Abstract
Advances in wireless communication and ubiquitous mobile networks have resulted in location-based services (LBSs) contributing significantly to providing real-time services. These services enable mobile users to enjoy entertainment services by utilizing their location data. In most existing approaches, the anonymous server immediately generates the cloaking region based on the current location information when receiving requests for LBSs from users to ensure location privacy protection. However, for requests with a limited time frame, such as “where is the gas station nearest to me?” the anonymous server must refine all the candidate results and return the optimal result to me within 5 minutes,” the cloaking region is formed by the anonymous server upon the $k$-anonymity requirement privacy level achieved in the methods mentioned above. Unfortunately, movement by the mobile user could cause the cloaking region to become slightly smaller. In a situation such as this, the existing algorithm would be unable to return the optimal cloaking region, and the result returned to the user is certain to be sub-optimal. This problem prompted us to develop a novel probability-based prediction query (PBPQ) algorithm, which is proposed in this paper. On the server side, our method firstly predicts the probability of the mobile user being on each road, following which the method calculates the probability of the user entering the cloaking region according to the normal distribution characteristics of the speed difference before it computes the final probability of the mobile user being in the cloaking region. The method of probability prediction enables our proposed algorithm to judge whether the cloaking region generated by the anonymous server at the current time would be optimal. If not, PBPQ continues to execute until the optimal cloaking region occurs within the limited time required by the user. Our proposed algorithm can achieve a remarkable improvement in the quality of service (QoS) of the request for LBSs from the mobile user within a limited time frame. In addition, the target user does not need to upload his location information to the anonymous server. As a result, communication overhead is reduced significantly. The experimental results show that our PBPQ algorithm can predict the optimal cloaking region effectively at the highly successful rate of 90% by testing the impact of prediction time, privacy level, and probability threshold. Thus, the PBPQ algorithm can decrease the size of the cloaking region considerably while greatly improving the QoS.

Keywords:
location-based services, PBPQ, optimal cloaking region, QoS

1. Introduction
Owing to the recent development in wireless communication computing and ubiquitous mobile networks, location-based services (LBSs) are experiencing tremendous breakthroughs that enable users to enjoy entertainment services by utilizing location data. These services, which have become widely noticed in recent years, are highly convenient for users with location-aware mobile devices, such as cell phones, which allow LBSs to be accessed everywhere and at any time.

In general, the key issue is finding the $k$-nearest neighbors ($k$-NNs). For example, a user uses an LBS to send query requests to the anonymous server to identify all hotels within a certain distance from their current location. However, the use of LBSs requires users’ current location information to be uploaded to the server. This process is likely to reveal mobile users’ personal information by means of their location information, finally resulting in privacy disclosure. Moreover, the leakage of location information to the server has a negative effect on the widespread application of LBSs. Thus, many researchers have aimed to address the issue of location privacy protection to avoid re-identification when mobile users send requests for LBSs [1, 2, 3, 4, 5]. A typical situation would be the following: Bob is driving along a road, but he is forced to locate the nearest gas station within 5 minutes’ drive by utilizing his smart phone, in which case concealment of his accurate position would be a fundamental concern at the same time.

The most simple and convenient approach under consideration is to directly apply a pseudonym instead of the user’s identifier (ID) when the user sends the query request to the service provider. Nevertheless, the user can be re-identified due
to the location information involved in the query request, which is utilized as a quasi-identifier [6, 7, 8, 9]. If adversaries were to track the user by linking the query request to some historical behavior, the user could easily be recognized. If this attempt to reveal the user’s identity were to succeed, the adversary would have the authority to obtain their required location information and privacy data.

The three main existing fundamental methods that are used to overcome location privacy disclosure [10] are dummy [4], spatial and spatio-temporal cloaking [2, 3]. These methods are based on the principle that a user’s true position is either substituted by a fake location or that a fake location is mixed with the true one. The DUMMY – Q proposed by Pingley et al. [11] involves consistently coordinating real query requests with fake ones to prevent the real request from being identified. Blurring the identification of a moving object is the basic principle underlying spatial cloaking. The precise location of the user, rather than their actual position, is an anonymous area used to disorient adversaries and protect location-sensitive information. Spatio-temporal cloaking, which is based on spatial cloaking, involves postponing the timing of the user’s query reply from the server while the temporal dimension is disguised with the purpose of location privacy protection. The third of the three methods is the most appealing approach toward location privacy protection. Gruteser and Grunwald [2] discussed the k-anonymity model to obfuscate the real position of the current user issuing a request, where there exists a region including k − 1 other users and the current user instead of the real position of the user by the use of spatial cloaking. However, this approach is likely to only meet the requirement of the current user making a query request. A personalized k-anonymity model proposed by Bu and Liu [1, 12], which has been accepted for publication, is able to satisfy the demand for privacy of every user within the same cloaked area, who may submit queries with the same sensitivity. Liu et al. [13] presented the l-diversity model to avoid disclosing the privacy of the query.

Location k-anonymity is a traditional technique to provide both location and query privacy simultaneously [1, 2, 14, 15, 16, 17, 18]. However, the model is ineffective to resist location-based attacks [19, 20] when users continually update their position information to the anonymous server. The privacy protection program for continuous queries was initially designed by Chow et al. [15], and states that the anonymous group bound to achieve at the initial time is valuable during the entire phase, which can resist attacks relating to query tracking. Nevertheless, it may be subjected to a maximum movement boundary attack. On the one hand, if users in the group are far away from each other, it will be difficult for the refiner to obtain the precise position because the cloaked region will be much too large. On the other hand, the cloaked region may even have reached such a small size that it can be represented by a point, indicating that the user’s location information is disclosed. Combining the location k-anonymity model with cloaking granularity, a new type of algorithm named ICliqueCloak was proposed and is based on the incremental clique presented by Pan et al. [21] to combat attacks related to the location. This approach requires the cloaked region to be smaller than the maximum clique. The main idea is to incrementally maintain the maximal cliques needed for location cloaking in an undirected graph that takes into consideration the effect of continuous location updates.

