Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2020) 28: 3438 – 3455

doi:10.3906/elk-1909-117

A new grid partitioning technology for location privacy protection

Yue SUN[1], **Lei ZHANG**[1,*], **Jing LI**[1], **Zhen ZHANG**[2]
[1]College of Information and Electronic Technology, Jiamusi University, Jiamusi, China
[2]College of Mechanical Engineering, Jiamusi University, Jiamusi, China

Received: 22.09.2019 • **Accepted/Published Online:** 11.05.2020 • **Final Version:** 30.11.2020

Abstract: Nowadays, the location-based service (LBS) has become an essential part of convenient service in people's daily life. However, the untrusted LBS servers can store lots of information about the user, such as the user's identity, location, and destination. Then the information can be used as background knowledge and combined with the query frequency of the user to launch the inference attack to obtain user's privacy. In most of the existing schemes, the author considers the algorithm of virtual location selection from the historical location of the user. However, the LBS server can infer the user's location information on the historical data that has been counted for a long time. In order to ensure that the users' historical query data and query frequency will not be obtained by the attacker, we propose a privacy protection algorithm based on grid expansion. With the help of third-party agents, the combination of cooperative users and pseudonyms can resist the privacy disclosure caused by users requesting services during the mobile process. Extensive simulation experiments have been carried out on Gowalla dataset to evaluate the efficiency of the proposed algorithm. By comparing with other existing methods, the experimental verify the effectiveness of our algorithm in privacy protection.

Key words: Location-based service, privacy, grid expansion, third-party agent

1. Introduction

Nowadays, in the wake of development in technology for network positioning, location-based service (LBS) is indispensable service in people's modern life [1, 2]. The development of LBS brings a lot of convenience to users, such as the typical applications involve the application map, the point of interest retrieval, the GPS navigation and so on [3, 4]. However, in addition to the location data submitted by the user, the LBS server can also obtain many private information such as the user's habits, health status, location preferences and so on. Therefore, disclosing information to an untrusted third party will jeopardizes all aspects of users' private information. Therefore, the protection of users' private information is an urgent problem that needs to be solved [5].

Since the location service providers (LSPs) are not trusted, the submitted queries may lead to some sensitive information about a user being revealed, such as her real identity, exact location and queried interest [6]. Therefore, effectively solving the problem of user privacy disclosure has become an important prerequisite for people to use the LBS. Many scholars have proposed algorithms to prevent location privacy data leakage [7–9]. For example, dummy location [10], space encryption [11, 12] and location k-anonymous [13–17]. Among these algorithms, the most widely used is the k-anonymity algorithm. To make the attacker unable to determine the location of the real user, the k-anonymous hides the real user in $k-1$ users. The algorithm of random [18]

*Correspondence: 8213662@163.com

does not consider the query probability and generates the anonymous area through random selection. The grid dummy [19] is an algorithm used to obtain the anonymous region based on users' privacy requirements. The en-DLS algorithm [20] selects the dummy variables from locations that have similar query probability to the current location. These algorithms do not take into account the background knowledge of the LBS server, so they can not effectively resist the reasoning attack.

In the inference attack, the attacker can utilize the user's long-term query data as background knowledge to identify the real user. In addition, in the process of continuous query, it is easy for attackers to associate discrete locations to tracks, and reveal more information through the user's movement track, until the user's privacy. Zhang et al. [21] proposed an attribute encryption algorithm, in which enough anonymous users are found by broadcast query, and utilized the cache strategy. The algorithm can effectively resist the inference attack of LBS server, but the problem of cache update is not considered in the algorithm. On the basis of the higher calculation cost brought by the encryption process, it will also increase the storage cost. Considering the shortcomings of the existing algorithm, we proposed a grid-based genetic privacy protection algorithm (GBGPPA). The algorithm improves the anonymity of users through grid expansion, utilizes the cached information of collaborative users to reduce the information interaction between real users and the LBS server, and utilizes the pseudonym strategy to prevent the LBS server from acquiring too much historical query information. These methods cooperate with each other to invalidate the inference attack. Finally, the effectiveness of the algorithm is further verified by experiments and analysis. The contributions of this paper are highlighted as follows.

- Optimize the method of the grid division by querying the frequency, which strengthens the degree of anonymous privacy of the spatial area.
- This paper presents a caching strategy. The user can download all the query contents in the current anonymous area at the terminal, and feedback the relevant query result in the cache information to the user when the request is sent. This strategy improves the running speed and the success rate of the anonymous algorithm.
- Through the multiple pseudonym strategy to solve the situation where the user cannot find the required information in the cache. Because the attacker will not associate two pseudonyms with the same user, using different pseudonyms can reduce the attacker's recognition rate of the real user when the user sends the query continuously.

The paper is organized as follows. In Section 2, we summarized the related work. We provided some preliminaries in Section 3. We present the GBGPPA algorithms and its security analysis in Section 4. Section 5 shows the experiments results. In Section 6 we conclude and give an overview of future work.

2. Related work

2.1. Location privacy metrics

In order to find out how the attacker inferred the actual coordinates of the user, the privacy metric was needed to be determined at first. Several models for measuring the privacy level have been proposed [22–25], including the model of the privacy measure that based on uncertainty, the privacy measure based on clustering error, the privacy measure based on distortion, the privacy measures based on k-anonymity and the privacy measure based on the traceable row. In Ref. [23], the author investigated the capability of the attacker to connect two

pseudonyms to a specific user and to infer the real user. In Refs. [24, 25], the ability of the attacker is measured by the error derived from the attack model. The error refers to the distance between the actual position and the predicted position. One of the most popular private metrics models used in recent years is the k-anonymous [3, 10, 11], which hides the real user with other $k-1$ users. In Ref. [22], the author suggested that the ability of an attacker was measured by the accuracy of identifying the real user in the anonymous region. When the attacker cannot accurately determine the actual user, k-anonymity is really implemented. In Ref. [19], Sun et al pointed out that k value was positively correlated with privacy protection ability, but the measurement has some loopholes. For instance, when users are located in a low-density area (such as mountains, rivers, etc.), the attacker can quickly identify the exact location of the real user. As an addition to k-anonymity, entropy [19, 21, 22, 25, 26] can also measure the extent of privacy protection. The communication framework based on the information theory [27, 28] proposed the information entropy model for privacy protection, which refers to the probability that an attacker can identify the real user among other $k-1$ users. In general, the attacker can measure the real location by entropy, and the higher the value, the more uncertainty. Suppose there is a set of uncertain locations, which we denote this set as L. For the random location l in L, the probability of success in identifying this location is that $P(x)$, x is the number of current locations in this set. Then the uncertainty guessed by the attacker in each query can be expressed as

$$H(X) = -\sum_{i=1}^{n} p\left(x_i\right) \log_2 p\left(x_i\right), \tag{1}$$

where n is the total number of locations in L. Thus, with entropy in per query of continuous query, we can denote the average entropy in continuous query as

$$\overline{H_c} = \frac{\sum_{i=1}^{m} H_c(i)}{m}, \tag{2}$$

where n is the number of the publisher requesting the query in a continuous query. Finally, based on the average entropy, we can get the variance of the average entropy, which is expressed as

$$\sigma^2 = E\left[\left(H_c - \overline{H_c}\right)^2\right] = \frac{\sum_{i=1}^{m} \left(H_c - \overline{H_c}\right)^2}{m} \tag{3}$$

Among them, the lower the σ^2 is, the higher privacy level will be. When a user sends a query request to the LBS server, the anonymous algorithm successfully confuses the real location of the user within the maximum response time set by the user to indicate the success of anonymity. The success rate of anonymity is also an important criterion for evaluating the success of the anonymous algorithm. The success anonymity ratio can be denoted as

$$c_{rate} = \frac{c_v}{c_t} \times 100\%, \tag{4}$$

where C_{rate} is the anonymous success rate, C_v is the number of successful anonymous LBS requests, and C_t is the total number of LBS requests during the query process.

