International Journal of Hybrid Information Technology
Vol.7, No.4 (2014), pp.139-152
http://dx.doi.org/10.14257/ijhit.2014.7.4.12

Privacy-Preserving Proximity Based Services

Min Li[1, 2], Ruijin Wang[1], Zhiguang Qin[1] and Cong Wang[1]

[1]*School of Computer Science & Engineering, University of Electronic and Technology of China, Chengdu, China*
[2]*College of Computer Science, Sichuan Normal University, Chengdu, China*
**Corresponding author: lm_turnip@126.com*

Abstract

Recently, with the dramatic popular of geo-social networks and location based services (LBS), more and more mobile users are willing to enjoy the proximity services, which is a friend-alarm when the buddies happen to be in proximity. However, the services provider (SP) and some compromised buddies could try to steal the user's exact location information. Hence, location privacy preserving is still a challenge question in proximity based services. Unfortunately, existing methods only consider Euclidean distance to calculate neighbors, which would bring great deviation for distance measurement in the actual terrain. To my knowledge, this article is the first time extending to the 3D surface and proposes a novel technique for the privacy-preserving proximity services. The technique utilizes simplicial triangulation decomposition on 3D surface, and the user's location is generalized with a triangular area space, then a more precise proximity distance measurement method based on the triangle fractal is proposed. Next, we design a privacy-preserving proximity query protocol, which provide complete privacy with respect to the SP and the buddies. Final, the experiments shows that our methods have practicality in the services quantity and privacy.

Keywords: *Proximity services, location privacy-preserving, 3D smooth surface, geodesic distance, the simplex segmentation, spatial generalization, commutative encryption*

1. Introduction

Location based services (LBS) have become popular due to the explosive growth of GPS-enabled mobile devices. Moreover, with the growth of LBS and social network services (SNS), a geo-social network is constructing connections between location information and social network. A good example is proximity based services, which is so-called "friend-alarm": alert me when any of my friends (also called buddies) is within 10 miles of me, so that I can contract with them. Such friend-alarm has more and more brought to the attention of the mobile application.

Proximity based services are one class of LBS, which process spatial range queries to check when moving buddies approach the vicinity of the query issuer. It is important to note that proximity based services focus on two aspects. First, spatial range is the circle centered at the issuer's location and having the proximity threshold radius defined by the issuer or moving buddies. Second, spatial range queries are with respect to the current location of the query issuer and moving buddies. Usually, proximity service is computed by a third party service provider (SP) or the issuer's itself, so it is required the mobile user to share his (her) location with SP or his (her) buddies. Because that location information could contain useful information which is far more than the location itself, location sharing could lead to the leakage of the key information. This is an issue that has been caused wide public concern in

ISSN: 1738-9968 IJHIT

the last year in LBS. Also, and this is a trade-off between protecting users' location privacy and acquiring proximity services quantity.

Therefore, a privacy-preserving proximity services should be designed to meet the need as follows, regarding a) to guarantee strong location privacy requirement, and b) to release as little location information as possible. Because that the user does not trusted the SP and may not be willing to share completely location information with the buddies. Therefore, in privacy-preserving proximity services, it is needed that, a) SP should not acquire the issuer's exact location information; b) a user may be agreed to obtain that any of the buddies is located in neighborhood, while only to get limited location information about the buddies.

Existing methods mainly classify two categories, one is spatial generalization, and another is the homomorphic encryption. Unfortunately, both of them consider that mobile users move in the Euclidean space, and the Euclidean distance between two users is introduced to measure the vicinity. However, the actual terrain is 3D surface, in which the Euclidean distance will no longer be suitable because of its huge deviation. Therefore, a protection privacy executable on 3D surface is needed in proximity services.

In this paper, we break through the limitation of traditional Euclidean distance, and propose a new accurate proximity method based on the triangle fractal to meet the proximity services in the actual terrain. The main contributions of this paper are the following.

- To our knowledge, this paper firstly considers privacy-preserving proximity services on 3D smooth terrain surface.
- This paper introduces the triangle fractal on 3D smooth surface, and then generalizes the user's location in a triangular area space, which can effectively protect the user's location privacy against the SP and the buddies.
- This paper proposes a method for measurement of the proximity range that is based the approximate geodesic distance between the two user's generalization regions, which can obtain more accurate results than using the Euclidean distance for 3D smooth surface.
- This paper designs a privacy-preserving proximity query protocol, which depends on commutative encryption on hashed granule index of location generalization. This protocol can provide complete privacy with respect to the SP and the buddies.