Figure 1. Example of a traditional query. Bob submits frequent queries at the time $t_1$. $C_2$ can satisfy his privacy level ($k=3$) at the time $t_2$ and the server immediately updates it as the cloaking region regardless of smaller cloaking regions, such as $C_4$, emerging at a later stage.

Reducing the location resolution is an effective way to depersonalize location information. However, to the best of our knowledge, the size of the cloaked region depends on the location of the neighbors around the target user who is requesting LBSs. Therefore, these methods have many limitations. On the server side, excessive reporting of location information would cause significant communication degradation and bottlenecks. Sometimes the user may not enjoy the best service because he/she cannot judge whether the current result they received is the best. Most existing algorithms generally submit multiple requests, calculate the real-time cloaking region at regular intervals and return the result immediately as long as it meets the user’s requirements. However, due to user movement, a smaller cloaking region may exist after the request. Continuing with the above example, as shown in Fig. 1, the area framed by the red dotted line is $A_{max}$, i.e., the cloaked region should be smaller than $A_{max}$ based on Bob’s k-anonymity requirement ($k = 3$). Intuitively, the cloaking region $C_2$ meets his privacy level and the server uploads this area as the anonymous area to the LBSs provider at the time $t_2$ in
Fig. 1. In fact, $C_4$ is the minimum cloaking region when he arrives at the time $t_4$. Therefore, how and when to choose the anonymous region has a great impact on the quality of service (QoS). Moreover, the frequent submission of queries may form a trajectory that may reveal the real identity of the user, a consequence of the maximum movement boundary attack, which is well discussed in [22]. Lately, little consideration has been paid to the spatial cloaking algorithm that only requests a query once but succeeds in obtaining the minimum cloaked region within the required limited time frame.

In the light of the above, we designed a novel query privacy preserving algorithm, named the probability-based predictable query (PBPQ) algorithm, from a new perspective. This algorithm obviates the need for frequent queries by mobile users to the server. Our proposed algorithm can return the minimum cloaking region within the required limited time frame. It can achieve two goals: (1) allowing mobile users on the road network to use private location-based services, (2) issuing only one query to the anonymous server at the initial time to realize the user’s anonymity. This algorithm is based on the idea of generating the smallest cloaked region at an optimal time on a road network by predicting the location probability of the mobile user depending on the user’s moving status and trend, i.e., their velocity and direction.

Realization of our goals requires the implementation of three main procedures. First, two probability matrices are defined. These matrices estimate the probability of the user’s possible location based on a network-based modeling method to attempt to predict the user’s movement on the road network, an approach introduced by Liu et al. in [23]. In addition, we can predict a user’s potential location, which is defined as their possible location, by utilizing the above matrices. Finally, we discuss an algorithm to predict and obtain the optimal time in which to form the minimum cloaking region for improved QoS via prediction of the user’s location.

A novel probability-based prediction query algorithm named PBPQ is proposed in this paper. The contributions of the algorithm are as follows:

- Predicting the probability of the user’s possible location and calculating the minimum cloaking region containing $k$ users within the required limited time frame while providing improved QoS. In general, a small result superset is preferred to reduce both the cost of data transmission and the workload of the process to refine the result, especially if this process is implemented on the mobile client [24].
- Providing high accuracy prediction results. Through our PBPQ algorithm, we can find the anonymity success rate can reach 90%. In this way, we observably improve the QoS.

The remainder of this paper is organized as follows. Section 2 introduces the related work on privacy protection in LBSs. Our proposed PBPQ algorithm is discussed in Section 3. Section 4 describes the investigation of the performance of PBPQ and the experimental results. Finally, Section 5 contains a brief conclusion.

2. Related Work

The underlying system architecture is formally defined in detail in Section 2.1. The system architecture is mainly composed of three parts: an authentication anonymous proxy, mobile users, and an untrusted service provider. The road network model and privacy model are described in sections 2.2 and 2.3, respectively.

2.1. System Architecture

The system architecture described in this paper consists of three types of components: an authentication anonymous proxy (AAP), the mobile users (MUs), and an untrusted service provider (uSP), similar to those in [24, 21, 22, 25, 26, 27]. We illustrate the system architecture in Fig. 2. A user sends a location-based query request in the form of $Q(ID, l, q, P_{model})$ to the AAP using a secure communication channel, where $ID$ is the user’s authentic identity, $l$ is the user’s location information consisting of a pair of geographic coordinates, $q$ represents the requested content, and $P_{model}$ denotes the three parameters for protecting the location privacy, i.e., $k$, $A_{max}$, $T$, and $v$ described below.

The AAP is composed of a cloaked engine and a result refiner. The cloaked engine finds the optimal time to generate a cloaked region $R$ in accordance with the user’s privacy requirement when receiving a location-based query request. Once the anonymity succeeds, the cloaked region is stored as a cloaked query request in the form of $Q'(ID, ID', P_{model}, R_{t_i}, t_i)$, where $ID'$ is the pseudonym of $ID$. $R_{t_i}$ is the user’s cloaked region at the time $t_i$. After that, the anonymous proxy transmits the modified query request in the form of $Q''(ID', q, R_{t_i}, t_i)$ to the uSP.

The uSP immediately executes the query and receives a sequence of candidate results upon receiving query request $Q''(ID', q, R_{t_i}, t_i)$. Subsequently, the AAP proxy refines all candidate results corresponding to the user’s precise location $l$. Finally, the MUs securely obtains the refined results from the result refiner.

