+1}} n_{xy}}. \tag{3}$$

B. Adversary model

We assume that the adversary wants to obtain the sensitive information of a particular user, such as the user's location information and daily trajectory information. In general, there are two types of adversary models: active adversary and passive adversary:

- passive adversary monitors and eavesdrops the communication between the LBSs provider and users. By analyzing the collected information, the passive adversary can compromise the location privacy.
- active adversary can further obtain all the information what the server acknowledged and he can also obtain the historical data of a particular user. Once the server is compromised, there would a serious privacy leakage.

In this work, we use the active adversary. As a result, the daily queries and the historical information of a particular user are known by the adversary.

C. Side information

Side information, such as users' query probabilities related to time and locations, and messages with sensitive information, can be used as clues for the adversary to infer the private information. Using the side information, the adversary can launch a specific attack to the user.

In this work, we suppose that the side information is related to the historical data of the location map, such as query probability and transition probability. The query probability is defined as the number of times that a particular location

has been queried. The adversary can infer the likelihood of a location being queried in the future by looking at the query probability. For instance, if a user queries two different locations, the one with a higher query probability can be considered as the true location.

In addition to possessing the query probability, we assume the adversary also has the access to the transition probability, which can be calculated by times that a location is queried after another location. In general, when it comes to the trajectories, we need to consider not only the location probability, but also the path probability, which is obtained from the transition probability.

III. Method of dummies generated based on the trajectory entropy

In this section, we propose an algorithm based on the trajectory entropy to achieve the privacy protection.

A. Dummy selection based on a stationary location query

Before formally proposing our algorithm, we first introduce a stationary query DLS algorithm [14]. To measure the anonymity degree of the stationary dummy locations, we introduce the concept of the set entropy. It can describe the uncertainty that the true location is identified from the dummy set. In the DLS algorithm, each stationary location has a query probability denoted by q_i to construct the entropy metric, and the sum of all probabilities q_i is one. Given a dummy set $S^t = \{s_1^t, s_2^t, ..., s_n^t\}$, and we extract the query probability of this set as $Q = \{q_1^t, q_2^t, ..., q_n^t\}$, where q_i^t is the query probability of location s_i^t. Then, the set entropy H of identifying the true location in the candidate is

$$H = - \sum_{i=1}^{k} q_i^t * \log_2 q_i^t, \tag{4}$$

where k denotes the total number of candidate locations.

According to the concept of entropy, the maximum value, $H_{max} = log_2 k$ can be obtained in the case that all the k possible locations have the same probability $1/k$. So in the stationary query algorithm, the aim is to make each dummy location to have an equal probability to achieve the maximum entropy. As a result, it yields the highest uncertainty of identifying the true location from the candidate set.

B. Dummy selection based on continuous location queries

When continuous queries sent by users are considered, the set entropy is not sufficient to satisfy to measure the degree of privacy protection. In the continuous queries process, two consecutive locations in the same path are not independent, so existing algorithms can expose the true trajectory. For example, suppose there are two location sets X and Y in different time slots, according to information theory [16], when the prior probability $P(x)$, $P(y)$, and transition probability $P(y|x)$ between two set are known, the mutual information $I(X;Y)$ can be written as

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} P(y|x)P(x)\log_2 \frac{P(\mathrm{y}|x)}{P(y)}. \tag{5}$$

According to (5), we can further deduce the uncertainty of set Y when the mutual information is known as

$$H(Y|X) = H(Y) - I(X;Y) \leq H(Y), \tag{6}$$

where $H(Y)$ is the entropy of the dummy set Y at time $t+1$. It can be seen that the set Y should be constructed carefully, otherwise the conditional entropy might be very small, which reveals the true location. Hence, a new metric should be developed to protect the privacy in trajectories. So we design an algorithm to generate dummy locations correlatively by utilizing the query probability and transition probability.

Considering that a continuous location queries process contains several steps, to properly protect the location privacy of a user, we need to capture the moving behavior of users first.

1) The transition probability model: For the sake of simplicity, we use the frequency of the movements to model the moving behaviors. Denoted by V the set of locations according to the history trajectories, a user moves from an arbitrary location i to another location j in the map, we get a *movement matrix* M_t to specify the history of past movements. The size of M_t is $N \times N$. We use $M_t[i][j]$ to denote the number of times that a user leaves from the location i to location j. The data is calculated from the historical database LBSs provider holds.

