PAPERS: Private and Precise Range Search for Location Based Services

Di Chen*, Peng Zhang*, Chengchen Hu†*, Huanzhao Wang*, Shun Wu‡, and Ningzhe Xing‡
* Department of Computer Science and Technology, Xi'an Jiaotong University, Xi'an, China
† Department of Telemactics, Norwegian University of Science and Technology
‡ State Grid Corp. of China

***Abstract*—Location Based Service (LBS) is gaining popularity on smart phones. One fundamental LBS is *range search*, which returns all Point of Interests (POIs) within a user-specified range. However, people also leave their location privacy at risks when using LBS like range search. How can a user invoke such service without revealing his location is an interesting, yet challenging problem to solve.**

Most existing approaches blur a user's location into a cloaked region, so that LBS cannot figure out the exact location of the requesting user. However, this would make the returning results inaccurate, containing some out-of-range POIs. To this end, we propose PAPERS, a new method to provide location privacy for users of range search. PAPERS leverage homomorphic encryption to let the user encrypt her location, and the LBS server can compute distances on ciphertext. In this way, the returning results by LBS are exactly the POIs within the specified range, while LBS learns nothing about user's real location. We implement a prototype of PAPERS, and evaluate it with real POI set of a large-scale production LBS. Experimental results show that PAPERS can achieve the goal of privacy protection, with reasonable overhead in response time and communication cost.

I. Introduction

With the widespread usage of smart phones, Location Based Service (LBS) is gaining more and more popularity. One fundamental class of LBS is range search, which serves as a building block for many other LBS services (e.g., route planning, friends finding, etc.). In range search, a user sends her location and the range to the LBS server, which would return all Point of Interests (POIs) within the range to the user. In this process, users have to reveal their locations to LBS servers, which later may be used inappropriately.

The risk of location privacy breach when using LBS has been widely recognized. The continued accumulation of query locations of a specific user may disclose her health condition, political view, religious belief, etc. Consider for example, Alice searches for nearby restaurants at 11 am, shopping malls at 6 pm, and night bar at 11 pm. An attacker can infer a lot of private information of the users, and can use locations in these queries to re-identify that the user is Alice.

This paper is supported by the 863 Plan (2013AA013501), the National Science and Technology Major Project (2013ZX03002003-004), the National Natural Science Foundation of China (61402357, 61403301, 61272459, 61221063, 61170245), the Program for New Century Excellent Talents in University, the Fundamental Research Funds for the Central Universities, the Jiangsu Future Internet Innovation Project (BY2013095-1-12), and the CETC 54 project (ITD-U14001/KX142600008).

Most existing approaches use spatial cloaking to hide the real location of a user [1]–[4]. In these approaches, an anonymization server transforms the location of a user to a cloaked region, and sends this region to the LBS server. Then, the LBS server returns all POIs within the proximity of the cloaked region. However, as the LBS does not know the exact location of the user, it cannot return the exact POIs within say r meters from the user.

In this paper, we aim to design a location privacy preserving method for range search with the following objectives: (1) Ensure high precision of the queried results; (2) Provide provable location privacy protection. With the goals above, we propose PAPERS, i.e., Private And PrecisE Range Search, for LBS. PAPERS leverages the *spatial cloaking* technique used in previous methods, and complements it with *homomorphic encryption* for improving precision. Specifically, there are two protocols in PAPERS. In Protocol 1, we use Voronoi [5] diagram to help quickly filter out all candidate POIs, given the user-specified cloaking region. Then all these POIs are returned to the user, who would then evaluate the distances of these POIs to her real location locally. In Protocol 2, the user would send both the cloaking region and her location encrypted with homomorphic encryption to the LBS server. Then, the LBS server can filter out candidate POIs just like in Protocol 1, and then calculate the distances of these POIs from user's location by computing on the ciphertext. In this way, the user does not need to calculate the distances herself.

