A Hybrid Spatiotemporal Attack in Continuous LBS Queries

1st Yongjie Zhan
School of Electronic and Optical Engineering
Nanjing University of Science and Technology
Nanjing, China
a1002180276@outlook.com

2nd Le Li
School of Electronic and Optical Engineering
Nanjing University of Science and Technology
Nanjing, China
1985308669@qq.com

3rd Yuwen Qian
School of Electronic and Optical Engineering
Nanjing University of Science and Technology
Nanjing, China
admon@njust.edu.cn

4th Chuan Ma
School of Electronic and Optical Engineering
Nanjing University of Science and Technology
Nanjing, China
chuan.ma@njust.edu.cn

5th Ming Ding
Data61
CSIRO
Sydney, Australia
ming.ding@data61.csiro.au

6th Bo Liu
School of Computer Science
University of Technology Sydney
NSW, Australia
liubo.lb2006@gmail.com

***Abstract*—In recent years, location-based services (LBS) enjoy a rapidly increasing popularity among the various mobile applications. As a result, how to protect users' location privacy has become an urgent issue. To alleviate this issue, the dummy location selection (DLS) algorithm based on the k-anonymity criterion has been studied in many existing works, and it can provide protection against the adversary with the query probability information. However, such a method losses its effectiveness when facing the hybrid Retrospect attack proposed in this paper. The proposed attacking method is designed to capture the spatial continuity between adjacent locations along with the temporal information in continuous queries of the whole movement trajectory. To demonstrate the effectiveness of the proposed attacking method, two real-life datasets are evaluated in comparison with the state-of-the-art algorithms.**

***Index Terms*—k-anonymity, continuous queries, privacy-preserving, location-based services**

I. Introduction

With the rapid development of the Internet-based applications and the popularity of smart mobile terminals, location-based services (LBS) enjoy a burgeoning development in real-life applications. Typical instances include finding a nearby parking lot, querying the fastest route to a destination, or checking local weather. However, privacy risks exist when users sending their personal information to the LBS providers, and an adversary may infer sensitive information from the collected data. Therefore, how to prevent the privacy leakage in the LBS becomes a hot research topic.

In [1], Samarati et al. proposed a raw data protection method called the k-anonymity criterion, which is designed to hide the sensitive record in other $k-1$ similar records. This method is first applied in the location privacy protection in [2] that dummy locations with the real one are sent to the LBS service provider. Then, the disclosure probability can be decreased to $\frac{1}{k}$ in the single query. Several following up works [3], [4] based on the k-anonymity were then proposed to protect LBS privacy. Nevertheless, there is no definite answer about how to choose dummy locations appropriately, and the authors in [5] indicated that the dummy locations generated by ignoring geographic location are more than 50% implausible. Therefore, Yamin *et al.* in [3] selected the dummy locations according to the spatial multi-swapping scheme involving peers and fog nodes. Additionally, the dummy-location selection (DLS) algorithm was proposed in [4], which can select a required number of dummy locations that closes to the user to achieve the maximum information entropy in terms of the query probability. In detail, a query means the user launches a LBS request, and the DLS algorithm can achieve a $\frac{k-1}{k}$ privacy protection performance in one single query. However, it losses the effectiveness in the scenario of continuous queries, in which time correlation exists in adjacent queries [6]. To address this issue, the concept of cloaking region in continuous queries was introduced in [7], and it can resist attacks by sending a wide range of regional locations instead of accurate location.

With the development of defensive methods, how to discriminate fake locations also attracts increasingly attentions. With the help of the spatial relationship, the location-dependent attack (LDA) was proposed in [8], which utilizes the distance constraints between anonymity sets (ASs) at adjacent queries to exclude part of unreasonable dummy locations. Note that, although the time series factor is considered in the LDA, it can only realize the single query attack in continuous queries.

Recently, Shaham *et al.* [6] applied the Viterbi attack in the hidden markov model (HMM) which can fully capture the characteristics of the temporal information in continuous queries. By this method, the multiple location disclosure problem can be transformed into a path selection problem in ASs and related analysis is conducted. To tackle the Viterbi attack, they then proposed a defensive method based on the transition entropy. Nevertheless, due to the limitation of the HMM, the Viterbi attack can only consider the influence of the previous moments on the subsequent moments but ignores the extra information that the subsequent moments bring to the previous ones.

With the consideration of the time factor, the DLS algorithm is no longer able to cope with the challenge of protecting user location privacy. Therefore, in this work, we propose an attacking method called the Retrospect attack, which combines the spatial and time factors to find the real trajectory of the user in continuous queries. The main contributions of this paper are as follows.

- We propose a hybrid retospect attack that considers the impact of subsequent time stamps on previous stamps combined with the HMM. This algorithm can analyze the real location of the user at a time stamp by propagating spatial attack (PSA) and then find the movement trajectory throughout the continuous queries by the sequent time attack (STA).
- We validate the effectiveness of the proposed attacking method by evaluating the comparison on two real-life datasets with the state-of-the-art algorithms.

The remainder of the paper is organized as follows. Section II introduces the system model, including the system architecture, the adversary model and the privacy metrics used in the paper. Section III illustrates the proposed Retrospect attack algorithm. The analysis of the proposed metrics and algorithms are provided in Section IV. Finally, we conclude our work in Section V.