In the algorithm of constructing anonymous area, the small anonymous area can improve the quality of service of LBS, so the area of anonymous region is an important parameter to measure the location ambiguity

algorithm. The calculation method is shown in Formula (5), where $a(c(i))$ is the area of the anonymous area constructed for the user in the location privacy protection algorithm. The average value $\overline{a(c(i))}$ is usually utilized to evaluate the generation efficiency of anonymous area algorithm. The stability of the algorithm is reflected by the variance S^2 of the anonymous area.

$$\overline{a(c(i))} = \frac{\sum a(c(i))}{N_c} \tag{5}$$

$$s^2 = \frac{\sum (a(c(i)) - \overline{a(c(i))})^2}{N_c} \tag{6}$$

Now that we have developed the criteria for evaluating the algorithm, and we will show the results of the evaluation in the experimental verification.

2.2. Location privacy mechanisms

In recent years, the issue of protecting users' location privacy has been widely concerned. In addition to policy-based and cryptographic-based algorithms, space-time stealthy [4, 5, 29, 30], location confusion [7, 10, 13–17, 22] and dummy locations [8, 21, 27] are widely utilized. At present, fuzzy user's real location is the most commonly used and most effective location privacy protection algorithms. The function of location confusion is to prevent the malicious attacker from directly obtaining accurate location information. It core idea of location obfuscation is to modify the location information or service information before the user sends the query request, and then combine the request with the modified information. But the realization of this technology often depends on the side knowledge of the attacker [31]. In addition, the query information sent to the LBS server contains the anonymous area of the user's real location, which will lead to the decline of the service accuracy of LBS. Therefore, the core of the technology based on distortion is to design the best privacy protection algorithm when the attacker has obtained the side information. This algorithm cannot only satisfy the needs of users for the quality of service but also meet the requirements of users to protect privacy [32].

In Ref. [18], the anonymity of location is realized by randomly generating a dummy position. However, the virtual location can be excluded when the attacker has side information, so this algorithm cannot guarantee the user's privacy. In Ref. [19], the grid dummy algorithm is used to generate an anonymous region. The algorithm is based on the grid to generate virtual locations to achieve k-anonymity. Niu et al. [20] put forward an algorithm that selects the dummy variables from locations that have similar query probability to the current location. But this algorithm cannot resist the attack that based on long term statistics. The long-term statistical attack model refers to the behavior that the attacker obtains the historical query information by invading the LBS server and infringes the user's privacy by this. Different from the above existing work, the algorithm proposed in this paper realizes k-anonymity with the grid expansion and offers the cache and multiple pseudonyms strategy for mobile users. In addition, we verify the algorithm by experiments, and the algorithm can effectively resist the long-term statistics attack.

3. Preliminaries

3.1. System architecture

The algorithm adopts the central anonymous structure, which includes three parts: the mobile user's intelligent terminal, the central anonymous device and the LBS server. A complete request flow of LBS is summarized by

summarizing the system structure of Figure 1.

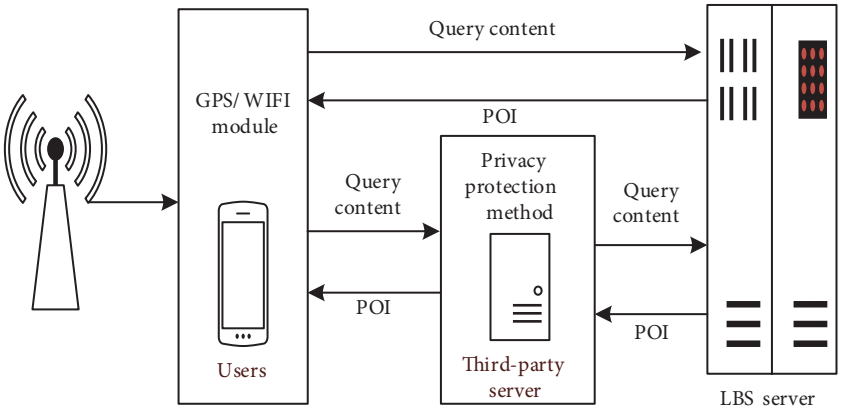

Figure 1. The system structure of LBS.

(1) The mobile intelligent terminal transmits the grid ID, privacy configuration, query request and other information to the central anonymous device.

(2) After receiving the service request of the user, the central anonymity server utilizes the privacy protection algorithm to construct the anonymous area, and then sends the coordinates of the anonymous area and the user's query request to the LBS server.

(3) The LBS server queries the stored road network data, searches the POI information set that matches the user's request, and feeds it back to the central anonymous device.

(4) The central anonymity filters the POI set according to the user's region and feeds back the final result to the user.

In order to prevent the disclosure of user privacy, the traditional privacy protection algorithms are often implemented by establishing a collaborative group. However, the anonymous areas generated in areas with low user density are large, so it is difficult to select suitable cooperative users, which makes the real users easy to be identified by attackers. Dummy location algorithm can solve these problems well. However, most of the current work usually utilize the random algorithm to select the dummy location. For example, Alice sends a query request to the LBS server. At this point, the probability that the attacker recognizes Alice is $1/k$ according to the theory of k-anonymous algorithm. However, the attacker will filter out unduly false locations according to the query probability (for example mountains and rivers) because the dummy location is randomly selected. Figure 2 shows how an attacker attacks the user by identifying the dummy users.

3.2. Attack model

In this work, the real user sends query requests in the anonymous area covering k users, while the attacker identify the actual location of the specific user by side information. Attackers mainly obtain side information from the following methods. The attacker can monitor an area to get private information sent by the user (such as name, POI, and real location, etc.). The attacker can also access the feedback information by invading the LBS server. Similarly, the attacker can obtain historical data directly from the LBS server and infer information related to user privacy. Because the LBS server contains all the data in the system, it is considered

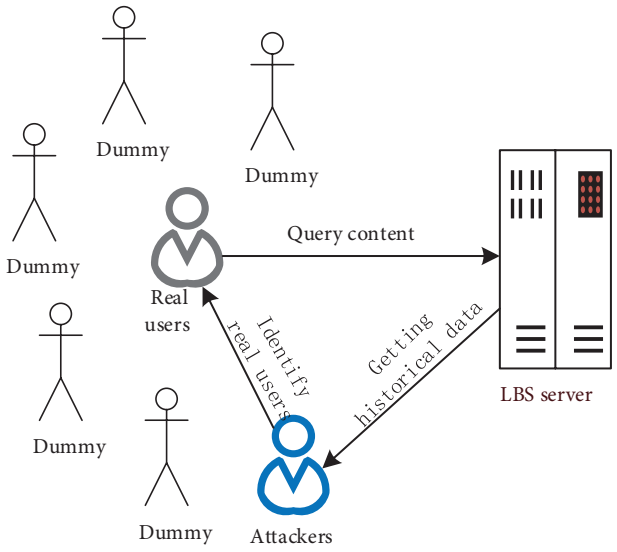

Figure 2. Method of attack by analyzing false position.

as the attacker in this paper. He will not only get the LBS query of the current user but also obtain the user's historical data.