In sum, PAPERS has the following major advantages over existing methods based solely on spatial cloaking:

- *High precision of returned results.* As the LBS server calculates the distances from candidate POIs from user's location, the user can filter out POIs for exact results, without distance calculation.
- *Location privacy protection.* LBS operates on the ciphertext of user's location when calculating distances, and thus our scheme is provable secure given the security assumption of the homomorphic encryption primitive.

We make the following contributions in this paper:

- We propose PAPERS, a private and precise range search method in LBS. It outperforms other schemes in terms of precision of returned results, with reasonable computation cost and communication overhead.
- We implement a prototype of PAPERS, and evaluate it

978-1-4673-6432-4/15/$31.00 ©2015 IEEE

TABLE I
NOTATIONS

Symbol	Implication
U	LBS User
S	LBS server
Q	User's location while querying
QR	Cloak region
$V-cell$	Voronoi cell
$V_{(x,y)}$	Coordinate of the center of Voronoi cell (POI)
P	A single POI point
CD_POI	Set of candidate POIs

using real POI dataset to evaluate its performance.

The rest of the paper is organized as follows: Section II defines the problem we are addressing. Section III introduces some technical background on Voronoi diagram and homomorphic encryption. Section IV and Section V present the design of two protocols of PAPERS in details, Section VI analyzes the privacy provided by PAPERS, Section VII evaluates PAPERS with experiments. Section VIII surveys related work and Section IX concludes this paper.

II. Problem Statement

Consider there is a user U, who can be using a smart phone, tablet PC, etc. U is located at a place that is specified with the latitude and longitude $Q = (lat, lon)$. Consider there is a LBS server S that is offering range query services, and U queries S with Q_u. Given Q_u contains information about U's location, and a range parameter r. S then sends all Point of Interests (POIs) that located less than r from U's location.

This paper is aimed at solving the following problems. How can we design a protocol so that U can make range search queries without letting S knowing her location? Also, we should guarantee that the results returned by S are precise. That is, all POIs returned by S are strictly within the circle with radius r centered at U's location.

III. Preliminary

A. Voronoi diagram

Voronoi diagram was first proposed by Holland climate scientist A.H.Thiessen, in order to calculate the average rainfall from discrete distributed meteorological stations. We compute the Voronoi diagram [5] for all POIs inspired by [6]. As shown in Fig. 1, each POI p_i is assigned to its Voronoi cell, by definition, p_i is the NN of any point within that cell. We superimpose a regular grid of arbitrary granularity on top of the Voronoi diagram. Each grid cell stores information about the Voronoi cells intersecting it. Take Fig. 1 as an example, the right one is a Voronoi diagram superimposed on grids, and the left one records the V-cells that are within or intersect with a given grid. Now, suppose user U is within grid $C2$, by looking up in the table to the left of the diagram, we can easily find that $p3$ and $p4$ are the closest neighbours of $C2$.

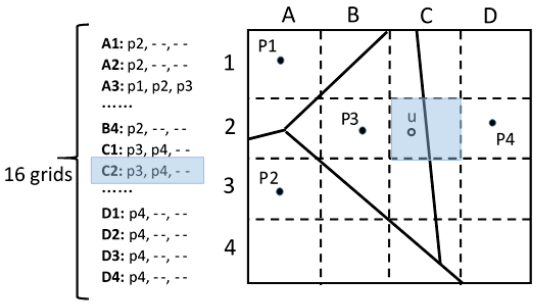


Fig. 1. An example of superimposing Voronoi diagram with grids.