The entire process relies on the location algorithm, which aims at searching and returning the minimum cloaking region $R$ within the service time $T$ without compromising the user’s location privacy.
Figure 2. Overview of system architecture: an authentication anonymous proxy (AAP), mobile users (MUs), and an untrusted service provider (uSP). The AAP is used for anonymity when a mobile user submits a request, in response to which the uSP generates a sequence of candidate results according to the request from the AAP. Finally, the AAP refines these results and returns the best result to the MUs.

2.2. Road Network Model

We can build a road network model based on the map of a city, which is composed of edges and vertices. Each edge represents part of the road and one vertex refers to an intersection. Generally, a road network determines the movement of traffic, because moving objects always follow the road network, outside of which no moving objects can be observed. In the real world, a road network is composed of a main road and various branch roads. Many different kinds of moving objects can be found on the road, including automobiles, bicycles, and pedestrians. They reach their destinations with a corresponding velocity. In this paper, it is assumed that all users always travel along the road.

2.3. Privacy Model

Three parameters are used to enable each user to accommodate their personalized privacy requirements \([1, 2, 3]\).

- \(k\): It specifies the anonymous level of the user’s location measuring up the \(k\)-anonymity model. In other words, a cloaked region has to contain more than \(k\) different users. The level of privacy increases as the value of \(k\) increases. At the same time, increasing communication overhead becomes problematic.

- \(A_{\text{max}}\): It denotes the maximum cloaking region whose size should be larger than the user’s cloaking region calculated by our proposed algorithm. The value is set approximately 1% of the area of the entire map.

- \(T\): It is the service time. AAP should return the query results within the time \(T\); otherwise, the query is considered to have failed.

- \(v\): It represents the velocity of the mobile user on the road network.

3. Probability-based Prediction Query Algorithm

As mentioned above, a user issues queries frequently within time \(T\) in a continuous query algorithm. Upon satisfying the anonymous requirement for the first time, the anonymous server returns the result, although the cloaking region is unlikely to have the minimum possible size. For example, as shown in Fig. 3, a user is searching for the nearest gas station within 5 minutes drive along the road. Utilizing the existing method, the AAP is capable of returning a result within one minute, whereas it is established afterwards that a more appropriate result would have been possible. This means that the algorithm simply returns a response instead of returning the best response, thereby lowering the QoS. Ideally, the server should return the result corresponding to the minimum anonymous area within the required service time. Obviously, in practice this would almost be impossible. Therefore, we address the above problem by presenting an algorithm based on the principle of determining the minimum cloaking region within the optimal time with high accuracy through scientific prediction. For example, a mobile user shown in Fig. 3 is moving along the road network. When they submit a query request “where is the nearest gas station” at the time \(t_1\), the anonymous cloaking region may be \(C_1\) if it meets the anonymous requirement of the traditional method. After that, the continuous query proposed by Chow et al. allows the mobile user to choose \(C_2\) as the anonymous cloaking region once it implements the anonymous requirement \((k=3)\) according to the moving characteristics of the mobile user in the limited service time. However, we can see from Fig. 3 that \(C_4\) is the minimum region for achieving privacy protection if it can satisfy the privacy level. Therefore, a novel algorithm that would select \(C_4\) as the minimum cloaking region to realize anonymity in a limited service time \(T\) would need to be designed in this work.
In this paper, we employ the user’s historical location information to conduct a reasonable prediction. In reality, the velocity of the user can generally be expected to remain constant instead of changing too much within a certain time. Moreover, a user always habitually chooses the same intersection with a certain probability when faced with several intersections. It indicates that the probability of a user choosing a certain intersection stays the same on the whole. Based on this, we build a road network that can simulate users’ state of motion to allow us to assess users’ location information within a certain time.

Within the service time $T$, a user may traverse one or more intersections. The anonymous region will always change, some of which would meet the anonymous requirement and some of which may not. During this progression, there should be an optimal moment with the smallest cloaking region; thus, we should be able to get close to the best time, while at the same time calculating the minimum area through prediction. Obviously, the area obtained at any given time should always be greater than at the optimal time. Thus, on determining a mobile user’s location according to prediction at some point, we can first calculate the anonymous region and compare it with the predictable region afterwards to judge whether we have reached the optimal time. If not, we should push the time to the next moment as the current time and start a new round of prediction until we succeed. It is clear that the prediction process is quite an important part of our algorithm. Now, we specially introduce the prediction process. It includes how to calculate mobile users’ location probability distribution at the next point in time as well as how to update their geographical coordinates, and finally, how to form the anonymous group according to the user’s location information.

### 3.1. Predict the user’s location probability distribution at the next point in time

We first need to build a road network model. However, it is quite difficult to determine users’ locations are because they have various ways to reach the destination. Therefore, a method based on probability is proposed in this paper. The probability approach aims at predicting the uncertainty of the location of users on the road by tracking their historical driving trajectory. We also observe that it is the intersections that lead to the uncertainty because they present users with different traveling possibilities. Next, the historical moving predicted probability matrix $I(v_{ui,j})$ is defined to capture the probability location information of the user $u_i$ with the speed $v_{ui,j}$ at an intersection $V_j$ with $n$ road segments:

$$I(v_{ui,j}) = \begin{bmatrix} p(e_1|e_1) & p(e_1|e_1) & \cdots & p(e_m|e_1) & \cdots & p(e_n|e_1) \\ p(e_1|e_2) & p(e_2|e_2) & \cdots & p(e_m|e_2) & \cdots & p(e_n|e_2) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ p(e_1|e_n) & p(e_2|e_n) & \cdots & p(e_m|e_n) & \cdots & p(e_n|e_n) \end{bmatrix}$$

where $e_1, e_2, \ldots, e_m, \ldots, e_n$ are the road segments that share the speed $v_{ui,j}$ at this intersection $V_j$, and $p(e_m|e_n)$ represents the probability of the user taking road $e_m$ when they are currently traveling along road $e_n$.