2. Related Work

Previous works for privacy preservation in LBS mainly focus on the following three dimensions: a) location privacy b) identity privacy c) query privacy, that is, "who is it? Where is it? What to do? " An effective means to protect the privacy is anonymous, in which *K*-anonymous is the most commonly and the most effective anonymous method. *K* anonymous initially by Samarati [1] and Sweeney [2], it is mainly used for making any user with another *K*-1 undistinguished. Based on this, Machanavajjhala put forward *L*-diversity concept [3], whose basic idea is that each *K* anonymous group should contain at least *L* difference of sensitive information.

LBS can provide services for mobile users based on their location, including a) place of interesting (POI) finder (*e.g.*, a *K*NN query), mobile users query stationary, b) proximity query, mobile users query mobile users. In LBS, the purpose of location privacy-preserving ensure that the mobile user's exact location don't leak when they enjoy the high quantity services. Existing location privacy-preserving technique in POI finder mainly focuses on false dummies [11], space transformation [12], and spatial cloaking [1-10]. Indeed, both false dummies and space transformation rely on the fake location, with which will bring huge deviation for proximity query, so both of them are not suitable for proximity query.

Spatial cloaking [4, 5] is based on the idea of generalizing users' exact location, that is, users send the generalization region to SP instead of the exact location, and Casper [6], Interval [5] and Hilbert-based cloaking [10] are the typical anonymous technique. Some of these techniques can be applied for proximity-based services. Among these proposals, Zhong et al. propose Pierre [20]. S. Mascetti [13, 14] proposes that the measurement of the distance between two mobile users are using the distance of two generalization granule where the two users located, and the works present two protocol that the query user send to SP with encrypted or hashed the granule index, which can prevent the leaking of the query user's exact location from SP and the buddies.

Another solution is the encryption-based technique. Zhong *et al.*, propose Louis and Lester [20], which are centralized secure computation protocols based on public key cryptography. Louis is a three-party secure computation. However, Louis incurs in significant communication overheads and Lester in high computational costs. Li *et al.*, [15] use homomorphic encryption to process upon the encrypted location information. Therefore, these techniques can guarantee location privacy because no location information disclosed with respect to the SP and buddies, but their efficiency is very low.

Unfortunately, the two mentioned techniques are designed for Euclidean space, and not consider that the actual terrain is 3D surface, in which the Euclidean distance will no longer be suit because of its huge deviation. For this reason, this paper introduces approximate geodesic distance in triangulation on 3D smooth surface. B. Delaunay [16] initial presents Delaunary triangulation, we make the expansion on 3D surface [17, 18]. J. Tang [19] proposes a computing geodesic distance on a triangle mesh. So, in this paper we adopt these methods for precise neighbor, and we introduce a commutative encryption to guarantee privacy with respect to the SP and the buddies.

3. System Model

3.1. Proximity Services

Proximity services should include three participators: 1) the location querier, 2) the location publisher, 2） the proximity Services Provider (SP). Notice that a mobile user can be a querier and a publisher at the same time. In this paper, it is provided that the buddies are predefined, and the buddy list can be dynamic updated. A querier *Q* initiates the proximity query for each of her buddies *P*, if the following condition is satisfied, then *P* is in the proximity of *Q*.

$$d(loc_Q, loc_P) < \Theta_Q \tag{1}$$

Where $d(loc_Q, loc_P)$ denotes the geodesic distance between *Q* and *P* and Θ_Q is the proximity threshold given by *P*.

3.2. Privacy Requirements

A Naïve proximity services method processes as follows: when a user issues a proximity request, she will update her location to the SP, which charges the distance calculation. In the Naïve method, the user's exact location privacy is disclosed with respect to the SP and her buddies. In this paper, we focus on how to acquire the services as little as possible leaking location information, that is, the user should be able to control the location information to be closed.

This paper introduces the triangle fractal on 3D smooth surface, and then generalizes the user's location in a triangular region. It is considered that any two triangle generalization areas do not interact and the unions of all triangle generalization area are exactly the whole spatial domain. All users express their privacy requirements by specifying the granularities of the minimum generalization region, that is, MGR. While in theory each user can choose a different MGR value, for simplicity, this paper considers a unified MGR value for each user.

3.3. Privacy Threats

It is assumed that both SP and buddies are considered as potential adversaries, moreover, both of them are semi-honest and independent. That is, they will follow the query process and will not collude with each other, but they are likely to try to obtain the issuer's exact location. At

If user *P* who is one of *Q*'s buddies is a neighbor of *Q*, then *Q* will discover that *P* is located in the circle with the radius Θ_Q, which is defined by *Q*. So, user *P* cannot control the location information disclosure to his buddies. Therefore, this system requires a minimum proximity threshold $\Theta_{\min}$, which is smaller than the defined proximity threshold value of each user. That is, there is a safe proximity range for each user to disclose to buddies.