B. Paillier homomorphic encryption

Homomorphic Encryption (HE) allows direct addition and multiplication on ciphertexts while preserving decryptability. We choose Paillier's system [7] to provide homomorphic encryption of user's location which is simple and efficient. Pailliers cryptosystem is composed of three algorithms KeyGenerate, Encrypt and Decrypt. The Paillier's cryptosystem satisfies the following homomorphic properties:

$$E(m_1, r_1) \cdot E(m_2, r_2) = E(m_1 + m_2, r_1 r_2) \, mod \, n^2 \quad (1)$$

$$E(m_1, r_1)^{m_2} = E(m_1 \cdot m_2, r_1^{m_2}) \, mod \, n^2 \quad (2)$$

Note that the random number r does not contribute to decryption or other homomorphic operation. For the sake of simplicity, we use $E(m)$ instead of $E(m, r)$ in the remaining paper.

Before the implementation of PAPERS, we complete the pre-computing as below:

1. For a given POI set, LBS server generates corresponding Voronoi diagram. The Voronoi diagram information is stored in our dataset, each Voronoi cell corresponds to one POI point.

2. For protocol 2 (with encryption), encryption key $EK_u = (n, g)$ and decryption key $DK_u = (\lambda, \mu)$ are generated for Paillier's cryptosystem on the client side. We assume the length of the key is 128 bits. The corresponding public key EK_s and private key DK_s are generated and assigned to the server by a certain public key cipher system.

IV. Protocol 1

In this section, we introduce Protocol 1, a baseline protocol as a building block for Protocol 2. The basic idea is that when a user needs to query for all POIs within a circular region r, she replace her location (Q) with a cloak region (QR). On receiving QR and r, the server runs an algorithm to filter out all POIs that are *possible* to have a distance less than r to Q. We term the set of all such POIs as *candidate POIs*. For each candidate POI, there exists at least one point p in QR, satisfying the distance of p and the candidate POI is less than r.

Before introducing the algorithm to find candidate POIs, we first need to define the distance from a POI to a cloak region QR.

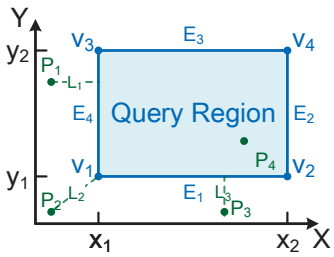


Fig. 2. Distance from a POI to a cloaking region.

A. Distance to Cloaking Region

Definition 1: The distance from a point $P(P_x, P_y)$ to a cloak region QR, denoted as $dist(P, QR)$, is defined as follows.

$$dist(P, QR) = \sqrt{(P_x - QR_x)^2 + (P_y - QR_y)^2}$$

Among them,

$$QR_x = \begin{cases} x_1 & P_x < x_1 \\ x_2 & P_x > x_2 \\ P_x & Other\ circumstances \end{cases}$$

$$QR_y = \begin{cases} y_1 & P_y < y_1 \\ y_2 & P_y > y_2 \\ P_y & Other\ circumstances \end{cases}$$

Fig. 2 illustrates this definition. Here, $QR = \{E_1 : (v_1, v_2), E_2 : (v_3, v_4), E_3 : (v_3, v_4), E_4 : (v_1, v_3)\}$. According to Definition 1, the distances from four points P_1, P_2, P_3 and P_4 to cloak region QR are L_1, L_2, L_3 and 0, respectively.

B. Privacy-Preserving Candidate POI Filtering

The algorithm to filter out candidate POIs while preserving location privacy is shown as Algorithm 1. Given the cloak region and range specified by the user, the algorithm returns all candidate POIs. As seen in Algorithm 1, we use a queue P to store the candidate POIs (Line 1). First, we select all V-cells that intersect with QR. These V-cells correspond to POIs that are in the vicinity of QR according to the property of Voronoi diagram. Among these V-cells, we select those satisfying $dist(v_{(x,y)}, QR) \leq r$ (Line 2-6). Then, we do a breadth-first-search to find neighboring V-cells of those already in P, and check the condition $dist(v_{(x,y)}, QR) \leq r$. If the condition is satisfied, we push the corresponding POI into the end of queue P. This process continues until the queue P no longer grows (Line 7-17). Finally, P is returned as the result (Line 18).