To best illustrate this moving behavior, we show a 4x4 movement matrix example in Table I. There are 4 locations considered, and each cell shows the number of times a user left from a location to another one.

TABLE I
MOVEMENT MATRIX

	A	B	C	D
A	-	-	20	20
B	10	-	20	10
C	20	10	-	10
D	10	-	10	-

TABLE II
TRANSITION MATRIX

	A	B	C	D
A	-	-	0.50	0.50
B	0.25	-	0.50	0.25
C	0.50	0.25	-	0.25
D	0.50	-	0.50	-

Based on M_t, we can obtain a transition probability matrix T in the follow, where $T[i][j] = P(j|i)$, $\forall i, j \in V$. And $P(j|i)$ can be computed as $P(j|i) = \frac{M_{ij}}{\sum_i M_{ij}}$. Furthermore, T satisfies the following properties,

(1) $\forall i = 1, 2, ..., N \in V, \forall j = 1, 2, ..., N \in V, T[i][j] \geq 0$, that is, all elements in the matrix must be no less than zero.

(2) $\forall i = 1, 2, ..., N \in V, \sum_{j=1}^{N} T[i][j] = 1$, that is, the summation of probability on each row equals to one.

For instance, in Table I, a user departures from location B to location A, C and D and the number of times can be calculated as 10, 20 and 10, respectively. Furthermore, in Table II, the transition probability of this move can be calculated as 0.25, 0.5 and 0.25, respectively.

2) The path probability model: In each time slot, we can obtain the movement and transition probability matrix. Without loss of generality, the process of user releasing continuous queries can be seen as a Markov process.

Definition 1: Given a path $\langle v_1, v_2, ..., v_j \rangle$ for a user u, the path probability can be calculated as:

$$P[\langle v_1, v_2, ..., v_j \rangle] = \prod_{l=1}^{j-1} P[l+1|l], \tag{7}$$

where j is the path length.

3) The trajectory entropy model: To hide the true location carefully, we need to choose $k-1$ paths to achieve the dummy set, which guarantees the next selected location can be achieved by the transition probability matrix and the path probability is also similar to the real one. To measure the degree of privacy, we propose a definition named *trajectory entropy* as follows.

Definition 2: Considering that a trajectory contains m locations, that is the path length is m, and there are one real trajectory and $k-1$ false trajectories. For a particular set, the trajectory entropy is defined as

$$H_{\mathrm{t}} = -\sum_{i=1}^{k} \Pr[\Omega = R] * \log_2 \Pr[\Omega = R], \tag{8}$$

where $Pr[\Omega = R]$ is the probability that the chosen path is the real one, Ω is a set of chosen locations in each time, and R is the real location set in each time. $Pr[\Omega = R]$ can be calculated by normalization of each path probability. Hence the trajectory entropy can be written in (9), which is shown at the top of next page.

The trajectory entropy metric demonstrates the uncertainty of disguising the real location from the dummy set. The larger value of trajectory entropy indicates the probability of each path is more similar to others. Hence, it would be difficult to infer the true location from the dummy set even though the side information is considered by the attacker. In order to achieve the privacy protection, our target is to get the maximum trajectory entropy.

We propose a dummy selection algorithm based on trajectory entropy named as the PBDS algorithm. As shown in Algorithm l, the PBDS algorithm starts by generating a pool dummies based on the DLS algorithm to make sure high performance of set entropy, and the size of pool dummies is larger than k. Usually the pool size can be multiple times of k and here we set it as $4k$. Then we consider the continuous locations in a trajectory. Suppose the user has made a query for the service at time t, and released a continuous query at time $t+1$. At time t, we choose one location from the pool as the dummy one. Then at time $t+1$, we choose the location with maximum trajectory entropy and add it to the dummy set. Then we repeat the action in each time slot until the path is achieved.

The main advantage of our algorithm can be considered as:

(1) We consider the privacy protection of continuous queries in a trajectory for a specific user rather than a stationary location.