II. System Model

A. System Architecture

In this paper, the system is composed of three entities: users, the anonymizer and one LBS server, as shown in Fig. 1.

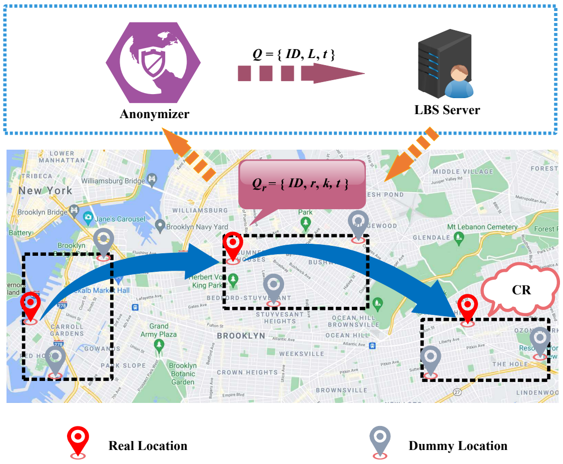


Fig. 1. Systey Model. The user sends the anonymity query request to the LBS providers through the anonymizer, and then the LBS providers return the request results to the user.

1) Users: One user intends to send an original query request $Q_r^{t_i} = \{ID, r, k, t_i\}$ to the LBS server, where ID represents the user's unique identity, $r = (x, y)$ represents the real location of the user, k donates the privacy requirement that the disclosure probability of the real location should be $\frac{1}{k}$ [2], and t_i represents the current time stamp, where $i \in (1, 2, \dots, c)$ and c represents the total number of queries.

2) Anonymizer: Upon receiving a user's request $Q_r^{t_i}$, the anonymizer generates $k-1$ virtual requests to build the anonymity set (AS). The real request and the virtual requests constitute the new request set $Q^{t_i} = \{ID, \boldsymbol{L}, t_i\}$, where $\boldsymbol{L}$ represents the AS. In continuous queries, the AS at time stamp t_i can be expressed as

$$\boldsymbol{L}^{t_i} = (e_1^{t_i}, e_2^{t_i}, \dots, e_{k^{t_i}}^{t_i}), \tag{1}$$

where $e_j^{t_i}$ represents the real or dummy location and k^{t_i} represents the privacy requirement at the time stamp t_i. In addition, the minimum area that surrounds $\boldsymbol{L}^{t_i}$ denotes the cloaking region (CR^{t_i}). After the process of the anonymizer, Q^{t_i} is sent to LBS server.

3) LBS Server: The LBS server receives Q^{t_i} and returns all the query results of $\boldsymbol{L}^{t_i}$ to the user.

B. Adversary Model

In this paper, we consider that the LBS server is honest-but-curious. It may use additional background knowledge to dig the users' private information. We assume that the background knowledge includes the following.

1) Maximum Moving Velocity: The adversary has the speed limitation of different regions in the map. Therefore, the adversary can estimate the maximum moving distance within each time interval in continuous queries [8]. By comparing the maximum moving distance with the real distance between two ASs, the adversary can exclude some unreasonable dummy locations.

2) Hidden Markov Model (HMM): HMM is used to describe a markov process with hidden parameters, which is assumed the homogeneous markov hypothesis in this paper.

Definition 1. *Homogeneous markov hypothesis: A state at current time stamp of HMM only depends on the state at the previous time stamp.*

For instance, suppose W is a sequence of states of length M, where $W = (w_1, w_2, \cdots, w_M)$, and O is the corresponding observation sequence. Then the homogeneous markov hypothesis can be expressed as

$$Pr(w_t | w_{t-1}, o_{t-1}, \cdots, w_1, o_1) = Pr(w_t | w_{t-1}). \quad (2)$$

According to (2), when the adversary knows the real location at the previous time stamp, he can select a most likely location at current time stamp [6], [9].

C. Performance Metrics of Privacy

Since the privacy requirement k is only suitable to evaluate the location disclosure risk in a single query, it restricts the accuracy to estimate the trajectory disclosure probability in continuous queries. In this section, we introduce a metric called the trajectory disclosure probability, which measures the privacy level in continuous queries. We first explain the metrics including the Hausdorff distance, the trajectory hit ratio and the trajectory probability used in continuous queries. The Hausdorff distance is a measurement of the distance between two regions. The trajectory hit ratio measures the successful rate of the attacking in one time, and based on the trajectory probability, the adversary can find the real trajectory of the user. Then the privacy expression is obtained with the expectation of trajectory hit ratios.

1) Hausdorff Distance: Consider two cloaking regions CR^{t_i} and $\text{CR}^{t_{i+1}}$, the Hausdorff distance [10] represents the regional distance between CR^{t_i} and $\text{CR}^{t_{i+1}}$. It can be formally expressed as

$$\begin{aligned} &d_H(\text{CR}^{t_i}, \text{CR}^{t_{i+1}}) \\ &\quad = \max\left\{d_h(\text{CR}^{t_i}, \text{CR}^{t_{i+1}}), d_h(\text{CR}^{t_{i+1}}, \text{CR}^{t_i})\right\}, \end{aligned} \quad (3)$$

where

$$d_h(S_1, S_2) = \max_{p_1 \in S_1} \min_{p_2 \in S_2} d(p_1, p_2). \quad (4)$$

In (4), $d(p_1, p_2)$ represents the distance between two locations p_1 and p_2.