To get a feel why the above algorithm is valid, let us connect each pair of adjacent POIs together to form a graph G. Then, G is a connected graph according to the properties of Voronoi diagram, Therefore every POI can be reached in graph G starting from the initial candidate POI set, and thus no candidate POI will be missed. We omit the proof here due to limited space.

Algorithm 1: `Candidate NN Search(`QR, r, V`)`

Input: QR: cloak region of user's location
Input: r: queried range
Input: V: the set of all V-cells (each representing a POI)

```
1  P ← ∅ {Queue to store outcomes};
2  foreach v in V do
3      if v intersects with QR and dist(v_(x,y), QR) ≤ r then
4          P ← P ∪ {v};
5      end
6  end
7  A ← The first item in P;
8  while A → next ≠ NULL do
9      v_i = A;
10     foreach v ∈ V adjacent to v_i do
11         if v ∉ P and dist(v_(x,y), QR) ≤ r then
12             P ← P ∪ {v};
13         end
14     end
15 end
16 return P;
```

Protocol 1

User u: Sends a rectangle cloaked region QR containing the query point Q
Server: Calculates CD_POI according to QR we assume there are m items in CD_POI CD_POI=$\{POI_1(x_1, y_1), \cdots, POI_i(x_i, y_i), \cdots, POI_m(x_m, y_m)\}$
Server: Sends CD_POI
User u: For every $POI_i \in CD\_POI$, calculates $dist(Q, POI_i), (i = 1, 2, 3, \cdots, m)$
User u: Selects all POI_i to satisfy $dist(Q, POI_i) < r$ as the results

C. Protocol 1

The message flow of Protocol 1 is as shown as Protocol 1.

Note that the distance calculation used in our protocol uses the latitude and longitude of POIs. The distance between $Q(x_Q, y_Q)$ and $P(x_P, y_P)$ is defined as follows.

$$dist(Q, P)^2 = 2R^2 - 2R^2 \cdot cos(y_Q) \cdot cos(y_P) \cdot cos(x_Q - x_P) - 2R^2 \cdot sin(y_Q) \cdots (y_P) \quad (3)$$

Here, R is the radius of the earth, longitude and latitude are represented by radians, east longitude and north latitude are positive, correspondingly, west longitude and south latitude is negative. This is different from most existing works, which simply calculate distance in Euclidean plane. Our motivation is that in real situation, the locations returned from GPS or other positioning techniques are all geographical locations represented with the latitude and longitude.

V. Protocol 2

Based on Protocol 1, we introduce Protocol 2 in this section. In Protocol 2, we use homomorphic encryption, so that the server can compute the distance from POIs to user locations.

Protocol 2

User u: Sends the rectangle cloaking region $E_s(QR)$ encrypted by EK_s
User u: Sends $E_u(2R^2)$, $E_u(-2R \cdot cos(x_Q)cos(y_Q))$, $E_u(-2R \cdot sin(x_Q) \cdot cos(y_Q))$ and $E_u(-2R \cdot sin(y_Q))$ encrypted by EK_u
Server: Decrypts to get QR by DK_s and computes CD_POI using Alg 1
Server: Calculates and sends $E_u(dist(Q,P)^2)$ using the private distant calculation method in Sec V-A for each $P \in CD\_POI$
User u: Computes $\{D_u(E_u(X_i)), D_u(E_u(Y_i)), D_u(E_u(dist(Q,POI_i)^2))\}$ using DK_u and finds the results to satisfy $dist(Q,P) < r$.

A. Homomorphic Encryption for Private Distance Calculation

First, we show how to enable the LBS server to calculate distance between a POI $P(x_P, y_P)$ and the query location $Q(x_Q, y_Q)$ without actually knowing Q. We use homomorphic encryption for distance calculation given geographic coordinate (the latitude and longitude) as follows.