Algorithm 1: Dummy-location selection algorithm based on PBDS

Input: the true location set, the dummy selection set D^{t+c}, dummy set only includes the real location of user at $(q+1)^{-th}$ query

Result: dummy set, trajectory entropy

Initialization: $D^{t+c} \leftarrow$ generate a pool of $4k^{t+c}$ dummies set using the DLS algorithm in each time slot

Sort the dummy location by difference value of transition times between each path and real path

for $1 \leq$ *member* $\leq k^t - 1$ **do**

  for $1 < c \leq$ *path length* **do**

    choose location from D^{t+c} to update dummy set

    calculate the path probability $P(\langle v_1, v_2, ..., v_j \rangle)$

    calculate the trajectory entropy H_t with c path

  Chosen Member : the location with maximum trajectory entropy in D^{t+c};

  dummy set $\leftarrow$ Chosen Member;

  D^{t+c} =D^{t+c} - Chosen Member ;

(2) We consider the path probability rather than the position probability to ensure each path is similarly.

(3) We introduce a constraint on location reachability to ensure the path can be considered for real.

IV. ATTACK MODEL

An attacker can explore various detection algorithms to obtain users' location information. If the attack from the adversary reduces the degree of user's anonymity, or excludes other users in the anonymous set, we suppose such attack is effective.

To model the attack, we use the Viterbi attack mentioned in [15]. The Viterbi algorithm is used to find the most likely path or the shortest path, which is a highly effective method to compromise the location privacy by searching for the most probable path. In addition, the adversary can exploit the side information to violate the location privacy by Viterbi attack. In the follow, we will briefly explain the Viterbi attack.

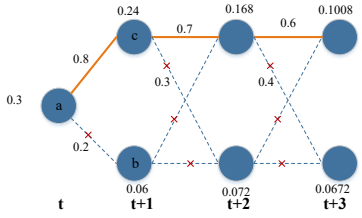

Fig. 3. An example of Viterbi

Given the queried location set $S^t, S^{t+1}, ..., S^{t+n}$, which path length is $n+1$. The aim of the attack is to find the most probable state sequence to destruct the location privacy. For example, in Fig. 3, the user at time t can move from location a, with query probability 0.3, to location b and c,

$$H_{\mathrm{t}} = -\sum_{i=1}^{k} \Pr[\frac{(\prod_{l=1}^{m-1} P[l+1|l])}{\sum_{i=1}^{k}(\prod_{l=1}^{m-1} P[l+1|l])}] * \log_2(\frac{(\prod_{l=1}^{j-1} P[l+1|l])}{\sum_{i=1}^{k}(\prod_{l=1}^{m-1} P[l+1|l])}). \quad (9)$$

with transition probabilities 0.8 and 0.2, respectively. Then the Viterbi attack can determine the location c with higher probability 0.24 as the true location. Based on this location, the adversary can search for the next true location and till the end. To this end, the Viterbi attack can search the true location and threaten the privacy effectively with side information against most algorithms.

V. Simulation Results

In this section, we experimentally evaluate the proposed algorithm, by comparing with other related algorithms in terms of the key system parameters.

A. Experiment Setup

To validate the effectiveness of our proposed scheme, we use the data collected by Geolife project, which includes the GPS trajectories of 182 users from April 2007 to August 2012 in Beijing, China. The data set contains 17621 trajectories with a total distance of 1292951 km.

We select an experiment zones in the centre of Beijing which size is *1 km* $\times$ *1 km* in the map and divide it into girds each cell *0.5 km*$\times$ *0.05 km*. We also suppose the length of a sequential query p is 2 to 8, and the number of users k related to k-anonymity is varied from 2 to 30 usually.

We compare the proposed PBDS algorithm with other schemes. The DLS algorithm is designed to achieve the privacy protection based on the stationary location query in terms of the query probability [14]. The RDG algorithm is proposed to achieve the privacy protection of continuous location query in [15]. The random algorithm generates dummy locations randomly without considering any side information. The upper bound is calculated when all locations are queried by users with the same probability of $\frac{1}{k}$, and the set entropy is the maximum entropy $h = \log_2 k$.

B. Performance Analysis

1) Set entropy vs. k: We first evaluate the relationship between the set entropy metric and the size of the dummy set k. In Fig. 4, we compare our algorithm with the DLS, RDG, random and upper bound algorithms. It can be seen in the figure, with an increasing k, the entropies of all algorithms grow continuously. In other words, enlarging the dummy set can significantly improve the privacy protection. In addition, the DLS, RDG and PBDS algorithms can achieve much better performance than the random scheme, because the random scheme generates dummies randomly without any consideration of the realistic user movement patterns. The RDG and PBDS algorithms have a lower level than the DLS algorithm, because they both sacrifice some set entropy to achieve trajectory entropy. Even though the PBDS loses some set entropy, it still has a higher level than the random