3.3. Motivation and basic idea

The attacker can grasp side information of the specific user, and utilizes this auxiliary information to identify the particular user's location. Therefore, for the long-term statistical attack and the regional attack, this paper proposes a model for optimizing the grid expansion algorithm and suggests combining the cache strategy with the multiple pseudonym strategy. Based on this idea, we offer a GBGPPA. First of all, users determine the degree of grid division according to the privacy protection requirements, and project their own location to the corresponding grid. Utilizing the genetic algorithm to obtain the weight of each cell, and the sum of these weights is averaged to obtain a threshold. Secondly, the adjacency grid expansion algorithm is used. The algorithm starts from the first cell and adds the weight of the horizontal cell, and then the program determines whether the sum of the current weight values is equal to the threshold. The algorithm is executed recursively until it traverses all the cells in the grid. The result of the expansion is that the weight of each cell is basically the same. Thirdly, we utilize the caching strategy. Members of the collaboration group can upload the query content to the cloud, and the user can obtain the desired query results in the cloud information. When users continuously send LBS requests within a certain period of time, they can use the anonymous area formed by the last request to get feedback information. Finally, in order to cover up the relationship between the user and the changing location information, we utilize pseudonyms strategy. When the user enters the anonymous area, the mobile terminal chooses a pseudonym as the user name. Whenever the user requests LBS, she selects one of the multiple pseudonyms as the current username. Usually the attacker would not associate two pseudonyms with the same user, and the strategy reduces the recognition rate of the attacker to real users.

4. GBGPPA algorithm

In this section, we present the GBGPPA algorithm. The algorithm aims to protect the user's location privacy when the attacker masters the historical query frequency. A basic idea is to let the query frequency of the

cells in which all anonymous users are located basically equal. The steps of the privacy protection algorithm proposed in this paper are as follows:

Step 1. The user authorizes to a trusted third-party agent, which can obtain the location information of the user.

Step 2. After the agent obtains the query frequency of the current anonymous area, the user can determine the degree of grid division according to the privacy requirements.

Step 3. The weight of each cell is obtained according to the genetic algorithm. To utilizing the method of grid expansion to regrid the anonymous region. The result of repartition is that each new cell has basically the same weight.

Step 4. Create collaborative users who also utilize the third-party agent. Collaborative users can upload their own query requests and results data.

Step 5. When the user enters the current cell, the data uploaded by the cooperative user is cached at the mobile terminal. When a user applies for a service request, it first looks for the result in the cache information. If there is no query information in the cache information, the request is sent to the LBS server.

Step 6. Before sending the query request to the LBS server, the mobile terminal selects a pseudonym as the current user name.

Step 7. When the user leaves the current cell, the cache region is released. When she enters the next cell, recache the operation.

The specific process can be seen in Figure 3.

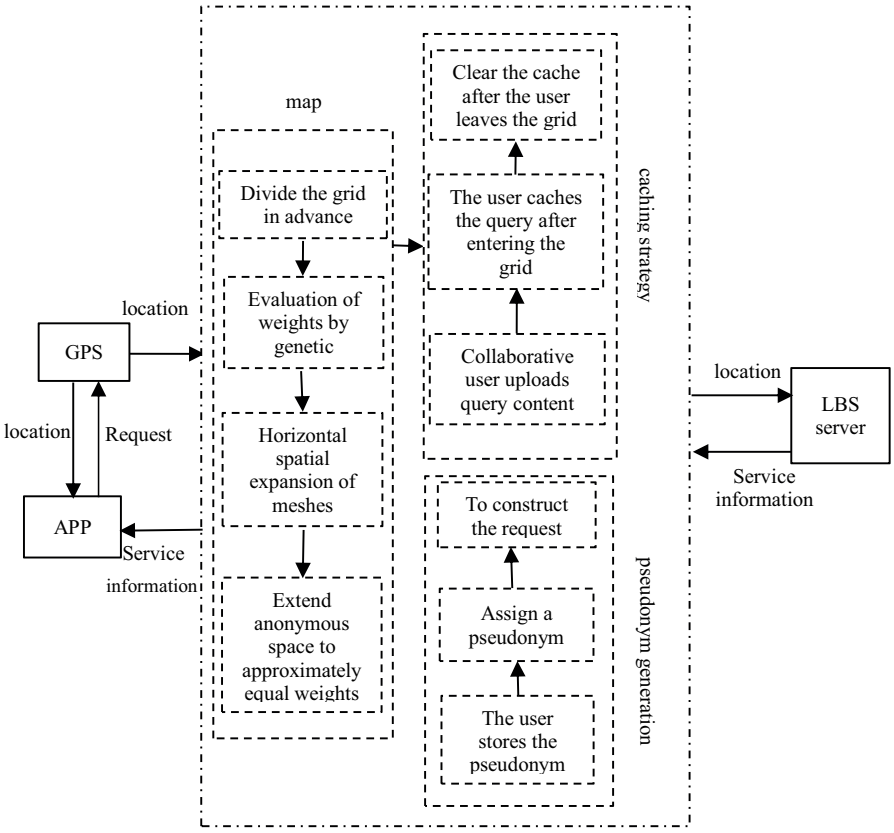

Figure 3. GBGPPA algorithm scheme diagram.

4.1. The algorithm of grid expansion

The user registers and grants permissions on a third-party application and assumes that the agent has access to historical data for the entire area. The user sets the extent to which the grid is divided according to her privacy protection requirements. Assume that the setting condition of the user is n. When the user determines the value of n, the entire area was divided into $n \times n$ grids. The agent can utilize the historical query data to calculate the query probability of each cell, and the result of the partition is shown in Figure 4. The darker grid denotes the area with higher query frequency, while the lighter grid denotes the area with lower query frequency.

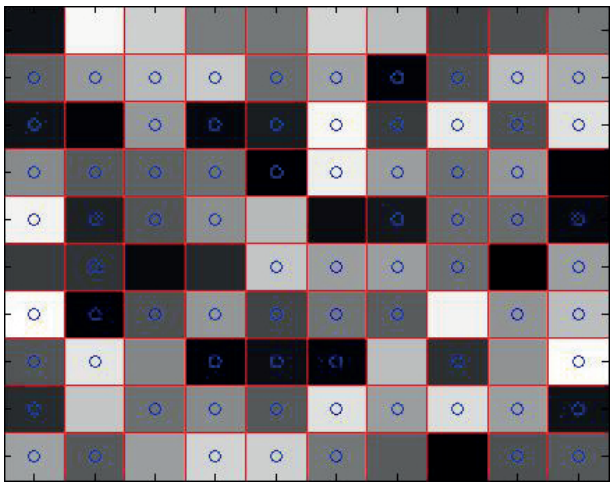

Figure 4. The result of grid division in advance.