1. U generates ciphertext $E_u(2R^2)$, $E_u(-2R \cdot cos(x_Q)cos(y_Q))$, $E_u(-2R \cdot sin(x_Q) \cdot cos(y_Q))$ and $E_u(-2R \cdot sin(y_Q))$, and sends them to S. Here $E_u(\cdot)$ means homomorphic encryption using U's key.

2. S, after receiving the ciphertexts, calculates the following homomorphic operations:

$$E_u(-2R \cdot cos(x_Q)cos(y_Q))^{R \cdot cos(x_P) \cdot cos(y_P)} = E_u(-2R^2 \cdot cos(y_Q) \cdot cos(y_P) \cdot cos(x_Q) \cdot cos(x_P)) \quad (4)$$

$$E_u(-2R \cdot sin(x_Q) \cdot cos(y_Q))^{R \cdot sin(x_P) \cdot cos(y_P)} = E_u(-2R^2 \cdot cos(y_Q) \cdot cos(y_P) \cdot sin(x_Q) \cdot sin(x_P)) \quad (5)$$

$$E_u(-2R \cdot sin(y_Q))^{R \cdot sin(y_P)} = E_u(-2R^2 \cdot sin(y_Q) \cdot sin(y_P)) \quad (6)$$

Then computes the following transformation and sends $E_u(dist(Q,P)^2$ to U:

$$E_u(2R^2) + (4) + (5) + (6) = (3) = E_u(dist(Q,P)^2)$$

3. U computes $D_u(E_u(dist(Q,P)^2)$ using DK_u to get $dist(Q,P)^2$.

B. Protocol 2

The workflow of Protocol 2 is as shown in Protocol 2.

VI. Privacy Analysis

In this section, we analyse privacy metric of PAPERS. Privacy protection degree of PAPERS is controlled by client side, and performs as a controllable variables in experiments.

Location privacy metrics can be measured by *disclosure risks* to the adversary. Disclosure risk represents the probability that an attacker may know about the user's location and other sensitive information according to the public information and other background knowledge. Typically, the more background knowledge about the public information, the greater the risk of disclosure. We use D to denote public information, D_k to denote disclosing D using background knowledges K,so $r(D,K)$ represents as $r(D,K) = P_r(D_k)$.

In the previous protocols, assuming the attacker has not obtained relevant background knowledges of user's location, the user only sends a cloak region instead of query point, which can be an arbitrary point to the server. In order to analyse the privacy degree, we divide the whole area into grid cells that can be sufficient to distinguish each query point, the area of each grid cell is S_0, the area of QR is S_{QR}. Intuitively, the possibility to acquire exact Q from QR by server is S_0/S_{QR}, that is,

$$Disclosure\,possibility:\ r(D,K) = P_r(D_k) = \frac{S_0}{S_{QR}}$$

Consequently, we can achieve custom settings for user's privacy requirements by controlling the size of QR, which varies as independent variables in our experiment.

Moreover, apart from privacy protection provided by cloak region, protocol 2 using homomorphic encryption can provide higher degree of privacy protection against ciphertext-only attack, since all information obtained by third-party attackers are ciphertext. Attackers can hardly acquire valid location information unless crack Paillier's system, that is to crack homomorphic encryption scheme and RSA scheme as well, which is hard to achieve.

VII. Experiments

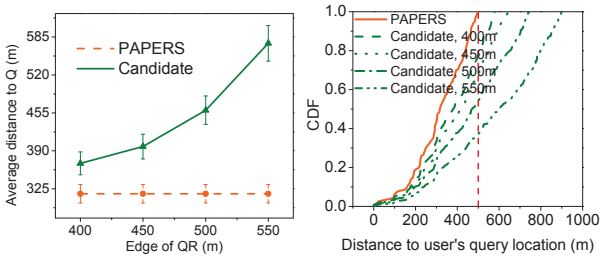
In this section, we evaluate PAPERS with experimental results, based on real data from a large-scale production LBS. The metrics we considered include result precision, response time and communication overhead.