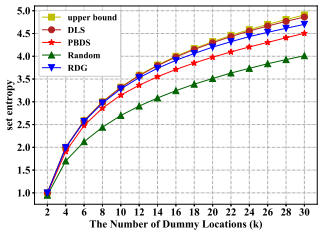

Fig. 4. Set entropy vs. k

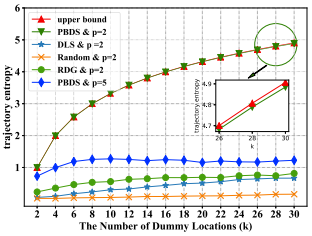

Fig. 5. Trajectory entropy vs. k

algorithm. It means the proposed PBDS can hold set entropy in continuous queries locations. Compared with the PBDS algorithm, the performance of the DLS is very close to the upper bound since it does not consider the influence of the trajectory.

2) Trajectory entropy vs. k: The metric of set entropy only considers the stationary location query, and ignores that the adversary may be able to access the trajectory information to compromise the privacy. So we compare the performance of different algorithms in terms of trajectory entropy in Fig. 5. As shown in this figure, the number of locations k varies from 2 to 30 and the path length is set to 5 to measure the trajectory entropy. The performance trend in Fig. 5 indicates that the trajectory entropy keeps rising with the increasing number of dummy locations. It can be seen that the proposed PBDS algorithm shows an excellent performance since it approaches to the upper bound when $p = 2$. This is because that the dummy path is chosen according to the possibility of the true path. Both of the DLS and RDG algorithms show a worse performance compared with our algorithm, since they ignore the probability of the trajectory. The random scheme displays a worst performance because it does not take any side information into consideration. In addition, we also compare the performance with a different path length, i.e., 2 vs. 5. The trajectory entropy is worse with a longer path length because more side information can be obtained by the adversary.

3) Percentage of privacy protection vs. k: In Fig. 6, we set path length to 5, and further discuss the performance in terms of privacy protection against the Viterbi attack. We can see that with an increasing k, the percentage of protecting the real locations is rising. It means that enlarging the dummy set is an effective method to protect the location privacy. More importantly, considering side information can improve the performance of privacy protection. For instance, when k is 14, both the PBDS and RDG algorithms show more than 2 times of privacy protection ratio than the DLS algorithm. This is because the DLS algorithm only takes limited side information into account. Besides, the proposed algorithm PBDS shows a better level in terms of privacy protection than the RDG

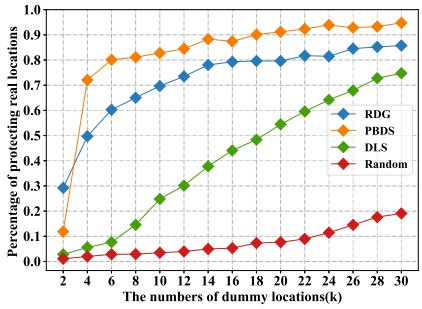

Fig. 6. The performance of all algorithms against Viterbi attack

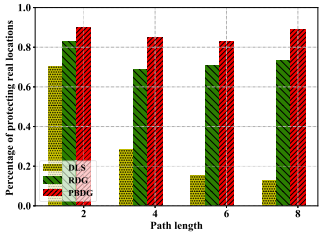

Fig. 7. Comparison of all algorithms

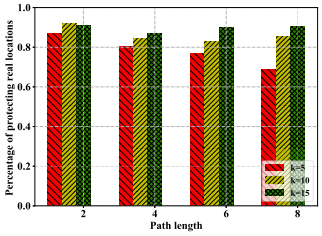

Fig. 8. The PBDS algorithm

algorithm, which means the path likelihood has a significant impact on the trajectory protection. The DLS algorithm also shows a bad performance of protecting real locations than the other two schemes since it does not consider the correlation between continuous queried locations in the trajectory.

After discussing the influence of the number of dummy locations k, we further analyze the performance in terms of path length. As shown in Fig. 7, we set k as 10 and the path length varies from 2 to 8. We compare the DLS, RDG, and PBDS schemes and discuss the performance against the Viterbi attack. When k is 10, the percentage of privacy leakage should be no more than 0.1, however, the DLS scheme apparently has inferior performance in protecting privacy. By contrast, the PBDS algorithm can achieve a better performance than others with considering the path probability. For instance, when path length is 4, the privacy protection percentage of PBDS algorithm enhances almost 60% than the DLS algorithm and 20% than the RDG algorithm, respectively. With the increasing path length, the DLS algorithm also shows a descending trend in privacy protection, since the adversary can infer the connection between continuous queried locations in a trajectory to compromise the protection scheme.