2) Trajectory Hit Ratio: There are several locations in one trajectory, and we define the trajectory hit ratio as the number of real locations chosen by the adversary in all the locations. Before formally presenting this expression, we first introduce the definition of the real trajectory and possible trajectories.

Definition 2. *(Real Trajectory) The real trajectory of the user in continuous queries is expressed as*

$$\boldsymbol{R} = (r^{t_1}, r^{t_2}, \cdots, r^{t_c}). \quad (5)$$

Definition 3. *(Possible Trajectories) From the time stamp t_1 to t_c, the α-th possible trajectory of the user can be expressed as*

$$\boldsymbol{T}_\alpha = (l_\alpha^{t_1}, l_\alpha^{t_2}, \cdots, l_\alpha^{t_c}), \quad (6)$$

where $1 \leq \alpha \leq \prod_{i=1}^{c} k^{t_i}$ and $l_\alpha^{t_i} \in \boldsymbol{L}^{t_i}$.

Note that, for any $\boldsymbol{T}_\alpha, \boldsymbol{T}_\beta$ and $\alpha \neq \beta$, there exists at least a time stamp t_i that satisfies the condition $l_\alpha^{t_i} \neq l_\beta^{t_i}$.

We then use N_i to indicate whether the adversary can obtain the real location at the t_i-th time stamp in a possible trajectory, and it can be expressed as

$$N_i = \begin{cases} 1, & l_\alpha^{t_i} = r^{t_i}; \\ 0, & \text{others}. \end{cases} \quad (7)$$

As a result, the trajectory hit ratio can be calculated by

$$\eta = \frac{\sum_{i=1}^{c} N_i}{c}. \quad (8)$$

Additionally, the low privacy level represents a higher trajectory hit ratio η.

3) Trajectory Probability: Based on the trajectory hit ratio, we then define the trajectory probability as the probability that most clients choose one trajectory as the real trajectory from all possible trajectories, which consists of two components.

The first component is the probability that a client chooses a location $l_\alpha^{t_1}$ as the origin point in the real trajectory. It can be expressed by the normalized query probability and written as

$$\hat{p}(l_\alpha^{t_i}) = \frac{p(l_\alpha^{t_i})}{\sum_{j=1}^{k^{t_i}} p(e_j^{t_i})}, \quad (9)$$

where $p(l_\alpha^{t_i})$ represents the query probability on $l_\alpha^{t_i}$.

The second component is the probability that the client moves along this trajectory from the origin point $l_\alpha^{t_1}$. It can be expressed by the product of posterior probabilities as

$$\hat{q}(l_\alpha^{t_{i+1}} | l_\alpha^{t_i}) = \frac{q(l_\alpha^{t_{i+1}} | l_\alpha^{t_i})}{\sum_{j=1}^{k^{t_{i+1}}} q(e_j^{t_{i+1}} | l_\alpha^{t_i})}, \quad (10)$$

where $q(l_\alpha^{t_{i+1}} | l_\alpha^{t_i})$ represents the transition probability from $l_\alpha^{t_i}$ to $l_\alpha^{t_{i+1}}$.

Therefore, the trajectory probability can be expressed as

$$\begin{aligned} Pr(\boldsymbol{T}_\alpha) &= Pr(l_\alpha^{t_1}, l_\alpha^{t_2}, \cdots, l_\alpha^{t_c}) \\ &= Pr(l_\alpha^{t_1}) Pr(l_\alpha^{t_2}, \cdots, l_\alpha^{t_c} | l_\alpha^{t_1}) \\ &= Pr(l_\alpha^{t_1}) Pr(l_\alpha^{t_2} | l_\alpha^{t_1}) Pr(l_\alpha^{t_3} | l_\alpha^{t_1}, l_\alpha^{t_2}) \\ &\quad \cdots Pr(l_\alpha^{t_c} | l_\alpha^{t_1}, l_\alpha^{t_2}, \cdots, l_\alpha^{t_{c-1}}), \end{aligned} \tag{11}$$

According to the Homogeneous markov hypothesis in **Definition 1**, Equation (11) can be further simplified as

$$\begin{aligned} Pr(\boldsymbol{T}_\alpha) &= Pr(l_\alpha^{t_1}) Pr(l_\alpha^{t_2} | l_\alpha^{t_1}) Pr(l_\alpha^{t_3} | l_\alpha^{t_2}) \cdots Pr(l_\alpha^{t_c} | l_\alpha^{t_{c-1}}) \\ &= \hat{p}(l_\alpha^{t_1}) \ \hat{q}(l_\alpha^{t_2} | l_\alpha^{t_1}) \ \hat{q}(l_\alpha^{t_3} | l_\alpha^{t_2}) \cdots \hat{q}(l_\alpha^{t_c} | l_\alpha^{t_{c-1}}). \end{aligned} \tag{12}$$

III. RETROSPECT ATTACK

In this section, we propose the Retrospect attack that can analyze the real location at one time stamp by the propagating spatial attack (PSA) and then find the real trajectory by the sequent time attack (STA).