4. Background

4.1. The Simplex Segmentation Method

The simplex segmentation is a triangulation method which is a very important pretreatment technology for the numerical analysis and graphics. B.Delaunary discussed in detail Delaunary triangulation, which is evolved from Voronoi-graph and is the most close to the rules of the triangulation.

Definition 1 A Delaunay triangulation for a set *P* of points in a plane is a triangulation DT(*P*) such that no point in *P* is inside the circumcircle of any triangle in DT(*P*), and it maximize the minimum angle of all the angles of the triangles in the triangulation.

4.2. Approximate Geodesic Distance on a Triangle Mesh

Similar that Euclidean distance is the shortest distance between two points on the plane, the geodesic distance is the local shortest path between two points on the 3D space. For example, the geodesic distance between two points on sphere is the shortest arc to connect two points. In this paper, we introduce the work [19], in which to map the different planes of points to the same plane is by constructing a virtual point. The following descriptions are about that the two triangles are not in the same plane. There are two collinear ($v_x v_{x+1}$) triangles about $\triangle v_x v_{x+1} v_{x-1}$ (o_i is the one interior point) and $\triangle v_x v_{x+1} v_{x+2}$ $\triangle v_x v_{x+1} v_{x-1}$ (o_j is the one interior point), It is assumed that the distance from o_i to v_x ($dist(o_i, v_x)$) and from o_i to v_{x+1} ($dist(o_i, v_{x+1})$) is precomputed. So, the geodesic distance ($dist()$) between o_i and o_j is computed as follows.

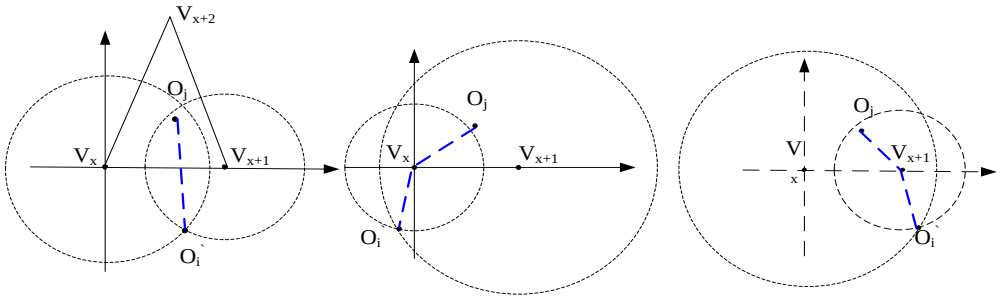


Figure 1. Approximate Geodesic Distance on Two Triangles

Step 1: to map o_i to $o_i'(x_{o_i}', y_{o_i}')$ in $v_x v_{x+1} v_{x+2}$, meet the equation 1) and 2).

$$x_{o_i}'^2 + y_{o_i}'^2 = dist(o_i, v_x)^2 = d_x^2 \quad 1)$$

$$(x_{v_{x+1}} - x_{o_i}')^2 + y_{O_i}'^2 = dist(o_i, v_{x+1})^2 = d_{x+1}^2 \quad 2)$$

Where $dist(o_i, v_x) = \|o_i v_x\|$ and $dist(o_i, v_{x+1}) = \|o_i v_{x+1}\|$ ($\|.\|$ means the Euclidean distance)

Equation1) and 2) have two solutions, the point which is far away from o_j is chosen as o_i'.

Step 2: to calculate $dist(o_i, o_j)$, three cases (see Figure 1) is listed as follows.

a) Case 1, $0 \le x_{o_i}' \le x_{v_{x+1}}$ $\quad dist(o_i, o_j) = \|o_i' o_j\|$ 3)

b) Case 2, $x_{o_i}' < 0$ $\quad dist(o_i, o_j) = \|v_x o_j\| + d_x$ 4)

c) Case 3, $x_{o_i}' > x_{v_{x+1}}$ $\quad dist(o_i, o_j) = \|v_{x+1} o_j\| + d_{x+1}$ 5)

4.3. Commutative Encryption

Commutative encryption is a technique for secure multi-parties computation. Informally, a commutative encryption $E : Key\Gamma \times Dom\Gamma \to Dom\Gamma$, defined on finite computable domains, that satisfies commutativity properties:

1) Commutativity: For all $K_1, K_2 \in Key\Gamma$ *we have*

$$E_{K_1}(E_{K_2}(x)) = E_{K_2}(E_{K_1}(x)), \quad 6)$$

That is, $E_{K_1}(E_{K_2}(x)) = E_{K_2}(E_{K_1}(x'))$, *if and only if* $x = x'$.

2) Each $E_K : Dom\Gamma \to Dom\Gamma$ *is a bijection.*

5. System Design

In this paper, we consider to cloak the user's location within a generalization region with a Delaunary triangulation, and then to measure the proximity range with approximate geodesic distance on triangle granule. Final, a privacy-preserving proximity request protocol based on commutative encryption can be presented to process proximity services.