A. Setup

In the experiment, we have two physical hosts, one as the client and one as the server. The server is configured with Intel i7 CPU (3.40GHz) and 16GB DDR3 memory, running Windows 8 64bit OS. The client is configured with Intel i5 CPU (3.1GHz) and 4GB DDR3 memory, also running Windows 8 64bit OS. The length of Paillier's public key is 64 bits; Our test data set was collected from a large-scale production LBS (from May to August 2013, containing 3,211,157 entries from more than 2,569,245 users).

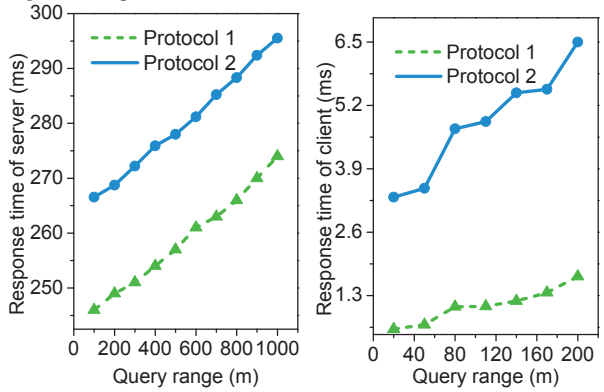
B. Precision of results

The standard to measure precision is whether the returned POI set contains all POIs within the query range. Spatial cloaking method [1]–[4] uses cloaking region as input to request LBS to get approximate nearest neighbors (approximate NN for short), which cannot guarantee the accuracy of the results. The size of cloaking region influences the returned approximate nearest results to a large extent. In PAPERS, candidate POIs are generated only according to cloak region by the server, just similar with other approximate NN method,



(a) Average distance to query location (b) CDF of distance to query location

Fig. 3. Comparison between PAPER results and candidate results



(a) Server (b) Client

Fig. 4. The server- and client-side response time, variable query range, $QR = 1600m^2$.

so we regard candidate POI results as reference object for the final results.

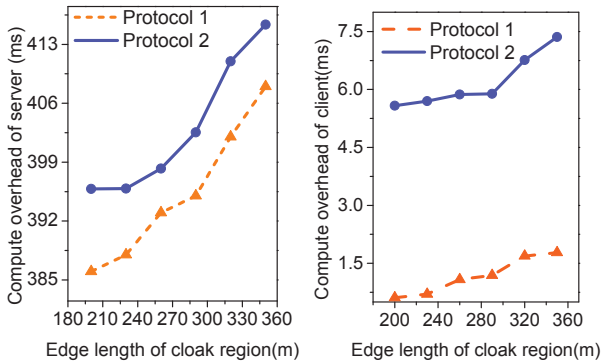
We set query range as $500m$ and vary the edge of cloak region QR from $400m$ to $550m$, analyse cumulated distribution and calculate the average of distance between user's query point Q and each POI in the results. As shown in Fig. 3, the average distances between Q and each POI in PAPERS results remain $316m$ as the edge of QR grows according to Fig. 3(a), all of results are within $500m$ query range according to Fig. 3(b). In Fig. 3(a), average distances between Q and each POI in candidate results increases from $368m$ to $574m$ as the edge of QR increases from $400m$ to $550m$, and distributed broader as shown in Fig. 3(b).

The result generated according to cloak region always contains redundant POIs outside the query range, and the scope of results becomes broader as the cloak region grows. However, The change of cloak region cannot influence the results of PAPERS, since the result set are selected as all POIs within query range in PAPERS, which does not contain any omissions and outliers.

C. Response Time

We evaluate the response time of 2 protocols for PAPERS in this subsection, and study the correlation between computational overhead and two adjustable variables, the size of cloak region and query range.