A. Propagating Spatial Attack

Traditional spatial attacks in [8], [11] use a reasonable moving regions of clients within adjacent time stamps to exclude fake locations. However, they do not consider the influence of the reduced region after excluding some fake locations on the original spatial relationship. Therefore, in this subsection we first enhance the spatial attack, which leads to the propagating spatial attack (PSA).

In more detail, when some fake locations are excluded at the current time stamp, certain fake locations at the previous and next time stamps may be identified by the decreased Hausdorff distance. Before fully presenting the PSA, we will first introduce two relevant definitions as follows.

Definition 4. *The maximum movement boundary (MMB) refers to the maximum region that the client in CR^{t_i} can reach in a time interval $\triangle t_i$, where $\triangle t_i = t_{i+1} - t_i$. The maximum movement region is denoted as MR^{t_i}.*

For instance, as shown in Fig. 2, there are three ASs in the continuous queries. The solid line regions represent CR^{t_1}, CR^{t_2} and CR^{t_3}, and each of them consists of the real location and dummy locations. In addition, the dotted line region represents MR^{t_i} which is extended from CR^{t_i} at a maxium moving distance of $d_{\text{max}}^{t_i}$. Fig. 2(a) indicates that the locations in the green solid line region CR^{t_3} but outside of the green dotted line region MR^{t_2} will be recognized as fake locations by the adversary.

Definition 5. *The maximum arrival boundary (MAB) refers to the maximum region constituted by all real locations that the client can reach to $CR^{t_{i+1}}$ after a time interval $\triangle t_i$, and it is denoted as $AR^{t_{i+1}}$.*

For convenience, we suppose all $d_{\text{max}}^{t_i}$ are equal in Fig. 2[1]. Then the dotted line region MR^{t_i} also represents the AR^{t_i} which is extended from CR^{t_i} at a maximum moving distance of $d_{\text{max}}^{t_{i-1}}$. Fig. 2(a) indicates that the locations in the orange solid line region CR^{t_2} but outside of the orange dotted line region AR^{t_3} will be recognized as fake locations by the adversary.

From **Definition 4** and **Definition 5**, we can observe that, if $d_H(\text{CR}^{t_i}, \text{CR}^{t_{i+1}}) > d_{\text{max}}^{t_i}$, the PSA works on the two ASs from time stamp t_i to t_{i+1}.

After the comparison with $d_H(\text{CR}^{t_i}, \text{CR}^{t_{i+1}})$ and $d_{\text{max}}^{t_i}$ on all time stamps, the adversary can recognize and exclude three gray fake locations in Fig. 2(a). Then as shown in Fig. 2(b), the area of CR^{t_2} and CR^{t_3} both decrease. By repeating the attack, we can observe that there are two red fake locations revealed in CR^{t_1} but outside of AR^{t_2}, while the adversary cannot exclude them with the traditional attack [7], [8]. It means that in comparison with traditional methods, the PSA can further explore spatial relationships between ASs. The PSA stops as shown in Fig. 2(c), when the adversary can no longer recognize any fake location for all time stamps. After the PSA attack, CR^{t_i} should satisfy the following condition:

$$\text{CR}^{t_i} \subseteq \left(\text{MR}^{t_{i-1}} \cap \text{AR}^{t_{i+1}} \right). \tag{13}$$

Then the remaining valid locations in the AS at the time stamp t_i are denoted as

$$\boldsymbol{LP}^{t_i} = (lp_1^{t_i}, lp_2^{t_i}, \ldots, lp_{k_v^{t_i}}^{t_i}), \tag{14}$$

where $1 \leq k_v^{t_i} \leq k^{t_i}$. Additonally, $k_v^{t_i} = 1$ indicates $\boldsymbol{LP}^{t_i}$ only has the real location. The privacy disclosure probability of the client will increase with a decrease of the $k_v^{t_i}$.

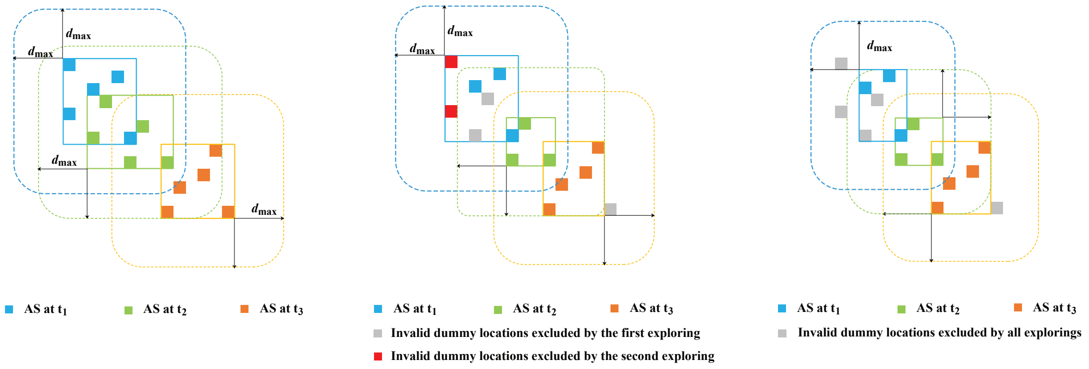
The PSA algorithm is given in Alg. 1. In lines 2-6, the adversary searches all time stamps to find if there exists unreasonable locations and excludes them. The loop ends when no fake location is excluded from the remaining locations.