5.1. Spatial Generalization

Spatial generalization is offline run by a query user to cloaking her exact location within a triangulation region, whose spatial granularity is the user's privacy requirement. In this paper, for simplicity, the size of the spatial granularity is preliminary defined by the system, that is, all users have the same privacy profile. Furthermore, the user's location is expressed as a unique granule index i of the generalization granularity $g(i)$.

In this paper, we introduce our research on the triangulation algorithm based on fractal network topology to build triangle model [17]. In the work, sensor nodes should be evenly spread on 3D smooth surface, and the boundary points will be located, then we take advantage of 3DT-ST algorithm to process Delaunary triangulation for the boundary points and SPN nodes [17]. The result of spatial generalization is shown as Figure 2.

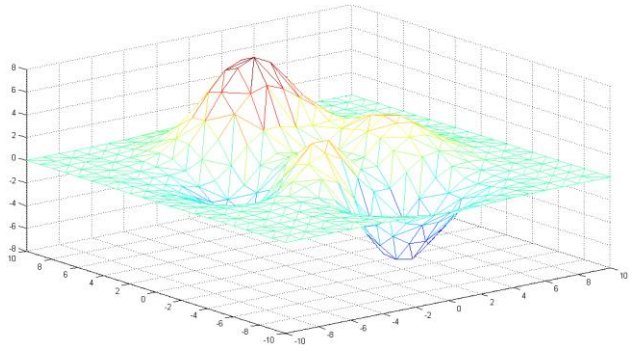


Figure 2. An Example of Spatial Generalization

5.2. Measurement of the Proximity Range

As discussed before, user *Q*'s proximity is located in the proximity circle centered in her location with radius Θ_Q, but due to spatial generalization, the proximity circle should be replace with proximity granule set G_Q, which is the union of granules which intersect with the proximity circle. That is, G_Q meets the equation 7. When user *Q* issues a proximity query, if any buddy *P* locates in one of G_Q, then *P* is the proximity of *Q*.

$$G_Q = \{ j \mid dist(loc_Q, loc_j) < \Theta_Q \} \quad 7)$$

Where loc_Q represents the *Q*'s location and j is the granule index in which each buddy *P* is located.

Before executing proximity services, the geodesic distance between any two triangles is need to pre-computed offline. For simplicity, we will transform the geodesic distance from a point to a plane into the geodesic distance from a point to a point. In this paper, we will choose interior point of the triangle instead of the triangulation granularity, that is, $dist(loc_Q, loc_j) \approx dist(loc_{o_i}, loc_{o_j})$ (where i is the granule index in which Q locates,

o_i and o_j are interior point of the granule index i and j respectively). Therefore, geodesic distance measurement problem can be transformed into the calculation of the shortest path between any two points in the weighted graph, where, the vertex of the graph is the interior point of the triangle, the weigh value on the edge is the geodesic distance between two collinear triangles, which can be first calculated offline. So, a Dijkstra algorithm will be introduced to calculate the geodesic distance from a point to the rest of the points.

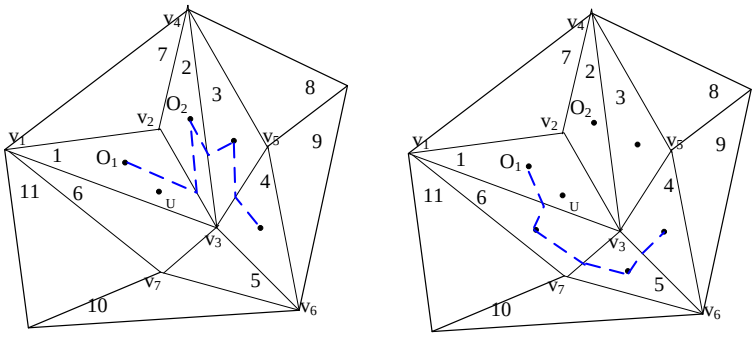


Figure 3. An Example of Calculation G_Q

Figure 3 depicts an example of geodesic distance calculation starting from o_1, when $d_{i,j} = dist(o_i, o_j)$, if the profile is

$\{ d_{1,2} = 5m$, $d_{1,7} = 12m$, $d_{1,6} = 7m$, $d_{2,3} = 2.5m$, $d_{2,7} = 5.5m$, $d_{6,5} = 2m$, $d_{5,4} = 0.5m$, $d_{5,10} = 4m$, $d_{3,4} = 4m$, $d_{4,9} = 3m$, $d_{3,8} = 4.5m$ $\cdots\cdots\}$.