In this paper, the specific steps of the optimized meshing algorithm proposed are as follows. Above all, according to the query frequency of each grid cell, the weight value is obtained by the genetic algorithm.

The main parameters involved in the algorithm are crossover probability P_c, mutation probability P_m, population size M and iterative times R. The main steps can be described as follows:

A. Population initialization: The initial population with the number of individuals $M = n \times n$ is randomly generated, $w_i = [w_{i1}, w_{i2}, \cdots, w_{in}]$ is a chromosome $i = 1, 2, \cdots, M$, where n is the number of variables. And the gene called chromosome $w_{i1}, w_{i2}, \cdots, w_{in}$ is randomly generated in the range of $[0, 1]$.

B. Coding: The n gene $w_{i1}, w_{i2}, \cdots, w_{in}$ of each chromosome is encoded in binary respectively, so that each chromosome has n binary strings.

C. Calculate the fitness value: The optimization goal of this paper is to find the minimum value and the value of the objective function is always positive, so adding a negative sign directly in front of the objective function can be used as the fitness function.

D. Selection, crossover, mutation: Choose the appropriate strategy and use roulette to choose, then the probability of each individual being selected is:

$$P_i = \frac{F(w_i)}{\sum_{i=1}^{M} F(w_i)} \tag{7}$$

$F(w_i)$ is the fitness value of the ith individual, then the probability of being selected is P_i. The selected individuals were randomly paired, and then the paired individuals exchanged corresponding gene bits with the crossover probability P_c. Random alteration of a gene site in a chromosome was performed with mutation probability P_m.

E. Stop the iterate: The genetic algorithm will cycle through steps C and D. Until the number of iterations is reached or the constraints are met.

The prediction matrix of the m-th component of the contrast matrix can be expressed as follows:

$$\begin{pmatrix} w_{1m}^1 & w_{2m}^1 & \cdots & w_{nm}^1 \\ w_{1m}^2 & w_{2m}^2 & \cdots & w_{nm}^2 \\ \vdots & \vdots & \ddots & \vdots \\ w_{1m}^N & w_{2m}^N & \cdots & w_{nm}^N \end{pmatrix}, m = 1, 2, \cdots, D-1 \tag{8}$$

Thus, the variable weight combination prediction value of the m component of the logarithmic ratio matrix at t time can be expressed as

$$\hat{y}_m^t = w_{1t} f_{1m}^t + w_{2t} f_{2m}^t + \cdots w_{nt} f_{nm}^t = \sum_{i=1}^{n} w_{im}^t f_{im}^t. \tag{9}$$

Considering the global optimization problem min $\{f(x)\}$, the optimal value of chromosome evolution is searched in D-dimensional space. And the optimal inertia weight of the location hidden is solved through the scientific configuration of genes, and the test sequence set of position hidden inertia weight is obtained, that is

$$\hat{y}^t = \left[\hat{y}_1^t, \hat{y}_2^t \cdots \hat{y}_{D-1}^t\right]. \tag{10}$$

The predicted value $\hat{y}_m^t$ of the logarithm ratio vector is inversely transformed into the component data, and the predicted value $\hat{x}^t = [\hat{x}_1^t, \hat{x}_2^t \ldots, \hat{x}_D^t]$ the component data is obtained. The inverse transformation formula is:

$$\begin{cases} \hat{x}_l = e^{\hat{y}_l^t} / \left(1 + \sum\limits_{m=1}^{D-1} e^{\hat{y}_m^l}\right), l = 1, 2, \cdots, D-1 \\ \hat{x}_D^t = 1 / \left(1 + \sum\limits_{m=1}^{D-1} e^{\hat{y}_m^l}\right) \end{cases} \tag{11}$$

Secondly, after determining the weight of each cell, we add these weights and calculate the average value, which we call the threshold.

$$avg = \frac{\hat{x}_1 + \hat{x}_2 + \cdots + \hat{x}_D^t}{M} \tag{12}$$

Finally, the adjacency grid expansion algorithm is utilized. The algorithm adds the weight of the horizontal cell from the first grid cell, and then the program determines whether the sum of the current weight is equal to the threshold. If the sum of the current weight is similar to the threshold, the currently expanded grid cells become a new anonymous area. Otherwise, the grid continues to expand horizontally. This algorithm is executed recursively until all cells of the grid are traversed, and the result is that the weight of each newly generated anonymous region is basically the same. Algorithm 1 describes the region expansion in detail.

Algorithm 1

Input: the weight w_{ij} of each cell a_{ij}, and the average (avg), weight accumulation and $sum_1 = sum_2 = 0$;
Output: a collection of cells that are grouped into the same anonymous region;

```
for (i = 0; i < n; i++) do
    for (j = 0; j < n; j++) do
        sum1 = a_ij + a_i+1;
        if (sum1 < avg) then
            save the current a_ij + 1 to the set n;
            sum2 = sum1+ a_ij+1;
            if |sum2 − avg| < |sum1 − avg| then
                save the a_ij+2 to the set n;
            else
                returns the a_ij + 1, saves the collection n;
            end if
        end if
    end for
end for
return n;
```

There are two typical algorithms for dividing anonymous area: quad grid algorithm and quad tree meshing algorithm. Compared with the quad tree meshing algorithm, quad grid meshing algorithm generates smaller anonymous area and higher precision. In this paper, the grid expansion algorithm is completed through the following two steps. First of all, the user can determine the extent to the corresponding grid divided according to the frequency. Secondly, the adjacency grid expansion algorithm is utilized. Through experimental verification, it can be concluded that the proposed algorithm is finer in granularity and smaller in generated anonymous regions than the quad grad meshing algorithm. Besides, this paper utilizes the caching strategy and multiple pseudonyms strategy in the subsequent optimization algorithm. For users who continuously send query requests, the caching mechanism can reduce the probability of users sent query information to the LBS server. For users who remain static for a long time, utilize different pseudonyms to prevent attackers from confirming the private information of real users and resisting long-term statistical attacks.

4.2. Cache mechanism

For users who are moving, we set up the collaboration group. When other users in the collaboration group query in the grid, they can upload their query content and query results to the cloud for backup. The user can download the backup information in the current grid area to the mobile terminal, and search for the required content in the terminal before sending the query. If there is a corresponding query result in the cached information, the user can utilize it directly. At this time, the LBS server cannot get the private information of the user. In addition, in order to prevent the cache information from taking up too much memory, we set the time interval for the cache information to be refreshed. Here we set to clear all cached content of the area in the mobile terminal when the user leaves the current area. And when the user enters the next anonymous area, the backup information in this area is automatically downloaded. Furthermore, when the user cannot obtain the required query information from the cache information, we will utilize the multiple pseudonyms strategy.