B. Sequent Time Attack

After the PSA, the probability that a location in $\boldsymbol{LP}^{t_i}$ is the real location at the current time stamp is only related to (9), and is independent of the locations in $\boldsymbol{LP}^{t_i}$ at the subsequent time stamps. Therefore, the adversary can assume the movement of the client follows an HMM. Then the proposed algorithm focuses on how to find the real trajectory from the the remaining locations.

When the destination of the real trajectory is determined, the adversary can choose a target trajectory as the real trajectory with the maximum trajectory

[1] This assumption will not affect the generality of the derived results.


$d_{\max}$
AS at t_1
AS at t_2
AS at t_3
Invalid dummy locations excluded by the first exploring
Invalid dummy locations excluded by the second exploring
Invalid dummy locations excluded by all explorings

(a) The first exploring in PSA.

(b) The second exploring in PSA.

(c) The third exploring in PSA.

Fig. 2. The example of the adversary use the PSA in continuous queries with a total time stamps of three.

Algorithm 1: PSA Algorithm

```
Input: Continuous queries ASs: $\boldsymbol{L}^{t_i}$
Output: Remaining valid locations: $\boldsymbol{LP}^{t_i}$
while True do
    if all $d_H(\mathrm{CR}^{t_i}, \mathrm{CR}^{t_{i+1}}) <= d^{t_i}_{\max}$ then
        break ;
    else
        $\boldsymbol{LP}^{t_i} = \boldsymbol{L}^{t_i}$.remove(fake locations) ;
    end
end
return $\boldsymbol{LP}^{t_i}$
```

probability in all possible trajectories. Nevertheless, based on the available information, the adversary cannot recognize the real destination at time stamp t_c. Therefore, we assume the real trajectory is among $k_v^{t_c}$ retrospective trajectories, and the retrospective trajectories are defined as follows:

Definition 6. *(Retrospective trajectories) Searching forward from each $lp_j^{t_c}$, a target trajectory with the maximum trajectory probability can be found and is called the retrospective trajectory, where t_c represents the last time stamp. Additionally, the β-th retrospective trajectory is denoted as $\boldsymbol{T}'_\beta$, where $\boldsymbol{T}'_\beta \subset \boldsymbol{T}_\alpha$.*

The proposed sequent time attack (STA) is to find $k_v^{t_c}$ retroactive trajectories by comparing the probabilities of the possible trajectories that do not consist of fake locations in all time stamps. Then the algorithm selects one of them as the real trajectory by the probability selection method according to their normalized trajectory probabilities. Since the trajectory probability is a statistical analysis of clients, which cannot reflect the movement pattern of a single client. When one retrospective trajectory probability exceeds others by a predefined threshold (δ), the STA will select it as the result. When several retrospective trajectories have similar probabilities, the STA will probably choose the result from these N candidate trajectories. Additionally, when $\boldsymbol{T}'_\beta$ is in these candidate trajectories, the selection probability for $\boldsymbol{T}'_\beta$ as a result can be expressed as

$$Pr(T = \boldsymbol{T}'_\beta) = \frac{Pr(\boldsymbol{T}'_\beta)}{\sum_\Phi Pr(\boldsymbol{T}'_\zeta)}, \qquad (15)$$

where Φ represents the set of the N candidate trajectories.

For example, as shown in Fig. 3, the gray points are the fake locations excluded by the PSA. Since $\boldsymbol{LP}^{t_5}$ has four remaining locations, the adversary can get four retrospective trajectories. For instance, retrospective trajectory 1 has the highest trajectory probability in all possible trajectories that will reach location E1 at time stamp t_5. Based on a predefined δ, the algorithm can determine N and the corresponding set Φ. Then it can select the one by the probability selection method from the N retrospective trajectories and consider it as the real trajectory.

In conclusion, we propose the Retrospect attack, which contains two parts PSA and STA. Additionally, the Retrospect attack algorithm is given in Alg. 2. In line 2, the algorithm initializes the normalized query probabilities of all $lp_j^{t_1}$. Then starting from the time stamp t_2, $u[i][j]$ stores the maximum trajectory probability through $lp_j^{t_i}$ from the time stamp t_1 to t_i in line 5, and $\varphi[i][j]$ stores the coordinate of the trajectory in $u[i][j]$ at the previous time stamp in line 6. Note that, $u[i][j]$ only needs to consider the influence of all $u[i-1][j]$. When the algorithm gets all $u[i][j]$, it can calculate the retrospective trajectory probabilities $u[c][j]$ in line 10, and it can select all retrospective trajectories $\boldsymbol{T}'_\beta$ via $\varphi[c][j]$ in lines 13-16. Finally, in lines 18-19, the algorithm returns one trajectory as the real trajectory with the probability selection.