Table 1. The Approximate Geodesic Distance Starting from o_1

o_2	5(1->2)						
o_3	∞	7.5(1->2->3)	7.5(1->2->3)				
o_4	∞	∞	∞	11.5(1->2->3->4)	9.5(1->6->5->4)		
o_5	∞	∞	9(1->6->5)	9(1->6->5)			
o_6	7(1->6)	7(1->6)					
o_7	12	10.5(1->2->7)	10.5(1->2->7)	10.5(1->2->7)	10.5(1->2->7)	10.5(1->2->7)	
…	…	…	…	…	…	…	…
o_j	o_2	o_6	o_3	o_5	o_4	o_7	…

The process procedure is shown in Table 1. Obviously, the geodesic distance between o_1 to o_4 is 9.5(o_1 - o_6 -> o_5 -> o_4), not 11.5(o_1 - o_2 -> o_3 -> o_4).

Final, all geodesic distance is stored in a two-dimensional array by order. When a user Q (located triangle index is 1) issues a proximity request, an algorithm for the proximity range calculation can be executed to find the o_1 row by order. Once the geodesic distance from o_1 point is exceeds the proximity threshold (*e.g.*, $\Theta_Q = 10m$), the algorithm stops. Thus forming the proximity range of Q is $G_Q = \{1, 2, 6, 3, 5, 4\}$.

5.3. Privacy-preserving Proximity Query Protocol

The proximity request protocol is run by a querier Q that wants to search which of her buddies are in vicinity. It is assumed that all users update their buddies list to the SP before the implementation of the protocol. It is shown as Figure 4, the proximity request and response work as follows: When Q send a proximity request to the SP, then the SP send a location request to each of her buddy P, which is contained at the buddy list. Upon receiving the request, each of buddy P replies with a message containing the current hashed location. That is, for each of buddy P, Q receives a tuple $< P, H(i_P) >$, where $H(i_P)$ is the hashed value of granule index i_P where P is located.

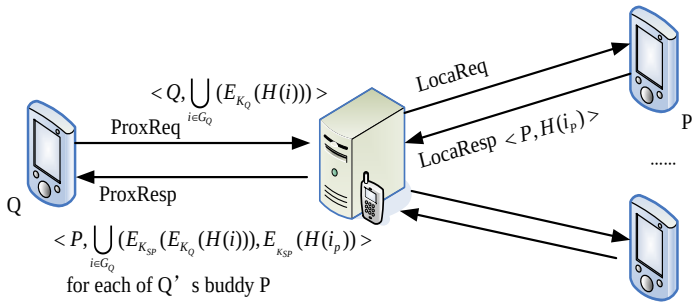


Figure 4. Privacy-preserving Proximity Query Protocol

More precisely, the proposed protocol works as protocol 1.

Statement 1. Assuming there are no hash collisions, $y_P' \in Y_Q'$ iff $i_P \in G_Q$.

Proof. Assuming there are no hash collisions, $h(i_P) \in h(G_Q)$ iff $i_P \in G_Q$

By definition, E_K *is commutative and bijective.* $y_P' \in Y_Q'$ iff $h(i_P) \in h(G_Q)$

So definition of G_Q, P should be in proximity for Q.

In the protocol, the SP is only responsible for forwarding the message, rather than participate in the operation. Therefore, the SP does not learn any of the user's location information and inquires proximity results. Moreover, the querier Q only whether each of her buddy P is in proximity or not, without learning in which granule P is located. So, the protocol can provide complete privacy with respect to the SP and the buddies.

Protocol 1 Security Proximity Request Protocol

Input: G_Q , i_P

Output: V_Q (Q's proximity set)

1) V_Q ={};

2) Q apply hash function h to the set G_Q : $H_Q = \bigcup_{i \in G_Q} h(i)$;

each of buddy P also apply the same hash function h to i_P : $h_P = h(i_P)$;

3) Q and SP randomly choose a secret key $K_Q \in Key\Gamma$ and $K_{SP} \in Key\Gamma$;

4) Q encrypt its hashed sets: $Y_Q = E_{K_Q}(H_Q) = \bigcup_{i \in G_Q} (E_{K_Q}(h(i)))$;

5) Q sends to SP the message ProxReq $< Q, Y_Q >=< Q, \bigcup_{i \in G_Q} (E_{K_Q}(h(i))) >$;

6) SP ships to each of Q's buddy P the message LocaReq,
and P replies with the message LocaResp $h_P = h(i_P)$;

7) (a) SP encrypts Y_Q with its key K_{SP} :

$$Y_Q' = E_{K_{SP}}(Y_Q) = E_{K_{SP}}(E_{K_Q}(H_Q)) = \bigcup_{i \in G_Q} E_{K_{SP}}((E_{K_Q}(h(i))))$$