As previously mentioned, the AAP can record the location information once these users have sent requests. Hence, the historical trajectory information stored on the server can be applied to determine the number of times the user has traveled on each road segment and the number of different path options at every intersection. We also define the other matrix named the historical trajectory matrix $H(u_i)$, which is utilized to calculate the matrix $I(v_{ui,j})$.

$$H(u_i) = \begin{bmatrix} N_{1,1}|N_1 & N_{2,1}|N_1 & \cdots & N_{m,1}|N_1 & \cdots & N_{n,1}|N_1 \\ N_{1,2}|N_2 & N_{2,2}|N_2 & \cdots & N_{m,2}|N_2 & \cdots & N_{n,2}|N_2 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ N_{1,n}|N_n & N_{2,n}|N_n & \cdots & N_{m,n}|N_n & \cdots & N_{n,n}|N_n \end{bmatrix}$$

$N_n$ denotes the number of times a user traveled along the road segment $e_n$ during a period of time, and $N_{m,n}$ represents the number of times the user changes to another road segment $e_m$ when traveling along the road $e_n$. We can obtain this
information directly by using the equation \( p(c_m|c_n) = N_{m,n}/N_n \). Meanwhile, the following holds:

\[
\sum_{\substack{j=1, j \neq i \in \mathbb{N} \backslash \{i\}}}^{m} N_{j,i} = N_i, i = 1, 2, \ldots, n
\]  

(1)

Therefore, \( (p(e_i), p(e_j|e_i)) \) is computed utilizing \( (N_i, N_{j,i}, e_i) \) according to statistics. \( (p(e_i), p(e_j|e_i)) \) is given by the following equation:

\[
p(e_i) = \frac{N_i}{\Sigma N_i}, p(e_j|e_i) = \frac{N_{j,i}}{N_i},
\]  

(2)

where \( \Sigma N_i \) is the whole number of the user traveling on the road \( e_i \) on the road network and \( p(e_i) \) presents the probability of the user moved on \( e_i \) within a limited time.

**Example 1.** This example explains what the above two matrices mean. A mobile user shown in Fig. 4 wants to reach \( B \) from \( A \). There exists two ways for him/her to arrive at the destination. \( (N_i, N_{j,i}, e_i) \) expresses the historical trajectory information of a mobile user driving on the road \( e_i \), where \( N_i \) indicates the total number of users on the road \( e_i \) and \( N_{j,i} \) in the matrix \( H(u_i) \), shows the number of users moving from road \( e_i \) to road \( e_j \) \((i = 1, j = 2; i = 2, j = 3, 4; i = 3, j = 5; i = 4, j = 6)\). \( (N_i, N_{j,i}) \) can be obtained from AAP. Therefore, \( (p(e_i), p(e_j|e_i)) \) can be counted directly utilizing equation (2).

![Diagram](image)

(a) one way

(b) the other way

Figure 4. Illustration of a mobile user moving to \( B \) from \( A \). The two ways to reach the destination are shown in (a) and (b), which are utilized to explain the meaning of \( H(u_i) \) and \( I(v_{u,i,j}) \).

Using this approach, we selected the movement information of 10000 users randomly to form a database \( F \) stored in AAP. These users are utilized to construct the matrixes.

**Definition 1. (probability distribution of a user in different locations)**

Considering that mobile users are not stationary, the user perhaps appears at various locations along the road network at different moments in time. A user’s probability of appearing at different geographic coordinates within a limited time frame can be defined as follows:

\[
((x_{t_1}, y_{t_1}), p_{t_1}), (x_{t_2}, y_{t_2}), p_{t_2}), \ldots, (x_{t_j}, y_{t_j}), p_{t_j}), \ldots, (x_{t_n}, y_{t_n}), p_{t_n}),
\]

where \( p_{t_j} \) is the probability of the user \( u_i \) being located at the coordinates \( (x_{t_j}, y_{t_j}) \) in the given anonymous region \( C \) at the time \( t_j \). Let \( p_{t_j} \) be described by the equation below:

\[
\Sigma_{j=1}^{n} p_{t_j} = 1,
\]  

(3)

where the default value of \( j \) is set to 1 as shown in (3) above. The computation of \( p_{t_j} \) is introduced in Section 3.2.

As we only consider the anonymous problem in this paper, we next need to employ \( p_{t_j} \) to compute the final probability of the user being in an anonymous region at some point.

There the probability of user \( u_i \) being in an anonymous area \( C \) along the road \( e_m \) when they are currently on road \( e_n \) at the time \( t_j - 1 \), can be calculated for the next point \( t_j \) in time according to \( I(v_{u,i,j}) \):

\[
p_{t_j}^{C} = p(c_m|c_n) \times p_{t_j}
\]  

(4)

**Example 2.** Fig. 5 shows the possible routes along which a mobile user may drive from \( A \) to \( B \) on the road network. Thus, there are two ways for the user to reach the destination. The mobile user enters cloaking region \( C_1 \) or \( C_2 \) for location privacy protection with a certain probability, which can be calculated by \( (p_{t_1}, p_{t_2}, e_m, t_1) \), where \( (p_{t_1}, p_{t_2}, e_m, t_1) \) denotes the location probability information of the mobile user at time \( t_1 \). \( p_t \) indicates the probability of the user being on road \( e_1 \),
and $p_{j,i}$ equaling $p(e_j|e_i)$ in matrix $I(v_{u,i,j})$ shows the probability of the user changing from road $e_i$ to road $e_j$. Then, the probability of entering $C_1$ or $C_2$, respectively, is computed by:

$$p_{C_1} = 0.6 \times 0.8 \times 0.9 = 0.432$$
$$p_{C_2} = 0.6 \times 0.8 \times 0.1 \times 0.6 = 0.0288$$

The values of $p_i$ and $p_{j,i}$ are obtained by using formula (2).