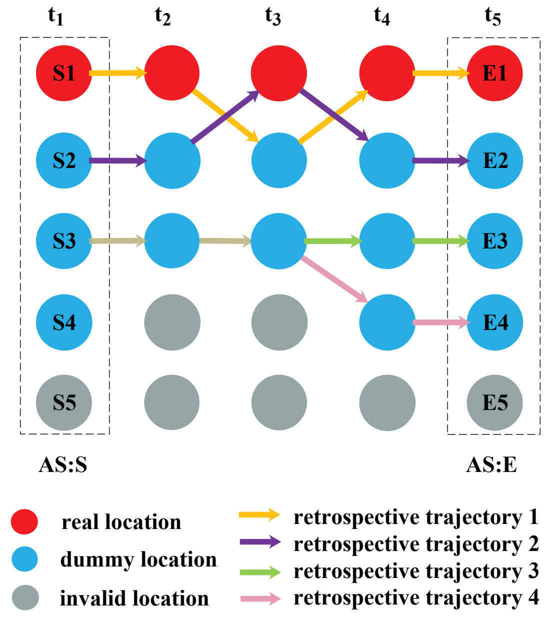

t1
t2
t3
t4
t5
S1
S2
S3
S4
S5
E1
E2
E3
E4
E5
AS:S
AS:E
real location
dummy location
invalid location
retrospective trajectory 1
retrospective trajectory 2
retrospective trajectory 3
retrospective trajectory 4

Fig. 3. The example of choosing retroactive trajectories that end with different $lp_j^{t_5}$ at time stamp t_5.

IV. Experimental Results

In this section, we evaluate the performance of the proposed Retorspect attack with/without the STA on two real-life datasets. First, we compared the performance of the proposed Retorspect attack without the STA with the LDA [8] in terms of the time stamps number (c) and privacy requirements (k), respectively, against the traditional DLS defensive method [4]. Second, we compare the proposed Retorspect attack with the STA, and the Viterbi attacks [6] in the same way. Additionally, we set the predefined threshold value (δ) to 0.1 in the Retrospect attack.

A. *Experiment Setup*

- Datasets: In our experiment, we ues the the Geolife project trajectory [12]–[14] and T-Drive dataset [15], [16].

  In the Geolife project trajectory dataset, the majority of the data was created in Beijing, China, and was collected by Microsoft Research Asia from April 2007 to August 2012. It contains 17,621 trajectories with a total distance of 1,292,951 kilometers and a total duration of 50,176 hours. Moreover, these trajectories were recorded a broad range of users' outdoor movements, including not only life routines like daily work routine but also some entertainments and sports activities, such as shopping, sightseeing, dining, hiking and cycling by different GPS loggers and GPS-phones, and have a variety of sampling rates.

Algorithm 2: Retrospect Attack Algorithm

```
Input: Continuous queries ASs: $\boldsymbol{L}^{t_i}$, the
       normalized query probability $\hat{q}(e_j^{t_1})$,
       matrix of transition probability $A^{t_i}$ and the
       threshold value $\delta$
Output: The client's most likely trajectory: $T$
$\boldsymbol{LP}^{t_i} \longleftarrow \boldsymbol{L}^{t_i}$ via the PSA ;
$u[1][j] = \hat{q}(lp_j^{t_1})$ ;
for $i$ in range(2, $c$+1) do
    for $y$ in range( $k_v^{t_{i+1}}$ ) do
        $u[i][y] = \max_{1 \le x \le k_v^{t_i}} u[i-1][x] * A^{t_i}[x][y]$ ;
        $\varphi[i][y] = \arg\max_{1 \le x \le k_v^{t_i}} u[i-1][x] * A^{t_i}[x][y]$ ;
    end
end
for $\beta$ in range( $k_v^{t_c}$ ) do
    $protra[\beta] = u[c][\beta]$ ;
    $T'[\beta][c] = lp_\beta^{t_c}$ ;
    $j = \beta$ ;
    for $i$ in range($c-1, 0, -1$) do
        $T'[\beta][i] = lp^{t_i}_{\varphi[i+1][j]}$ ;
        $j = \varphi[i+1][j]$ ;
    end
end
select one trajectory $T \longleftarrow N$ candidate
trajectories from all $\boldsymbol{T}'_\beta \longleftarrow$ the threshold value $\delta$ ;
return $T$
```

  The T-Drive dataset contains the GPS trajectories of 10,357 taxis from Feb. 2 to Feb. 8, 2008, within Beijing. The total number of locations in this dataset is about 15 million and the total distance of the trajectories reaches 9 million kilometers.

- Experiment Parameters: We divide a 30 $km\times$ 30 km regional grid area in Beijing and divide different numbers of grids in two datasets. The length of each grid and the selection of sampling frequency of each dataset are given in Table I. Therefore, the client can across the same number of grids with the same speed in both datasets in the adjacent time stamps.