(b) for each of buddy P, SP encrypt h_P with its key K_{SP} :

$$y_P = E_{K_{SP}}(h_P) = E_{K_{SP}}(h(i_P))$$

(c) SP sends back to Q a tuple

$$< P, Y_Q', y_P >=< P, E_{K_{SP}}(E_{K_Q}(H_Q)), E_{K_{SP}}(h_P) >$$

8) Q encrypts each y_P with Q's key K_Q

$$y_P' = E_{K_Q}(y_P) = E_{K_Q}(E_{K_{SP}}(h_P))$$

9) Intersection computation for P:

if $y_P' \in Y_Q'$ **then** $V_Q += P$

6. Experimental Evaluation

In this section, we will evaluate the performance of our protocol (donated as PPQR) and compare it with C-Hide&Hash(donated as C-H&Hash) and C-Hide&Seek(donated as C-H&Seek) protocols, who consider the user travel in the Euclidean space. In the other two protocols, the C-H&Hash provides the hashed location, which is the same as PPQR, while the C-Hide&Seek provides the encrypted location. For comparison purpose, in each of the protocol, we implement triangle_based granularities, whose projection on a two-dimensional plane are uniform triangle (see Figure 5).

In the experiments, we randomly generate a normally distributed 3D smooth surface whose projection total size is 215km^2, and the average density is 465users/km^2. We use a networked generator to randomly generate about 100,000 mobile users, and some of them are randomly selected to issue queries. The system runs about 4 hours for the users, and the locations are sampled every 2 minutes. Moreover, all users share the same proximity threshed (see Table 2) and don't move offline during the test. For simplicity, the granularity size is measured by two-dimensional projection triangulation. The parameter settings are shown on the Table 2.

Table 2. Parameter Settings

Parameters	Values
proximity threshed	200m, **400m**, 800m, 1600m
Area of a cell	$\frac{1}{2}100^2\text{m}^2$, $\frac{1}{2}\mathbf{200^2m^2}$, $\frac{1}{2}400^2\text{m}^2$, $\frac{1}{2}800^2\text{m}^2$
Number of buddies (on-line)	10,25,**50**,75,100

1) **Service precision.** This measures the exactness of the outcomes returned by each method, that is the ratio between the number of correct "in proximity" answers over the total number of outcomes. This measures the probability that a buddy reported "in proximity" is actually in proximity. Figure 5 shows that the precise with respect to varying the granularity size and the time interval respectively. Figure 5 shows that our method has better precision. This is due to the fact that our method use the geodesic distance, which is better precision than the Euclidean distance on 3D surface. Moreover, our method uses the user's real-time location, while C-H&Seek uses the location reported during the previous update interval. So, our method is not affected by the time interval.

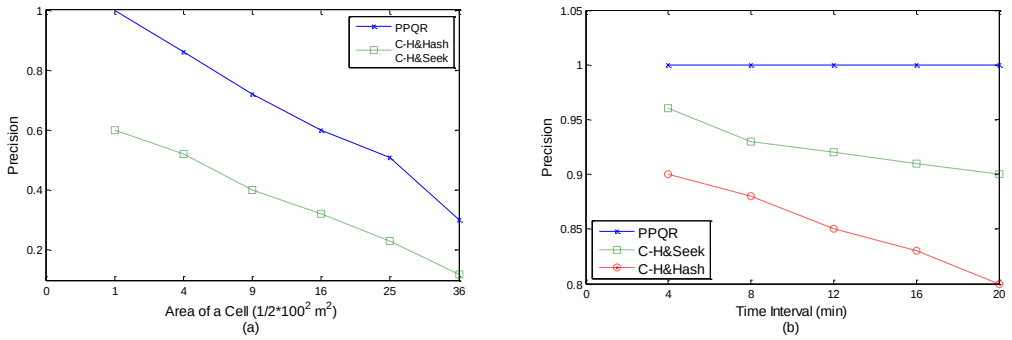


Figure 5. Quality of Service

2) Cost awareness. This evaluates the extra computation and communication costs introduced in these protocols. Computational costs on proximity services include: 1) the spatial generalization, 2) the hash or encryption operation, and 3) the judgment for proximity. The PPQR has the same computational costs with C-H&Hash, has more computational costs than the C-H&Seek because of the commutative encryption. Moreover, it is needed to judge each of buddies, so the computational costs grow linearly in the number of buddies in three protocols.

In Figure 6(a) we evaluate the average communication costs in the system for each proximity request. It is easily seen that the communication costs grow linearly with the number of buddies for the three protocols, and PPQR have slight larger than C-H&Hash and C-H&Seek. The reason is that, in each proximity request, only the request and the response messages are exchanged between the issuer and the SP in C-H&Hash and C-H&Seek, while each buddy and the SP can also exchange messages in PPQR. Figure 6(b) shows the average total communication costs of proximity services per hour. It is supposed that an update interval of 4 minutes using C-H&Hash and C-H&Seek, and 10 minutes as the average frequency of proximity requests. Since each location update cost less than 300 bytes, the total hourly cost for this sub-protocol is about 4KB. It is due to the fact that our protocol does not require the location update sub-protocol, so PPQR

incurs lower communication costs than C-H&Hash. But the costs of location update are small with respect to the communication cost of the proximity requests, so C-H&Seek incurs in lowest communication costs.