4.3. Multiple pseudonym algorithm

Due to the different layouts of topography and residential locations, the probability of LBS services applied in different regions is also different. For example, users generally do not apply for LBS services in these areas with

mountains and rivers. When the attacker has acquired the side information, the specific user can be determined according to the long-term statistical attack model. Therefore, the traditional algorithm for selecting k-1 dummy locations can no longer meet the current location privacy protection requirements. In this case, we utilize the algorithm of multiple pseudonyms to deal with this attack. The user stores multiple pseudonyms in the mobile terminal. Whenever a user requests LBS, she chooses one of many pseudonyms as the current user name and sends it to the LBS server together with the query information. We assume that user visits to LBS are sporadic. For the attacker, it means that there is a time interval between two consecutive query requests. As a result, it is difficult for an attacker to determine that query requests with two different user names are sent by one user. Therefore, this multiple pseudonyms strategy combined with the above method can effectively resist the long-term statistical attacks of the LBS server, so as to protect the privacy of users.

4.4. Performance analysis

In the following, we will analyze the performance of the proposed algorithm. By comparing with other algorithms, it is pointed out that the algorithm in this paper is more practical.

4.4.1. Utility

In this algorithm, the query frequency of each cell is basically the same by the way of grid expansion, which resists the reasoning attack. Through the caching strategy, the interaction between the user and the LBS server is reduced, thus the user's privacy information is effectively protected. In addition, the multiple pseudonyms strategy is adopted when the cache is not available, which only changes the identity of the user in the query process, and will not affect the accuracy of the feedback received by the user. In short, the utility of the proposed GBGPPA algorithm is reasonable.

4.4.2. Communication cost

The communication cost of the user is mainly composed of two parts. On the one hand, the overhead caused by the user building collaborative groups with other users in the space and collecting query data of collaborative users. If users keep moving, establishing different collaboration groups in different grids will increase the communication cost. The second is the communication cost of submitting query requests to untrusted LBS servers. The multiple pseudonyms strategy proposed in this paper is to store multiple pseudonyms in the mobile terminal, but use different pseudonym when sending the query, so it will not bring additional communication overhead. And for each message, the maximum is no more than 64 bytes, so the communication cost is $O(k)$. In addition, the caching strategy adopted by the user can reduce the interaction with the LBS server, and reduce the communication overhead to a certain extent.

4.4.3. Computational cost

Genetic algorithm belongs to double iteration, so the computational cost will not exceed $O(n^2)$. The region expansion algorithm includes two layers of for-loops and one layer of if-loops. Therefore, in the worst case, the algorithm takes $(n^2 + n + n^3)$ and the time complexity is $O(n^3)$. In the multiple pseudonym strategy, the user randomly selects a user name stored in the mobile terminal, in which the computational cost is very small. Therefore, this computational cost is ignored in the performance analysis. For the cache strategy, when evaluating the processing time of the cache strategy, we ignore the transmission delay between mobile users and only consider the processing time of the algorithm itself. Throughout the process, if there are enough users

in the cell, the number of cached content will increase. In this case, the user can resolve her LBS request by searching for service data in the cached data rather than through LSP. As a result, the average computational cost of LBS queries at each location decreases.

4.4.4. Storage cost

Multiple user names are stored in the mobile terminal, so the storage cost is related to the number of the stored user name, which is the $O(n)$, where n is the number of the stored user name. For cached content, it will occupy the certain storage space of the user. But because the user will empty the cached information in the current cell when entering the next cell, it can control the user's storage overhead to a certain extent.

4.4.5. Security analysis

The important basis for judging whether the privacy protection algorithm can resist reasoning attacks is as follows. After the grid expansion algorithm, the real users have the same chance to be in each grid cell. Assume that the chance of a real user in two arbitrary locations C_i and C_j in an anonymous area is P_i and P_j. Through the steps of the algorithm in this paper, we can make the query probability of each area basically the same, so $P_i = P_j$. However, when the query probability distribution of anonymous region is hugely uneven, the algorithm in this paper may be difficult to find a suitable anonymous region. In other words, it is possible for an attacker to extrapolate the location of the real user in this situation.

Additionally, this paper also adopts a caching strategy to solve the problem of location privacy leakage caused by users continuously updating their locations. The user can download the query information of the current grid to mobile terminals to reduce the frequency of sending LBS requests. In addition, in order to prevent the cache information from taking up too much memory, we set the time interval for the cache information to be refreshed. For the attack model based on long-term statistics, we adopt the algorithm of multiple pseudonyms. The algorithm means that the user utilizes a different username each time the query is sent. Therefore, even if the attacker has historical query data for this area, it is impossible to infer where the actual user located. For an attacker, it is difficult to contemplate that a single user will use two different user names to send two requests.

5. Experiments and results

This section carries on the functional analysis through the experiment to the GBGPPA algorithm. The experiments focus on three aspects, i.e. privacy protection ability of the algorithm, the communication cost of the user and the relevance of the content in the continuous query.

5.1. Experimental setup

The environment we experimental was a desktop with Intel Pentium Nvidia GeForce GTX 1050 Ti and 8.0 GB RAM. The experiment was carried out in the Win 10 64-bit computer and Matlab 2016b software. The experiment was carried out on the Gowalla dataset. The global access set in the Gowalla dataset is the check-in location owned by the user, including 196591 users, 1280969 POI, and 6442890 check-in data in the dataset. The spatial region selection experiment was conducted in the 40×40 km area of the United States.

We contrast the algorithm in this paper with three other anonymous protection algorithms, namely are the random, the grid dummy and the en-DLS algorithm. The random algorithm [18] is an algorithm for generating anonymous areas by randomly selecting without considering the query probability. The grid dummy [19] is an

algorithm for obtaining the anonymous area under the condition of satisfying the user's privacy requirements. The en-DLS algorithm [20] selects the dummy variables from locations that have similar query probability to the current location. SACU [21] is an attribute encryption algorithm that requests the establishment of a cooperative group by broadcast and combines the cache strategy. The performances of the five algorithms in entropy measurement, area generation of anonymity, communication cost are compared by experimental simulation.

5.2. Experimental simulation and analysis

Figure 5 shows the comparison results of the five algorithms at different k values. The experiment assessed the change in the level of privacy protection. When the probability of the attacker identifying k locations as real users is the same, the algorithm reaches the maximum entropy. Therefore, the scheme with the maximum entropy is optimal. When the attacker has mastered the query frequency, he can filter extremely dummy locations. Thus, the algorithm of Random selecting a dunmmy location is the least satisfactory among several algorithms. Because the random algorithm generates anonymous areas without considering other factors, the attacker can easily identify the real user. Grid dummy confirms grid points by selecting dummy positions, so the vertices of the grid are fixed. In this case, the entropy of the algorithm is decided by the query frequency of the anonymous region. Therefore, the grid dummy algorithm is similar to the random algorithm in performance. Because en-DLS is an algorithm to select dummy locations based on the user's historic location, it can achieve a higher level of privacy.The GBGPPA algorithm uses an improved grid expansion method to divide the space into anonymous regions with the same query frequency. Secondly, the caching mechanism is used to reduce the interaction between the user and the LBS server. At the same time, the strategy of pseudonym is used, which makes it impossible for the attacker to determine that it is the same user in sporadic queries. The SACU algorithm is a kind of attribute encryption algorithm which combines cache strategy. So we can see that both the GBGPPA algorithm and the SACU algorithm reach the maximum entropy, which means the adversary has the maximum uncertainty in determining which location belongs to the issuer.