**Figure 5.** Example showing the two possible routes, (a) and (b), for a mobile user traveling to $B$ from $A$. These routes are utilized to compute the probability of entering into $C_1$ and $C_2$, respectively.

### 3.2. Update the user’s geographical coordinates

Although users move randomly, a large volume of experiments has demonstrated that the process by which the velocity varies approximates a Gaussian distribution.

Let $v_{u_i,t_j}$ be the average speed of user $u_i$ between time stamps $t_{j-1}$ and $t_j$ (e.g., $\Delta t_j = t_j - t_{j-1}$ named a unit time). As a result, $\Delta v_{u_i,t_j}$ is the difference between a pair of velocities ($v_{u_i,t_j}$ and $v_{u_i,t_{j-1}}$) of user $u_i$.

$$\Delta v_{u_i,t_j} = v_{u_i,t_j} - v_{u_i,t_{j-1}},$$

where $\Delta v_{u_i}$ obeys the Gaussian distribution and we assume all mobile users are independent with an identical distribution. Further,

$$\Delta v_{u_i} \sim N(v_{u_i,0}, \sigma_{u_i}),$$

where $v_{u_i,0}$ is the average of $\Delta v_{u_i}$ for the whole period and $\sigma_{u_i}$ is the variance of $\Delta v_{u_i}$. Therefore, we finally have the location information $L_{u_i}(x_{t_j}, y_{t_j}, v_{t_j}, p_{t_j}, t_j)$ of user $u_i$ at the time $t_j$. We can take the velocity as being constant for a limited period of time; hence, the distance covered by the user is calculated within $\Delta t_j$ by the following formula:

$$D_{u_i} = v_{t_j} \times \Delta t_j.$$  

The AAP first has to decide which road the user is traveling along and in which direction while driving to the intersection $V_j$, by exploiting the location information $L_{u_i}$. The possible location of the user can be estimated by using the moving distance $D_{u_i}$ with the corresponding probability $p_{t_j}$ in a cloaking region at the time $t_j$. Finally, the AAP updates the geographical coordinates of the user and generates the new $L_{u_i}$. The process of computing $p_{t_j}$ is shown in Example 3.

**Example 3.** A mobile user is moving from $A$ to $B$ (Fig. 6). When arriving at intersection $V$ at the time $t_1$, the user will either be inside or outside the anonymous area $C$ at the next moment $t_2$, depending on the distance $D_{u_i}$ traveled within one unit of time. In other words, if

$$D_{V,E} \leq D_{u_i} \leq D_{V,F}$$

the user will be in $C$ at $t_2$ ($D_{V,E}$ is equal to the distance between $V$ and $E$ and $D_{V,F}$ has the similar meaning); otherwise, he/she will be outside of $C$. Finally, the location information of the user can be updated.

However, the probability of the user entering anonymous area $C$ is unknown. If the user wants to enter $C$, their difference $\Delta v$ is required to be in the range $D_{V,E} - v_{t_1}$ and $D_{V,F} - v_{t_1}$ at the time $t_j$. The equation is as follows:

$$p_{t_j} = \int_{D_{V,E} - v_{t_1}}^{D_{V,F} - v_{t_1}} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \, dx.$$  

8
$k$ is assumed to be 0.5, 0.5, 0.8, and 0.6, respectively. If the value of $t$ appear in the blue box with a certain probability at the time $C$ in cloaking region $P$.

3.3. Form the anonymous group

We can predict the location information of every user at $t_j$ on the basis of $t_1$. We next discuss how to form the anonymous group according to users’ location information. Because it is difficult to accurately determine the user’s location at the time $t_j$, the precise anonymous region is impossible to obtain, whereas the user’s location information with the corresponding probability is readily available. For example, as shown in Fig. 7 (a), a target user ($R$) is outside the cloaking region represented by the blue box for location privacy protection with privacy level $k = 3$ at the time $t_1$ and we assume that $R$ will move into the region at the next time $t_2$. Formation of an anonymous group becomes possible when at least two users among those being considered for the anonymous group, including users such as $A$, $B$, $C$, and $D$, are in the cloaking region at the current time. Two possible scenarios are shown in Fig. 7 (b) and (c) at the time $t_2$. Obviously, it is not possible to confirm whether this will meet the requirement of $k - NNs$. The algorithm we use can efficiently compute a cloaking region that can guarantee the presence of at least $k$ users, including the target user, in the area with high probability.

![Figure 6](image)

**Figure 6.** Illustration of the situation in which the mobile user is either inside or outside of the cloaking region. (a) indicates that the mobile user is outside of $C$ if $D_{u_i} < D_{V,E}$ holds. (b) shows that the mobile user is inside $C$ if $D_{V,E} \leq D_{u_i} \leq D_{V,F}$ holds.

![Figure 7](image)

**Figure 7.** Example of forming the cloaking region

Next, a few preliminaries are provided to facilitate our presentation.