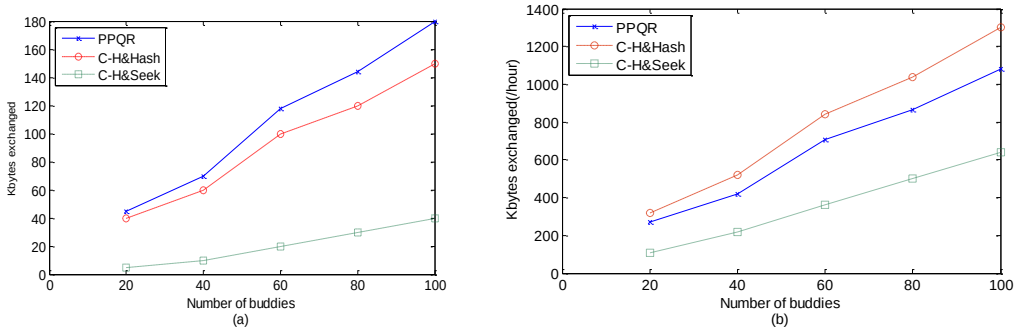


Figure 6. Cost Awareness

3) **Privacy level.** This evaluates how much additional privacy by each method in terms of the size of the generalization region, which is the privacy requirement of each user. As mentioned in Section 5.3, our method can guarantee complete privacy with respect to the SP and the buddies. Figure 7 shows our method and C-H&Hash can provide larger privacy than the privacy requirement, and C-H&Seek can provide the same privacy with the privacy requirement. This means that the privacy provided by our method and C-H&Hash is the uncertainty region.

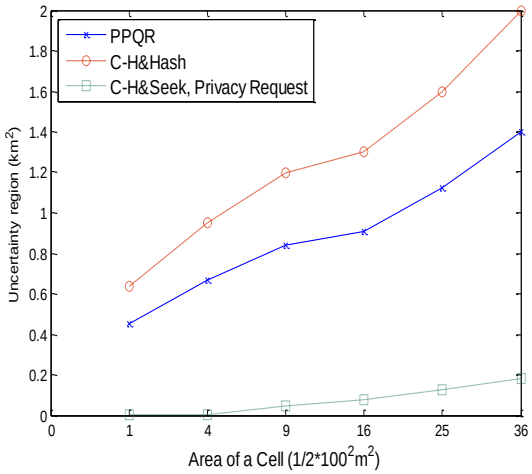


Figure 7. Privacy Protect Level

7. Conclusion

In this paper, we propose a privacy-preserving query method, which successfully provides more precise proximity services and solves the location privacy issues on 3D smooth surface. In this paper, we introduce the geodesic distance and put forward a more precise proximity range measurement method based on the triangle fractal. Based on this, a privacy-preserving proximity query protocol is proposed to provide complete privacy with respect to the SP and the buddies. The experiments indicate that our

method can combine high level of privacy-preserving with high quality of services, meanwhile the trade-off between privacy and service is well realized.

Acknowledgements

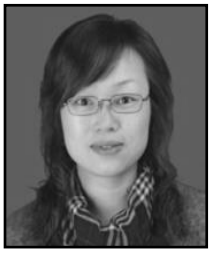
This work was supported by Key Project on the Integration of Industry, Education and Research of Guangdong Province (Grant No. 2012B091000054), and Sichuan Department of Education (Grant No.13ZB0152). This work was also partly supported by the National Nature Science Foundation of China under Grant (No.61373163).