The response time for different query ranges is shown in Fig. 4, where we fix the area of rectangle QR as $1600m^2$, and vary the query range from $100\,m$ to $1000\,m$. Fig. 4(a) shows



(a) Server (b) Client

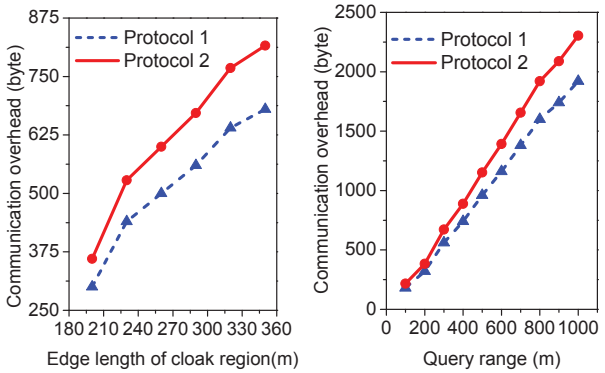
Fig. 5. The server- and client-side processing time, variable cloaking region, $r = 500m$.

the comparison of response time on the server side between protocol 1 and protocol 2. As the radius of query range increases from $100\,m$ to $1000\,m$, response time of protocol 1 increases from $246\,ms$ to $274\,ms$ and response time of protocol 2 increases from $266.55\,ms$ to $295.52\,ms$. On the client side, as shown in Fig. 4(b), computational overhead of both protocols is less than $10\,ms$, and also follows the similar positive correlation between query range and response time. Since more POIs need to be calculated and searched as the size of query range becomes larger.

The response time of different cloak regions QR is shown in Fig. 5, where we fix the query range as $500\,m$, and vary the edge length of QR from $200\,m$ to $360\,m$. Fig. 5(a) shows the comparison of response time on the server side between protocol 1 and protocol 2. As the edge length of QR increases from $200\,m$ to $360\,m$, the response time of protocol 1 increases from $380\,ms$ to $400\,ms$ and the response time of protocol 2 increases from $400.82\,ms$ to $419.35\,ms$. On the client side, as shown in Fig. 4(b), computational overhead of both protocols is less than $10\,ms$, and also follows the same positive correlation between query range and the response time with the server side. Since when QR becomes larger, more Voronoi cells need to retrieved in Algorithm 1, more POIs are appended into candidate POI set, and more distance calculation operations need to be handled consequently.

D. Communication Overhead

we evaluate the communication overhead incurred by our two protocols, in terms of bytes transmitted per query. The overall communication overhead in a query using protocol 1 consists of QR (transmitted from client side to server) and returned CD_POI (transmitted from server to client side). In protocol 2, two parts compose the communication overhead per query, one is ciphertext $E_u(2R^2)$, $E_u(-2R \cdot cos(x_Q)cos(y_Q))$, $E_u(-2R \cdot sin(x_Q) \cdot cos(y_Q))$ and $E_u(-2R \cdot sin(y_Q))$ (transmitted from client side to server), another is $E_u(dist(Q,P)^2)$ and $E_u(POI\_info)$ for each $POI \in CD\_POI$ (transmitted from server to client side). First, we fix the queried range as 500 m, and vary the size of rectangle cloaking region (in terms of its edge length). As shown in Fig. 6(a), for both protocols, the communication overhead



(a) Communication overhead, variable cloaking region size, $r = 500m$

(b) Communication overhead, variable query range, $QR = 1600m^2$

Fig. 6. Communication overhead of baseline and enhanced protocol.

grows as the size of cloaking region increases. Since the distance calculation is handed over to the server in protocol 2 using homomorphic encryption method, ciphertext for distance calculation and distance results are added during transmission, thus communication overhead of Protocol 2 is slightly higher than Protocol 1. Then, we fix the rectangle cloaking region as 1600 m^2 (40 $m \times 40$ m), and vary the length of queried range. Fig. 6(b) shows that both protocols have a larger communication overhead when the length of queried range increases, and still Protocol 2 has a larger bandwidth.