TABLE I
Experiment Parameters on Two Datasets

	Geolife Project Trajectory Dataset	***T-Drive Dataset***
The Length of Each Grid Cell (m)	10	100
Sampling Interval (min)	1	10

TABLE II
THE BASELINE OF THE LOCATION DISCLOSURE PROBABILITY OF THE DLS ALGORITHM

Privacy Requirement (k)	5	10	15	20
Location Disclosure Probability	20%	10%	6.7%	5%

B. Evaluation Metrics

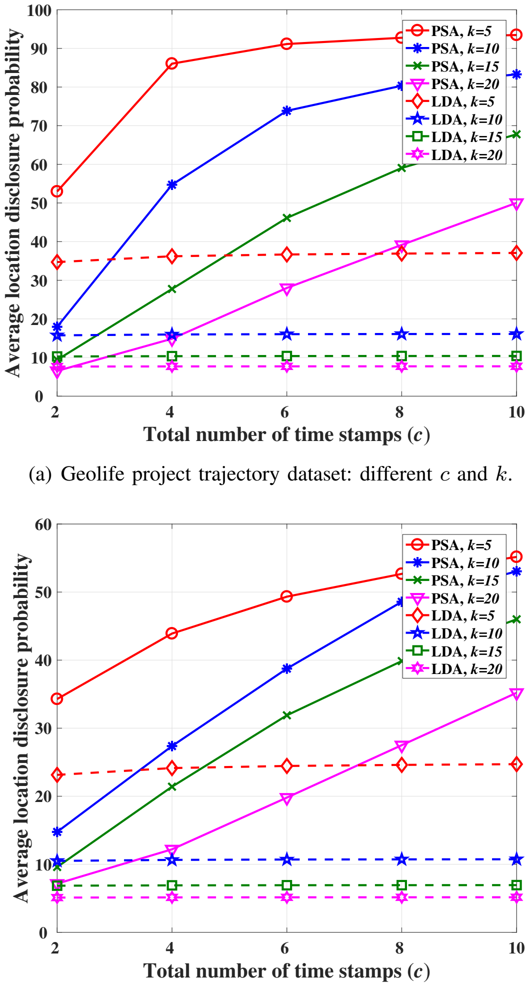
In this subsection, we introduce the evaluation metric including the average location disclosure probability and the average trajectory disclosure probability.

- Average Location Disclosure Probability: The average location disclosure probability measures the probability that the adversary reveals the real location at one time stamp. It will be used to measure the performance of the proposed Retrospect attack without the STA method and the LDA, respectively.
- Average Trajectory Disclosure Probability: The average trajectory disclosure probability measures the accuracy that the adversary finds the real trajectory. It will be used to compare the experimental results of the proposed complete Retrospect attack, the LDA and the Viterbi attack algorithms.

C. Performance of The Attacking Algorithm

In this subsection, we evaluate the performance of the proposed Retrospect attack method with/without the STA against the DLS defensive algorithm, and compare them with the results of the Viterbi attack and the LDA, respectively. Table II lists the baseline of the location disclosure probability in the DLS algorithm. It can be observed that the location disclosure probability is $\frac{1}{k}$.

1) The Retrospect Attack without the STA: Fig. 4 plots the experimental results of the location disclosure probability of the proposed Retrospect attack method without the STA (PSA), and the LDA methods against the DLS algorithm. First, it shows that the proposed PSA method has a higher average location disclosure probability than the LDA in the case of the same time stamps number (c) or privacy requirement (k). Second, we can observe that the proposed PSA method has an obvious growth trend on both datasets while the LDA curve almost keeps stable. For example, when $k = 5$, with c increases from 2 to 10, the result of the proposed PSA enhances 1.5 times, while that of the LDA enhances only 2%. This is because the PSA takes into account that the reduced of CR^{t_i} has a propagation effect on the spatial relationship of locations. When CR^{t_i} is reduced due to the presence of fake locations recognized by the adversary, it will decrease the Hausdorff distance between the CR^{t_i} and $\text{CR}^{t_{i+1}}$, and between the CR^{t_i} and $\text{CR}^{t_{i-1}}$. Then, the reduced Hausdorff distance may lead to more fake locations recognized at the time stamps t_{i-1} and t_{i+1}, which will change the Hausdorff distance at the adjacent time stamps with them again to find more fake locaions until all CR^{t_i} remain the same.

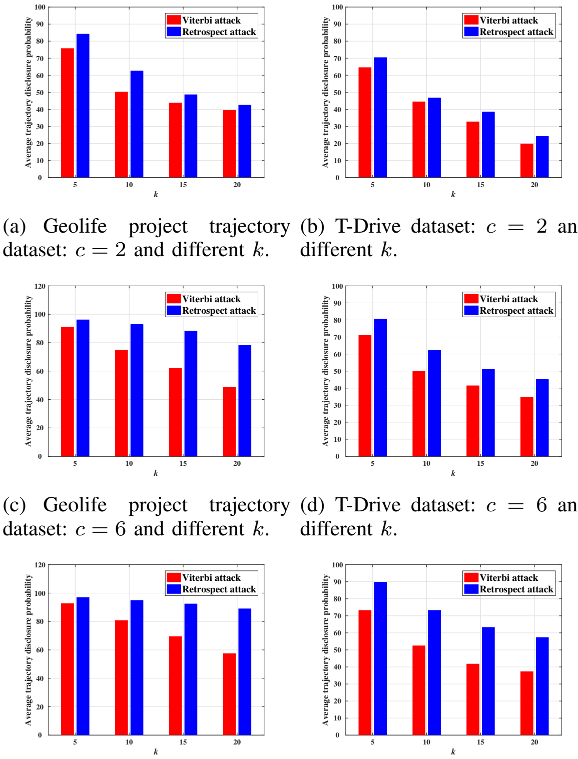


(a) Geolife project trajectory dataset: different c and k.