Figure 6 shows the variance of the average entropy in a continuous query. From the figure, we can see that there is not much difference in the variance between these algorithms, because the average entropy is close to the entropy of each query. We can see that the GBGPPA algorithm works best because of the use of multiple pseudonym strategies. It is difficult for the attacker to confirm that the user will send two requests from different user names. Even if the attacker utilizes the attack model of long-term statistics, it is not easy to confirm the private information of real users. Therefore, the average entropy of GBGPPA algorithm is smaller than the other three algorithms. When a certain amount of data is reached, the average entropy will not be changed. Besides, this algorithm can obtain the maximum entropy in each snapshot query, the difference between current entropy and average entropy can be ignored. So the square difference is almost zero. SACU is an algorithm that provides anonymity of attributes to the user. When a suitable cooperative user is not found in the space, the average variance will increase. In the en-DLS algorithm, an attacker can guess the user's entire movement trajectory based on the association of attributes. Grid dummy algorithm cannot provide anonymity of attributes to users, so the privacy of users will be exposed in the process of sending queries continuously. The random algorithm randomly generates hidden regions, which makes the difference between the average entropy and the snapshot query even greater.

The size of anonymous area is closely related to location privacy. Therefore, the size of the anonymous area generated by the five algorithms is evaluated by the change of anonymity. The comparison result is shown

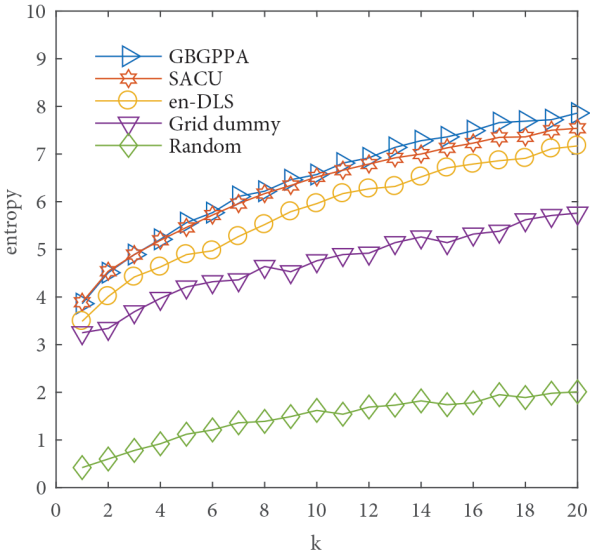

Figure 5. The value of entrypt in snapshot query.

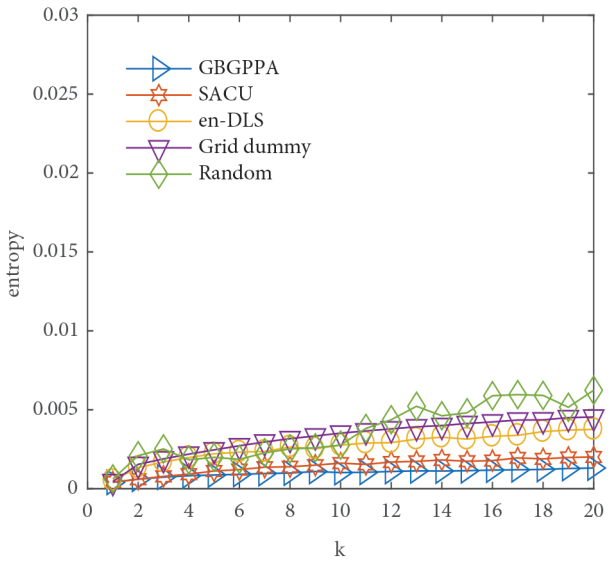

Figure 6. The variance of average entropy in continuous query.

in Figure 7. Because the random algorithm is an anonymous area generated by randomly selected locations, the anonymous area generated with the increase of k-anonymity is irregular. It turned out that the changes in the other four algorithms was relatively stable, and the anonymous area is increased with the increase of k. Grid dummy algorithm sets the anonymous space according to the needs of users, so the anonymous space does not change much. Because the choice of dummy location in en-DLS algorithm is based on the user's historical location, the generated anonymous area is the smallest as a whole. When the SACU algorithm publishes broadcasts to establish collaborative user groups, if the appropriate anonymous users can not be found quickly, the anonymous area will be increased. Although when the requirement of anonymity is low, the anonymous area generated by GBGPPA algorithm is larger than that of en-DLS algorithm. However, it can be seen that the area of anonymous area generated by GBGPPA algorithm does not increase much with the increase of anonymity. To sum up, the en-DLS and GBGPPA algorithms are relatively better when the anonymous region changes with the increase of anonymity.

Through the change of the number of user requests for service, we use five algorithms to analyze the communication cost. The comparison result is shown in Figure 8. The comparison results show that except for the grid dummy algorithm, the communication cost of the other four algorithms increases with the increase of the number of queries. This is because several other algorithms generate dummy variables on the client side, so the communication cost increases with the increase of k. In the grid dummy algorithm, the user only needs to send the real location to the third-party server, so the communication cost is a constant. The SACU algorithm will bring high communication overhead when sending broadcasts to establish cooperative users and attribute encryption, so the communication cost is high. The en-DLS algorithm selects a dummy location on the user's historical location, so it is similar to the multiple pseudonym method in communication cost. Because the GBGPPA algorithm utilizes the caching strategy, it will bring high communication cost to reestablish the collaboration group in the process of continuous movement. However, because most users can get the query results in the cache content, it can balance the communication cost caused by reestablishing the collaboration group, so the communication cost of GBGPPA algorithm is acceptable.

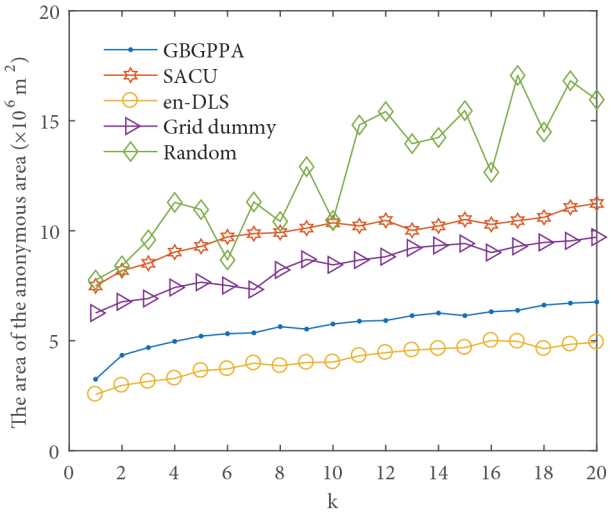

Figure 7. The relationship between anonymous areas and the degree of privacy.