At last, we analyze the performance of the proposed algorithm PBDS with different k in Fig. 8. It can be seen that when $k = 5$, with an increasing path length, the performance of privacy protection drops. However, this performance does not show an obvious decline when k is 10 or 15. It means that with an increasing path length, enlarging the dummy set is an effective method to mitigate privacy attacks. Given a simple case, when path length is 2, increasing the number of locations does not show a satisfactory performance of privacy protection. However, when path length is 8, we can clearly see an opposite case.

VI. Conclusions

In this work, we have proposed an algorithm named trajectory entropy to protect the location privacy. The proposed algorithm is based on the query and transition probabilities, which are assumed to be known by the adversary as side information. Simulation results show that, the proposed algorithm outperforms the existing algorithms. The results also indicate that the proposed algorithm can achieve a better performance with a larger dummy set. However, as the path length increases, there is an increasing probability that the location information can be inferred by the adversary, which will be addressed in our future work.

References

[1] J. Li, Y. Chen, Z. Lin, C. Wen, and L. Hanzo, "Distributed caching for data dissemination in the downlink of heterogeneous networks," *IEEE Transactions on Communications*, vol. 63, no. 10, pp. 1–1, 2015.

[2] J. Li, C. He, Y. Chen, Z. Lin, B. Vucetic, and L. Hanzo, "Pricing and resource allocation via game theory for a small-cell video caching system," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 8, pp. 2115–2129, 2016.

[3] C. A. Ardagna, M. Cremonini, E. Damiani, S. D. C. D. Vimercati, and P. Samarati, "Location privacy protection through obfuscation-based techniques," in *IFIP Annual Conference on Data and Applications Security and Privacy*, 2007.

[4] S. Mascetti, D. Freni, C. Bettini, X. S. Wang, and S. Jajodia, "Privacy in geo-social networks: proximity notification with untrusted service providers and curious buddies," *Vldb Journal*, vol. 20, no. 4, pp. 541–566, 2011.

[5] C. A. Ardagna, M. Cremonini, E. Damiani, S. D. C. D. Vimercati, and P. Samarati, "Location privacy protection through obfuscation-based techniques," in *Ifip Wg 113 Working Conference on Data and Applications Security*, 2007.

[6] A. Gutscher, "Coordinate transformation - a solution for the privacy problem of location based services?" in *International Parallel and Distributed Processing Symposium*, 2006.

[7] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.

[8] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking," in *Proceedings of the 1st international conference on Mobile systems, applications and services*. ACM, 2003, pp. 31–42.

[9] C. Y. Chow, M. F. Mokbel, and X. Liu, "A peer-to-peer spatial cloaking algorithm for anonymous location-based service," in *Acm International Symposium on Advances in Geographic Information Systems*, 2006.

[10] B. Niu, X. Zhu, H. Chi, and H. Li, "3plus: Privacy-preserving pseudo-location updating system in location-based services," in *Wireless Communications and Networking Conference*, 2013.

[11] H. Lu, C. S. Jensen, and L. Y. Man, "Pad:privacy-area aware, dummy-based location privacy in mobile services," in *Acm International Workshop on Data Engineering for Wireless and Mobile Access*, 2008.

[12] C. Y. T. Ma, D. K. Y. Yau, N. K. Yip, and N. S. V. Rao, "Privacy vulnerability of published anonymous mobility traces," *IEEE/ACM Transactions on Networking*, vol. 21, no. 3, pp. 720–733, 2013.

[13] B. Niu, Z. Zhang, X. Li, and H. Li, "Privacy-area aware dummy generation algorithms for location-based services," in *IEEE International Conference on Communications*, 2014.

[14] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, "Achieving k-anonymity in privacy-aware location-based services," in *IEEE Infocom*, 2014.

[15] S. Shaham, M. Ding, B. Liu, Z. Lin, and J. Li, "Privacy preservation in location-based services: A novel metric and attack model," *arXiv preprint arXiv:1805.06104*, 2018.

[16] C. E. Shannon, "Communication theory of secrecy systems," *Bell System Technical Journal*, 1949.