**Definition 2. (Anonymous probability)**

We define $\text{Pred}_{k,n}(p^C_{u_1}, p^C_{u_2}, ..., p^C_{u_m})$ as the anonymous probability that more than $k$ users from the anonymous group with $n$ mobile users appear in the cloaking region $C$ at the same time. $p^C_{u_i}$ indicates the probability of user $u_i$ appearing in cloaking region $C$. $\text{Pred}_{k,n}(p^C_{u_1}, p^C_{u_2}, ..., p^C_{u_m})$ holds:

$$\text{Pred}_{k,n} = \sum_{m=k}^{n} C_n^m p^C_{u_1} \times p^C_{u_2} \times \cdots \times p^C_{u_m} \times (1 - p^C_{u_{m+1}}) \times (1 - p^C_{u_{m+2}}) \times \cdots \times (1 - p^C_{u_m}).$$  \hspace{1cm} (9)

The concept of anonymous probability is illustrated in Fig. 7. $A$, $B$, $C$, and $D$ are possible users who are expected to appear in the blue box with a certain probability at the time $t_2$, as shown in Fig. 7 (c). The probability of this occurrence is assumed to be 0.5, 0.5, 0.8, and 0.6, respectively. If the value of $k$ is 3, $\text{Pred}_{3,5}$ is 0.87 via Definition 2. That is to say, more than 3 users are likely to appear in the blue box with 87% probability at $t_2$. Thus, the probability of us achieving anonymity is 87%.

A problem does remain with regard to judging whether the anonymous requirement is met. Now, a threshold parameter $P_k$ is introduced as a standard demanded by the anonymity. In other words, $\text{Pred}_{k,n}$ is required to be far larger than $P_k$ for the anonymity to be successful. If $\text{Pred}_{k,n} < P_k$, the anonymous region would have to be expanded by means of enlarging the anonymous group. Therefore, we need to consider how to add additional users to the anonymous group. The addition of users inevitably leads to increasing the cloaking region while ensuring a high appearing probability. We therefore present a definition to expand the anonymous group appropriately.
Definition 3. (benefit value of the added user)

When a new user \( u_i \) joins the anonymous group with \( n \) existing users, the new anonymous probability of the new anonymous group with \( n + 1 \) users can be determined in terms of Definition 2:

\[
\text{Pred}_{k;n+1} = \sum_{m=k}^{n+1} C_{n+1}^m p_{u_1}^C \times p_{u_2}^C \times \cdots \times p_{u_m}^C \times (1 - p_{u_{m+1}}^C) \times (1 - p_{u_{m+2}}^C) \times \cdots \times (1 - p_{u_n}^C) \times (1 - p_{u_{n+1}}^C).
\]

(10)

Then we define \( BV_{u_i} \) as the benefit value of the added user \( u_i \):

\[
BV_{u_i} = \frac{\text{Pred}_{k;n+1} - \text{Pred}_{k;n}}{|S_{n+1} - S_n|},
\]

(11)

where \( S_n \) means that the area of the cloaking region formed by the anonymous group with \( n \) users and \( S_{n+1} \), indicates the area of the cloaking region formed by the anonymous group with \( n + 1 \) users. \( BV_{u_i} \) increases with an increase in the anonymous probability under the condition that anonymous area undergoes the same change, which means that users who contribute to slightly increasing \( BV_{u_i} \) are beneficial for satisfying the anonymous requirement. The anonymity is successful until \( \text{Pred}_{k;n} > P_k \). Finally, we can calculate the corresponding anonymous region.

Figure 8. Extending the anonymous group when the anonymous requirement fails. Compute \( BV \) of \( E \), \( F \), and \( G \) who are near \( R \) and add the user with the highest \( BV \) into the anonymous group. The cloaking region may then be enlarged (red box).

It is noteworthy that the anonymous area and \( BV_{u_i} \) change upon addition of a new user, which requires us to recalculate them.

What we have elaborated on above is the prediction process, which is a very significant part of our algorithm. When a user submits a requirement to the AAP, it can predict and update his/her location information, and finally conduct the anonymous group. Then the minimum anonymous region is selected at the optimal time (discussed below). This enables the QoS to improve considerably.

In total, the algorithm mainly consists of the following steps. We also introduce the details shown in Algorithm 1.

\begin{itemize}
  \item a) Initialization process
  \begin{itemize}
    \item Step 1. Firstly, the AAP establishes a road network model. The probabilistic information at an intersection \( V \) with \( n \) road segments is obtained by defining two matrices to determine the probability of users passing through each intersection.
  \end{itemize}
  \item b) Query and handling process
  \begin{itemize}
    \item Step 2. A user sends a query in the form of \( Q(ID, l, q, P_{model}) \) to the AAP.
    \item Step 3. Upon receipt of the query, the AAP gathers all users’ current location information, locates them on the road network, and finally generates an initial anonymous region.
    \item Step 4. Based on the users’ location information, the AAP predicts their position at the next moment and continues to compute the relevant anonymous region. By comparing the size of both areas, it needs to judge whether the current time is optimal. If not, go to the next step.
    \item Step 5. We reach the next moment in time and repeat Steps 4 to 5 until the anonymous region at the current time is minimized within the service time \( T \). As a result, the present time is the optimal moment.
  \end{itemize}
\end{itemize}

\begin{algorithm}
\caption{Algorithm flow to obtain the minimum region \( R_{\text{mini}} \)}
\begin{algorithmic}[1]
  \State \textbf{if} \( k > n \) \textbf{then} // \( n \) indicates the number of users in the anonymous group and \( k \) is the privacy level
\end{algorithmic}
\end{algorithm}
2: Compute BV of the user nearest to the target user and add the user with the highest BV to the anonymous group until the privacy level is satisfied; output $R_{ini}$ as the minimum anonymous region;

3: else
4: Compute $ Pred_{k,n}$;
5: if $Pred_{k,n} > P_k$ then
6: Calculate ($S_{ini}, S_{ini+1}, S_{ini+2}, ...$);
7: if $S_{ini} < (S_{ini+1}, S_{ini+2}, ...)$ then
8: Output $R_{ini}$ as the minimum anonymous region;
9: break;
10: else
11: Proceed to the current time as the initial time and restart from 4;
12: end if
13: else
14: Compute BV of the user nearest to the target user and add the user with the highest BV into the anonymous group and restart from 2;
15: end if
16: end if

return valid;

Comparing and finding the minimum anonymous region is discussed in Section 3.4, after which a metric of success is introduced in Section 3.5.