(b) T-Drive dataset: different c and k.

Fig. 4. The comparison of the average location disclosure probability between the proposed Retorspect attack without the STA (PSA), and the LDA against the DLS defensive algorithm with the different time stamps number (c) and privacy requirements (k) on two datasets.

2) The Retrospect Attack with the STA: Fig. 5 plots the experimental results of the trajectory disclosure probability of the proposed complete Retrospect attack and the Viterbi attack against the DLS algorithm. It shows that the performance of the proposed Retrospect attack is better than that of the Viterbi attack in terms of the trajectory disclosure probability. For example, when $c = 10$ and $k = 20$ on Fig. 5(e), we can observe that the result of the proposed Retrospect attack is about 30% higher than that of the Viterbi attack.



(a) Geolife project trajectory dataset: $c = 2$ and different k.

(b) T-Drive dataset: $c = 2$ and different k.

(c) Geolife project trajectory dataset: $c = 6$ and different k.

(d) T-Drive dataset: $c = 6$ and different k.

(e) Geolife project trajectory dataset: $c = 10$ and different k.

(f) T-Drive dataset: $c = 10$ and different k.

Fig. 5. The comparison of the average trajectory disclosure probability between the proposed Retorspect attack with the STA, and the Viterbi attack against the DLS defensive algorithm with the different time stamps number (c) and privacy requirements (k) on two datasets.

V. Conclusion

In this paper, we proposed a hybrid Retrospect attack algorithm, which can discern the real location of the user in one time stamp by the propagaing spatial attack and then find the real trajectory throughout the continuous queries by the sequent time attack. Finally, the performance of the algorithms were validated via extensive experiments on two real-life datasets with the comparison of state-of-the-art algorithms.

References

[1] P. Samarati and L. Sweeney, "Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression," 1998.

[2] 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.

[3] M. Yamin and A. A. Abi Sen, "A new method with swapping of peers and fogs to protect user privacy in iot applications," *IEEE Access*, vol. 8, pp. 210 206–210 224, 2020.

[4] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, "Achieving k-anonymity in privacy-aware location-based services," in *IEEE INFOCOM 2014-IEEE Conference on Computer Communications*. IEEE, 2014, pp. 754–762.

[5] X. Li, Y. Ren, L. T. Yang, N. Zhang, B. Luo, J. Weng, and X. Liu, "Perturbation-hidden: Enhancement of vehicular privacy for location-based services in internet of vehicles," *IEEE Transactions on Network Science and Engineering*, 2020.

[6] S. Shaham, M. Ding, B. Liu, S. Dang, Z. Lin, and J. Li, "Privacy preservation in location-based services: a novel metric and attack model," *IEEE Transactions on Mobile Computing*, 2020.

[7] G. Ghinita, M. L. Damiani, C. Silvestri, and E. Bertino, "Preventing velocity-based linkage attacks in location-aware applications," in *Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems*, 2009, pp. 246–255.

[8] X. Pan, J. Xu, and X. Meng, "Protecting location privacy against location-dependent attacks in mobile services," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 8, pp. 1506–1519, 2011.

[9] L. Zhang, Y. Qian, M. Ding, C. Ma, J. Li, and S. Shaham, "Location privacy preservation based on continuous queries for location-based services," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2019, pp. 1–6.

[10] M. J. Atallah, "Algorithms and theory of computation handbook," *Wseas Transactions on Systems*, vol. 7, no. 10, pp. 920–929, 1998.

[11] H. Jiang, P. Zhao, and C. Wang, "Roblop: Towards robust privacy preserving against location dependent attacks in continuous lbs queries," *IEEE/ACM Transactions on Networking*, vol. 26, no. 2, pp. 1018–1032, 2018.

[12] Y. Zheng, Q. Li, Y. Chen, X. Xie, and W.-Y. Ma, "Understanding mobility based on gps data," in *Proceedings of the 10th international conference on Ubiquitous computing*, 2008, pp. 312–321.

[13] Y. Zheng, L. Zhang, X. Xie, and W.-Y. Ma, "Mining interesting locations and travel sequences from gps trajectories," in *Proceedings of the 18th international conference on World wide web*, 2009, pp. 791–800.

[14] Y. Zheng, X. Xie, W.-Y. Ma *et al.*, "Geolife: A collaborative social networking service among user, location and trajectory." *IEEE Data Eng. Bull.*, vol. 33, no. 2, pp. 32–39, 2010.

[15] J. Yuan, Y. Zheng, X. Xie, and G. Sun, "Driving with knowledge from the physical world," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2011, pp. 316–324.

[16] J. Yuan, Y. Zheng, C. Zhang, W. Xie, X. Xie, G. Sun, and Y. Huang, "T-drive: driving directions based on taxi trajectories," in *Proceedings of the 18th SIGSPATIAL International conference on advances in geographic information systems*, 2010, pp. 99–108.