References

[1] P. Samarati, "Protecting Respondents Identities in Microdata Release", IEEE Transanction on Knowledge and Data Engineering, vol. 13, **(2001)**, pp. 1010.
[2] L. Sweeney, "K-anonymity: A model for Protecting Privacy", International Journal of Uncertainty Fuzziness and Knowledge Based Systems, vol. 10, **(2002)**, pp. 557.
[3] A. Machanavajjhala, D. Kifer and J. Gehrke, "L-diversity: Privacy Beyond k-Anonymity", ACM Transactions on Knowledge Discovery from Data (TKDD), vol. 1, **(2007)**, p. 1.
[4] B. Gedik and L. Liu, "Location Privacy in Mobile Systems: A Personalized Anonymization Model", Proceedings of 25th IEEE International Conference on Distributed Computing System, **(2005)**, Columbus, OH.
[5] M. Gruteser and D. Grunwald, "Anonymous Usage of Location-Based Services through Spatial and Temporal Cloaking", Proceedings of 1st International Conference on Mobile System, **(2003)**; New York, pp. 31-42.
[6] F. M. Mohamed, C. Y. Chow and G. A. Walid, "The New Casper: Query Processing for Location Services without Compromising Privacy, Proceedings of 32nd International Conference on Very Large Data Bases, **(2006)**; ACM Press, pp.763-774.
[7] T. Xu and Y. Cai, "Location Anonymity in Continuous Location-based Services", Proceedings of 15th ACM Symposium on Advances in Geographic Information Systems, **(2007)**; New York.
[8] T. Xu and Y. Cai, "Exploring Historical Location Data for Anonymity Preservation in Location- based Services", Proceedings of IEEE INFOCOM 27th International Conference of the Computer and Communications Societies, **(2008)**; Phoenix, AZ, pp.547–555.
[9] B. Bamba, L. Liu and P. Pesti, "Supporting Anonymous Location Queries in Mobile Environments with PrivacyGrid, Proceedings of the 17th International Conference on World Wide Web, **(2008)**; New York, pp. 237–246.
[10] P. Kalnis, G. Ghinita and K. Mouratidis, "Preventing Location-based Identity Inference in Anonymous Spatial Queries", Proceedings of IEEE Transactions on Knowledge and Data Engineering, **(2007)**; pp.1719–1733.
[11] H. Kido, Y. Yanagisawa and T. Satoh, "An Anonymous Communication Technique Using Dummies for Location-based Services", Proceedings of IEEE International Conference on Pervasive Services, **(2005)**; ICPS.
[12] A. Khoshgozaran and C. Shahabi, "Blind Evaluation of Nearest Neighbor Queries Using Space Transformation to Preserve Location Privacy", Proceedings of the International Symposium on Spatial and Temporal Databases, **(2007)**.
[13] S. Mascetti, C. Bettini and D. Freni, "Privacy-Aware Proximity Based Services", Proceedings of the International Conference on Mobile Data Management, **(2009)**; Taipei, pp.31–40.
[14] S. Mascetti, D. Freni and C. Bettini, "Privacy in Geo-social Networks: Proximity Notication with Untrusted Service Providers and Curious Buddies", The VLDB Journal, vol. 20, **(2011)**, p. 541.
[15] X. Y. Li and T. Jung, "Search Me If You Can: Privacy-preserving Location Query Service", Proceedings of IEEE INFOCOM, **(2013)**; p.2760.
[16] B. Delaunay, "Sur La Sphere Vide", Izv. Akad. Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk, vol. 7, **(1934)**, p. 793.
[17] R. J.Wang, Z. G. Qin and H. l. Bao, "Triangulation-based Localization Algorithm Over Complex 3D Terrains", Application Research of Computers (in Chinese), vol. 30, **(2013)**, p. 2823.
[18] R. J.Wang, H. l. Bao and Z. G. Qin, "3D-CCD: A Novel 3D Localization Algorithm Based on Concave/Convex Decomposition and Layering Scheme in WSN", Ad Hoc & Sensor Wireless Networks (SCI Journal, accepted).

[19] J. Tang and F.Y. Zhang, "Computing Geodesic Distance on a Triangle Mesh", Proceeding of the 6th computer Graphics of China (in Chinese), **(2006)**; Hangzhou.
[20] G. Zhong, I. Goldberg and U. Hengartner, "Louis, Lester and Pierre: Three Protocols for Location Privacy", In Privacy Enhancing Technologies, Volume LNCS 4776, **(2007)**, pp. 62-76.

Authors



Min Li received the M.S. degree from the University of Electronic Science and Technology of China (UESTC). Currently she is working toward the Ph.D. degree in computer science at UESTC. Her main research interests include the network security protocol, privacy protection, specifically the location privacy in LBS.

Ruijin Wang received the M.S. and Ph.D. degree from the University of Electronic Science and Technology of China (UESTC). Currently he is teacher at the School of Computer Science and Engineering at UESTC. His main research interests include wireless ad-hoc sensor networks, pervasive computing, and mobile cloud computing. He is a student member of the ACM.

Zhiguang Qin received Ph.D degree from the University of Electronic Science & Technology of China. Now he is a professor, dean of Computer Science & Engineering Department, director of Computer Application Key Lab in Sichuan Province, member of IEEE. Currently his main research interests concern is the security of the networks.

Cong Wang received the B.S. and M.S. degrees from Southwest University of China, Chong-qing, China. Currently he is pursuing the Ph.D. degree in computer science at the University of Electronic Science & Technology of China. His main research interests are the applications of machine learning techniques to computer networking problems, specifically the prediction of latency in large-scale networks.