3.4. Compare and determine the minimum cloaking region

Utilizing the above method, the cloaking region can be calculated from the initial time to the end. This ensures that if the current time is the optimal moment, the minimum possible region is obtained for all the moments (i.e., before and after this time).

Firstly, we define the value of the area at the initial time as $R_{ini}$ and compare it with the areas calculated after the initial time. There is a theoretically perfect time when the minimum anonymous region always exists. Furthermore, there is always a moment at which the anonymous region is smaller than at the current time. Thus, all we need to do is to push the next moment to the initial time and perform a new round of prediction until it reaches the perfect time. The prediction process finishes. It is not necessary to compute the anonymous region after this time because the region computed at this time should be the smallest in theory.

However, the difficulty in calculating the exact size of the cloaking region due to the variable travel path and uncertain positions of users is such that we can only predict the trend according to which the area changes and the optimal time. Thus, we predict the value of the anonymous region $S_{pre,i}$ at the time $t_i$, and then compare it with the initial region $R_{ini}$. If this initial region coincides with the minimum that is theoretically possible, which means

$$R_{ini} < \min(S_{pre,ini+1}, S_{pre,ini+2}, ..., S_{pre,ini+n})$$

, then the initial time is considered the optimal time. If not, push the next time to the initial time and start a new round until the minimum cloaking region is obtained while meeting the anonymous privacy requirement.

Fig. 9 shows details of the prediction process with the server time $T$. The AAP handles the query at the time $t_1$, and then predicts the anonymous region and computes every one between $t_1$ to $T$. Forward the time to the next moment when exciting the smaller region. Continue executing the round until the minimum region is found. This approach enables us to find the minimum possible cloaking area at the time $t_4$ after several rounds of prediction.

3.5. Metric of success

We finally obtain an optimal moment $t_{op}'$ and its cloaking region $S_{pre,t_{op}'}$ based on the theory. At the same time, we also determine an optimal time $t_{op}$ and its anonymous area $S_{pre,t_{op}}$ in our experimental simulation system. However, there are inevitably some errors in the prediction process. We define a successful prediction as follows:

$$t_{op} = t_{op}'$$

or

$$R_d = \frac{|S_{pre,t_{opp}} - S_{pre,t_{op}'}|}{S_{pre,t_{op}'}^i} < \delta,$$

where $R_d$ indicates the decrease rate between the anonymous area $S_{pre,t_{op}'}^i$ based on the theory and the cloaking region $S_{pre,t_{op}}$ obtained by performing the simulation experiment and $\delta$ denotes a small constant. The smaller the value of $\delta$, the closer $S_{pre,t_{op}}$ is to $S_{pre,t_{op}'}$. In this work, the ability of the experimental result to satisfy either of the two above-mentioned conditions proves that our proposed algorithm is successful.
Figure 9. Process of predicting the optimal moment. The mobile user sends a request within the server time $T$. The AAP handles the request at the time $t_1$ defined as the initial time and generates the anonymous region and compares it with the areas calculated after $t_1$. Then push $t_2$ to the initial time and the prediction begins because there existing a small cloaking region after $t_1$. The next time is then considered the initial time once a smaller region is found. The round is repeated until the minimum cloaking region is found within $T$. Finally, the cloaking area is determined to be optimal at $t_4$.

4. Performance Analysis and Experimental Result

In this section, we experimentally evaluate our proposed PBPQ algorithm. Firstly, we define two simple metrics to measure the performance of the PBPQ algorithm in Section 4.1, followed by the experimental setup in Section 4.2. Finally, the evaluation results are discussed in Section 4.3.

Figure 10. Map of Oldenburg

4.1. Evaluation Metrics

The anonymity success rate or anonymity error rate and the region decrease rate $R_d$ are the two parameters that are used to assess the effectiveness of our algorithm by assessing the impact of three aspects, including the prediction time, privacy lever, and probability threshold.

4.2. Experimental Setup

We use a map of the city Oldenburg and data generated by the Thomas Brinkhoff network-based generator [28], which contains 80 000 real locations and 4000 mobile users moving to their destinations for a certain time. The length and width of the entire map shown in Fig. 10 is approximately 10 000 and 15 000, respectively, i.e., x-coordination takes a value from 0 to 10 000 and y-coordination from 0 to 15 000. There are various kinds of mobile users including walking men, cyclists, and drivers. The user information can be extracted from the generator. Their velocities vary from 10 to 250.

We assume that mobile users will continue moving along the road network and always appear during the service time. The AAP is able to determine the target user’s route and collect other users’ location information. The time interval $\Delta t$ is
an approximate constant value and the AAP divides the service time $T$ into $(t_1, t_2, ..., t_n)$ in the interval of $\Delta t$. We specify period $T$ as ranging from 1 to 8 to fulfill the simulation performance.

We implement our proposed $PBFPQ$ algorithm by $C++$. The performance evaluation of the AAP is simulated on a PC running Windows 7, equipped with an Intel i3-2120 3.30 GHz processor and 2 GB RAM.

4.3. Evaluation Result

Three factors including prediction time, privacy level, and probability threshold are evaluated on the performance of our proposed algorithm in the following experiments.

4.3.1. Impact of prediction time

In this section, we investigate the impact of various prediction times in our algorithm. Special cases were ruled out by choosing the privacy level as 6, 8, and 10, all of which have the same probability threshold, i.e., $p_k = 0.8$. In this experiment, the value of $T$ gradually increases from 1 to 8 to satisfy users’ different demands.