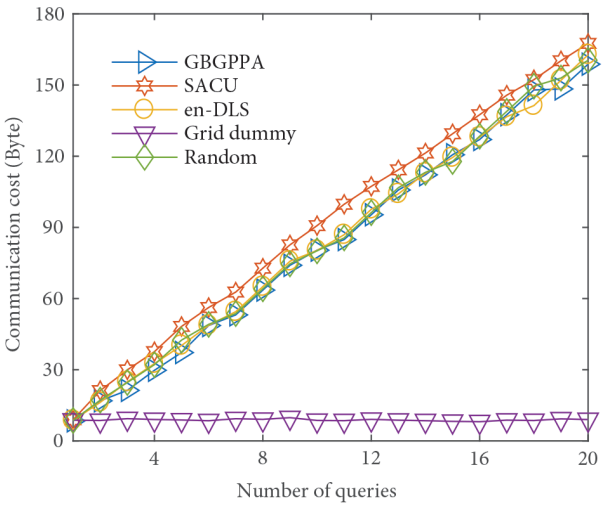

Figure 8. Communication cost of users.

Figure 9 shows the successful anonymity rate for each scenario in a single snapshot query. As the value of k increases, there may not be enough anonymous users to be found, so the success rate of anonymity decreases with the increase of k. The successful anonymity rate of Random algorithm is the lowest. This is because when a user sends a query request, the algorithm randomly generates a anonymous area, and the area may contain many false locations. The grid dummy algorithm selects the anonymous area according to the needs of the user, but when there are not enough users in the space, it will reduce the success rate of anonymity. The SACU algorithm utilizes queries to broadcast to a larger area to find anonymous users. However, the process of finding and encrypting attributes when establishing collaborative users is very time-consuming. When multiple users request services at the same time, the waiting time is too long. And the success rate of anonymity will be reduced when cooperative users cannot be trusted. The performance of the en-DLS algorithm is also satisfactory because it generates dummies in historical locations and utilizes dummy locations to achieve trajectory anonymity. Because the result of grid expansion algorithm is that the query frequency of each anonymous region is basically the same, so the performance of GBGPPA algorithm is the best. The GBGPPA algorithm can hide the location of the real user, so it is difficult for the attacker to find the real user according to the query frequency. In addition, the strategy of combining caching and multiple pseudonyms allows users to obtain higher privacy protection when sending query requests.

The anonymity of continuous queries is affected by the correlation between consecutive query requests. The correlation between consecutive query requests refers to the attacker can infer the location information of users according to the continuous query requests sent by users. The lower the degree of correlation, the higher the success rate of execution. Figure 10 shows the successful anonymity rate in consecutive queries. Similar to the results of snapshot queries, the success rate of continuous queries decreases with the increase of the number of consecutive queries. At least two snapshot query requests are included in the continuous query process. If the privacy protection algorithm fails to protect one of the queries, it means that the attacker may launch inference attacks according to the discrete locations where these protection failures are failed. Therefore, the anonymous success rate of continuous queries is generally lower than that of snapshot queries. We can also see from the figure that the GBGPPA algorithm proposed in this paper has the highest anonymous success rate. This is

because the algorithm utilizes the caching strategy, which enables users to find the required information in the cache content, thus reducing the number of query requests sent to the LBS server. Secondly, when the user is unable to get the required information in the cache, the algorithm uses a pseudonym strategy to make the user interact with LBS with different pseudonyms. Because the Random algorithm utilizes the randomly selected anonymous area to interfere with the trajectory generated by the region, the correlation is also low. However, the randomly selected anonymous area may expose personal sensitive information during the movement, so the success rate is not high. In addition, when there are untrusted nodes between cooperative users, the SACU algorithm will disclose the user's query information, which will affect the success rate. The other two algorithms do not take into account the anonymity of continuous queries, so the degree of association is higher than that of GBGPPA algorithm.

In conclusion, we can see our proposed algorithm GBGPPA has a better performance in privacy protection and the algorithm execution efficiency. In addition, the algorithm can also prevent long-term statistical attack models. Therefore, we consider that our algorithm will have a broader application prospect.

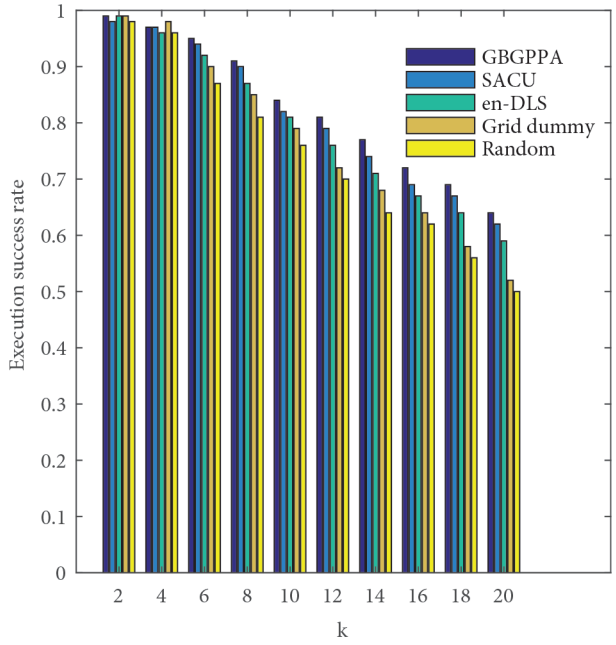

Figure 9. The anonymity rate of sporadic snapshot queries.

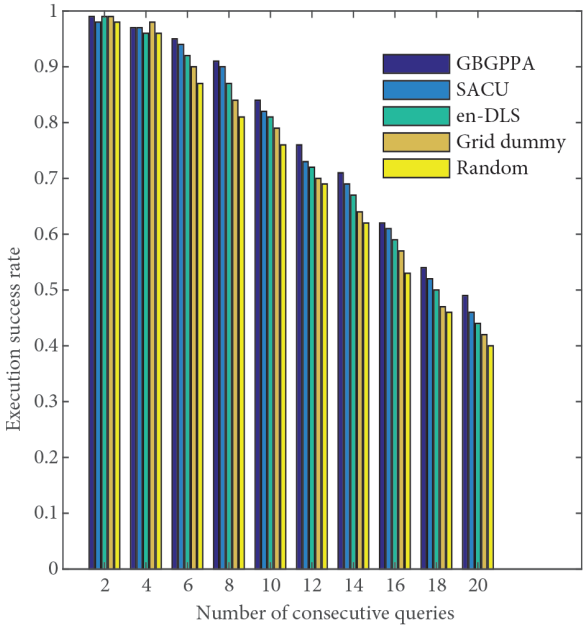

Figure 10. The success anonymity rate in continuous query.

6. Conclusion

In order to prevent the attacker from infringing upon the user's location privacy by the side information, this paper proposes a grid-based genetic privacy protection algorithm (GBGPPA). First of all, in order to enhance the privacy of anonymous areas, the grid expansion algorithm based on query frequency is optimized. According to the query frequency, the problem that the attacker invades the user's privacy is solved. Secondly, improve the efficiency and success rate of anonymous algorithms through the caching strategy. This strategy reduces the risk of location privacy leakage caused by users constantly updating their location. Finally, multiple pseudonyms are intended to dispose of the problem of location privacy leakage caused by user stillness. This strategy reduces the

probability of attackers identifying real users. In addition, we utilize a variety of means to analyze the privacy protection capacity of this algorithm. Experiments show that the algorithm can protect users' privacy, avoid attacks based on long-term statistics, and have good performance. However, when the distribution of query probability in the anonymous area is hugely uneven, the GBGPPA algorithm may fail to find the grid cells with the maximum threshold, thus failing to protect user privacy. This problem will be focused as the future work.