Summary: From the experimental results above, we conclude that (1) The results contain all POIs within query range, without any outlier and omission. The computation overhead of the two protocols are in millisecond level on a commodity server, and their communication overhead are around 2 KB, for queried range 500 m and cloaking size 1600 m^2 (40 $m \times 40$ m). (2) The overhead of both protocols increases for larger queried range and cloaking size, echoing a natural tradeoff between performance and privacy. (3) Since Protocol 2 has a relatively higher overhead, but preferable than Protocol 1 considering it provides a better privacy.

VIII. Related Work

People's demand for LBS services raises increasing concern about location privacy. A number of techniques are proposed for *NN search privacy protection*. Existing approaches can be roughly grouped into the following three classes.

A. Spatial and temporal cloaking

The cloaking approach was aimed to provide *location k-anonymity* [2], a variant of classic k-anonymity [8]. Spatial and temporal cloaking method [2]–[4] ensures each location based query is confused within other queries using k-anonymity concept with the help of a third trusted party (TTP). However, TTP will introduce new privacy holes, and a large number of users must subscribe to the service in order for the construction of cloak regions.

B. Space transformation

Space transformation method [6], [9] converts the location information into another space representation, which has a certain transformation relationship with the original one. The typical representative of this category is Hilbert space-filling curve space method which transforms the two-dimension POI information to a one-dimension Hilbert space-filling curve space [9]. Literature [6] uses Moore curve and applies secret circular shift for k nearest neighbor search on the encrypted database. However, the Hilbert and Moore space-filling curve methods cannot return the exact nearest neighbors and the result may generate serious deviation in some cases.

C. Cryptographic transformation

Cryptographic transformation [6], [10] methods provide privacy preserving by encryption. Users queries are transformed to the operations on ciphertext, and the LBS server homomorphicly performs the query on ciphertext without knowing the plaintext included in original query to get the encrypted query result. The user decrypts the result to get the actual result. But the scheme needs client to take part in the private homomorphic distance computing and cannot support the multiple neighbors query.

IX. Conclusions

In this paper, we designed and evaluated PAPERS, a new scheme for private range search. PAPERS has two nice features including (1) high precision of returned results, and (2) privacy protection by combining cloaking and cryptography. We implemented a prototype for PAPERS, and evaluate it with real user queries. Experimental results showed that PAPERS can provide both privacy protection and result accuracy, with reasonable computation and communication overhead.

References

[1] M. F. Mokbel, C.-Y. Chow, and W. G. Aref, "The new casper: query processing for location services without compromising privacy," in *VLDB*, 2006.

[2] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking," in *MobiSys*, 2003.

[3] M. L. Yiu, C. S. Jensen, X. Huang, and H. Lu, "Spacetwist: Managing the trade-offs among location privacy, query performance, and query accuracy in mobile services," in *ICDE*, 2008.

[4] B. Bamba, L. Liu, P. Pesti, and T. Wang, "Supporting anonymous location queries in mobile environments with privacygrid," in *WWW*, 2008.

[5] M. De Berg, M. Van Kreveld, M. Overmars, and O. C. Schwarzkopf, *Computational geometry*, 2000.

[6] G. Ghinita, P. Kalnis, A. Khoshgozaran, C. Shahabi, and K.-L. Tan, "Private queries in location based services: anonymizers are not necessary," in *International Conference on Management of Data*, 2008.

[7] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Advances in Cryptology EUROCRYPT 99*, ser. Lecture Notes in Computer Science, J. Stern, Ed. Springer Berlin Heidelberg, 1999, vol. 1592, pp. 223–238.

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

[9] A. Khoshgozaran and C. Shahabi, "Blind evaluation of nearest neighbor queries using space transformation to preserve location privacy," in *Advances in Spatial and Temporal Databases*, 2007.

[10] G. Ghinita, "Private queries and trajectory anonymization: a dual perspective on location privacy," 2009.