As shown in Fig. 11, we select three privacy levels to present the anonymity success rate versus the prediction time, and we set the value of $\delta$ as 0.1 at the same time. Firstly, it can easily be shown that the anonymity success rate of all three curves are relatively ideal when the prediction time is below 4, reaching a high of 95% and exceeding 80% most of the time. This demonstrates the ability of the algorithm to accurately predict the optimal time and conduct the cloaking region successfully within a certain time. The anonymity success rate decreases slowly as the prediction period increases, because the velocity continues to vary and the error in determining the speed grows along with an increase in the prediction duration. The velocity we employed is based on the initial velocity, which may not reflect reality and even causes prediction errors. Thus, the longer the prediction process, the lower the anonymity success rate. Nevertheless, the anonymity success rate remains steadily 80%. In real life, the prediction duration of most queries submitted by users ranges from 2 to 5 and this rate can be guaranteed at a very high success level by our proposed algorithm.

Figure 11. Anonymity success rate versus prediction duration

In addition, we also determine the experimental performance by using the anonymity error rate. Fig. 12 shows the anonymity error rate with $\delta = 0$ and $\delta = 0.1$, respectively. The curve in Fig. 12 (a) shows that the error rate is about 20%, although mostly below 15%. That means the imperfect anonymous region resulting from our predictions is extremely close to the perfect region even though at times we fail to predict the minimum region at the optimal time. In our experiment, the error rate is far smaller, i.e., less than approximately 8%, when $\delta$ is chosen as 0.1, as shown in Fig. 12 (b). We can see that the error rate decreases sharply when $\delta = 0.1$ compared with $\delta = 0$ shown in Fig. 12 (a). The reason is that by setting $\delta = 0$, it indicates the theoretical value $S'_{pre,t_{opt}}$ is equal to the experimental value $S_{pre,t_{opt}}$, which would indicate that we could exactly predict the minimum anonymous region at the optimal time. The results are ideal with $\delta = 0$. However,
the evaluated results with \( \delta = 0.1 \) allow mistakes in our experiments. The accuracy with which our approach is capable of predicting the minimum cloaking region allows us to regard it as a successful prediction algorithm in terms of Fig. 11 and Fig. 12.

In summary, although our approach performs slightly more accurate within the service time \( T \), those variations are still acceptable with negligible error.

4.3.2. Impact of privacy level

This section examines the impact of privacy level \( k \) on the performance of the cloaking algorithm. Without loss of generality, we choose several prediction times such as \( T = 2, 4, \) and 6. The result is presented in Fig. 13 and Fig. 14.

![Figure 13. Anonymity success rate versus privacy level](image1)

![Figure 14. Region decrease rate versus privacy level](image2)

In Fig. 13, the anonymity success rate for all three prediction times increases slowly versus \( k \), and finally stabilize. The anonymity success rate remains with approximately 80%. The reasons for this are the following. First, users’ locations affect the cloaking region significantly when \( k \) is small. When a user happens to be in an unpopulated area, its cloaking region can be relatively large because it needs to find at least \( k - 1 \) other users. Once he/she enters an intensive area, the area of the cloaking region change to the extent that the prediction result is affected. On the other hand, when the user anonymous level is lower, the corresponding perfect region should preferably also be small. Our algorithm usually produces a relatively large value of \( k \) on account of the calculation progress of the area. When the value of \( k \) grows, our algorithm can steadily predict the user’s location and determine the optimal time at a considerable anonymity success rate.

Fig. 14 indicates that the difference of the rate between the optimal anonymous region and the actual cloaking region we predict is roughly between 15 and 5%, changing slightly as \( k \) increases. This proves that our proposed method can predict the minimum anonymous region at the optimal time with a very small error.

Ultimately, this indicates that the \( PBPQ \) algorithm can mostly predict the optimal time with high accuracy.

4.3.3. Impact of probability threshold

In this section, we change the probability threshold to test the influence on the anonymity success rate, with the prediction time ranging from 2 to 6 as usual. From Fig. 15, it is obvious that the success rate remains at a high level, mostly above 85%. We also observe that it decreases slightly as the probability threshold decreases before finally stabilizing. This result involves two aspects: the first, namely that as the threshold \( p_k \) grows, it becomes increasingly difficult to form the cloaking region such that the anonymous requirement is met. Considering the potential error in every user’s location, when more users are added to the anonymous group, the predicted error becomes more serious. However, we find that a relatively larger anonymous group produces a more satisfactory anonymous result and a higher success rate for the stable area mentioned in Section 4.3.2.

![Figure 15. Anonymity success rate versus probability threshold](image3)
5. Conclusion

This paper proposes a novel algorithm named the probability-based prediction query (PBPQ). Our algorithm is based on the principle of utilizing the probability method to predict the location of mobile users. This has the purpose of generating the $k$-anonymity model and ultimately determining the optimal cloaking region by comparing requests for the LBSs with limited time frames. We showed that our proposed algorithm is superior to existing methods. The experimental results show that our PBPQ algorithm can form the cloaking region effectively without the need to frequently upload the target user’s location information to the anonymous server. The cloaking region is reduced significantly, a remarkable improvement in QoS is achieved, and the communication overhead is decreased dramatically.

Acknowledgement

This work was supported by the Key Program of NSFC-Guangdong Union Foundation under Grant U1401251, the National Natural Science Foundation of China under Grant, No. 61373170, the National Natural Science Foundation of China under Grant, No. 61172053 and Guangxi Colleges and Universities Key Laboratory of cloud computing and complex systems foundation, No. 15203.

References


[22] X. Pan, J. Xu, X. Meng, Protecting location privacy against location-dependent attacks in mobile services, IEEE Transactions on Knowledge & Data Engineering (2011) 1506–1519.