Acknowledgments

This work was supported by the Natural Science Fund of Heilongjiang Province for Outstanding Youth (YQ2019F018), Post Doctoral Fund Project in China (2019M661260), Excellent Discipline Team Project of Jiamusi University (JDXKTD-2019008), Special Doctor Scientific Research Fund Launch Project of Jiamusi University (JMSUZB2018-01).

References

[1] Gupta BB, Akhtar T. A survey on smart power grid: frameworks, tools, security issues, and solutions. Annals of Telecommunications 2017; 72 (9-10): 517-549.

[2] Stergiou C, Psannis KE, Kim BG, Gupta BB. Secure integration of IoT and cloud computing. Future Generation Computer Systems 2018; 78: 964-975.

[3] Zhang L, Li J, Yang S, Wang B. Privacy preserving in cloud environment for obstructed shortest path query. Wireless Personal Communications 2017; 96 (2): 2305-2322.

[4] Gupta BB, Gupta S, Chaudhary P. Enhancing the browser-side context-aware sanitization of suspicious HTML5 code for halting the DOM-based XSS vulnerabilities in cloud. International Journal of Cloud Applications and Computing 2017; 7 (1): 1-31.

[5] Niu B, Li Q, Zhu X, Cao G,Liet H. Enhancing privacy through caching in location-based services. In: 2015 IEEE Conference on Computer Communications; Hong Kong, China; 2015. pp. 1017-1025.

[6] Akhtar T, Gupta BB, Yamaguchi S. Malware propagation effects on SCADA system and smart power grid. In: 2018 IEEE International Conference on Consumer Electronics; New York, NY, USA; 2018. pp. 1-6.

[7] Schlegel R, Chow C, Huang Q, Wong DS. User-defined privacy grid system for continuous location-based services. IEEE Transactions on Mobile Computing 2015; 14 (10): 2158-2172.

[8] Ye A, Chen Q, Xu L, Wu W. The flexible and privacy-preserving proximity detection in mobile social network. Future Generation Computer Systems 2018; 79 (1): 271-283.

[9] Peng T, Liu Q, Meng D, Wang G. Collaborative trajectory privacy preserving scheme in location-based services. Information Sciences 2017; 387 (5): 165-179.

[10] Wang S, Hu Q, Sun Y, Huang J. Privacy preservation in location-based services. IEEE Communications Magazine 2018; 56 (3): 134-140.

[11] Li X, Miao M, Liu H, Ma J, Li K. An incentive mechanism for k-anonymity in LBS privacy protection based on credit mechanism. Soft Computing 2017; 21 (14): 3907-3917.

[12] Rohilla A, Khurana M, Singh L. Location privacy using homomorphic encryption over cloud. International Journal of Computer Network and Information Security 2017; 9(8): 32-40.

[13] Sun G, Liao D, Li H, Yu H, Chang V. L2P2: a location-label based approach for privacy preserving in LBS. Future Generation Computer Systems 2017; 74 (9): 375-384.

[14] Yin C, Xi J, Sun R, Wang J. Location privacy protection based on differential privacy strategy for big data in industrial internet of things. IEEE Transactions on Industrial Informatics 2018; 14 (8): 3628-3636.

[15] Dargahi T, Ambrosin M, Conti M, Asokan N. ABAKA: a novel attribute-based k-anonymous collaborative solution for LBSs. Computer Communications 2016; 85 (7): 1-13.

[16] Ni W, Gu M, Chen X. Location privacy-preserving k nearest neighbor query under user's preference. Knowledge-Based Systems 2016; 103 (7): 19-27.

[17] Huang Y, Cai Z, Bourgeois AG. Search locations safely and accurately: a location privacy protection algorithm with accurate service. Journal of Network and Computer Applications 2018; 103 (1): 146-156.

[18] Kido H, Yanagisawa Y, Satoh T. An anonymous communication technique using dummies for location-based services. In: Pervasive Services, 2005 ICPS'05 Proceedings International Conference; Santorini, Greece; 2005. pp. 88-97.

[19] Sun Y, Chen M, Hu L, Qian Y, Hassan MM. ASA: against statistical attacks for privacy-aware users in location based service. Future Generation Computer Systems 2017; 70 (5): 48-58.

[20] Niu B, Li Q, Zhu X, Cao G, Li H. Achieving k-anonymity in privacy-aware location-based services. In: IEEE INFOCOM 2014-IEEE Conference on Computer Communications; Westin Harbour Castle Toronto, Canada; 2014. pp. 754-762.

[21] Zhang L, Li J, Yang S, Wang B, Bian X. A novel attributes anonymity scheme in continuous query. Wireless Personal Communications 2018; 101 (2): 943-961.

[22] Chen J, He K, Yuan Q, Chen M, Du R et al. Blind filtering at third parties: an efficient privacy-preserving framework for location-based services. IEEE Transactions on Mobile Computing 2018; 17 (11): 2524-2535.

[23] Niu B, Zhu X, Li Q, Chen J, Li H. A novel attack to spatial cloaking schemes in location-based services. Future Generation Computer Systems 2015; 49 (8): 125-132.

[24] Lu R, Lin X, Shen X. SPOC: a secure and privacy-preserving opportunistic computing framework for mobile-healthcare emergency. IEEE Transactions on Parallel and Distributed Systems 2013; 24 (3): 614-624.

[25] Meyerowitz J, Roy Choudhury R, Romit. Hiding stars with fireworks: location privacy through camouflage. Proceedings of the Annual International Conference on Mobile Computing and Networking 2009; 1: 345-356.

[26] Shokri R, Theodorakopoulos G, Le Boudec JY, Hubaux JP. Quantifying location privacy. In: 2011 IEEE Symposium on Security and Privacy; Berkeley, CA, USA; 2011. pp. 247-262.

[27] Zhu X, Chi H, Niu B, Zhang W, Li Z et al. Mobicache: when k-anonymity meets cache. In: 2013 IEEE Global Communications Conference, (Globecom); Atlanta, GA, USA; 2013. pp. 9-13.

[28] Li Z, Wang J, Zhang W. Revisiting post-quantum hash proof systems over lattices for internet of thing authentications. Journal of Ambient Intelligence and Humanized Computing 2019; 1: 1-11.

[29] Palanisamy B, Liu L. Effective mix-zone anonymization techniques for mobile travelers. Geoinformatica 2014; 18 (1): 135-164.

[30] Lei Z, Lili H, Desheng L, Jing L, Jiang QF et al. An attribute generalization mix-zone without privacy leakage. IEEE Access 2019; 7: 57088-57099.

[31] Shokri R, Theodorakopoulos G, Troncoso C, Hubaux JP, Le Boudec JY et al. Protecting location privacy: optimal strategy against localization attacks. In: Proceedings of the 2012 ACM Conference on Computer and Communications Security; New York, NY, USA; 2012. pp. 617-627.

[32] Lei Z, Chunguang M, Songtao Y, Xiaodong Z. Probability indistinguishable: a query and location correlation attack resistance scheme. Wireless Personal Communications 2017; 97 (4): 6167-6187.