Preventing Multi-query Attack in Location-based Services

Nilothpal Talukder
Dept. of Computer Science
Purdue University
West Lafayette, IN, USA
ntalukde@cs.purdue.edu

Sheikh Iqbal Ahamed
Dept. of MSCS
Marquette University
Milwaukee, WI, USA
iq@mscs.mu.edu

ABSTRACT

Despite increasing popularity, Location-based Services (LBS) (e.g., searching nearby points-of-interest on map) on mobile handheld devices have been subject to major privacy concerns for users. The existing third-party privacy protection methods hide the exact location of users from service providers by sending cloaking regions (CR) that contain several other user locations in the vicinity. However, this has not ensured LBS full immunity from the privacy concerns. In this paper, we describe a serious privacy problem of LBS called multi-query attack. In this attack, the exact location of the service requester can be inferred by the adversary through obtaining cloaking regions that are shrunk or extended in subsequent queries. This problem can be addressed by judiciously retaining, over a period of time, the cloaking regions for the same set of users. Most methods in the literature are weakened for considering only a static snapshot of users during evaluation. Thus, any update due to user movements in real time becomes very costly. Our proposed approach, ANNC (Adaptive Nearest Neighborhood Cloaking) ,emphasizes developing disjoint sets of users dynamically over time in order to share the common CRs. The CRs are organized in balanced binary trees with restricted height. Thus ANNC achieves the balance between search efficiency and quality of cloaking with higher anonymity levels. The experimental evaluation demonstrates that ANNC will be more efficient in practice than other well-known approaches.

Categories and Subject Descriptors

C.2.0 [**Computer-Communication Networks**]: General – *Security and protection*

General Terms

Performance, Design, Security

Keywords

Location privacy, Adaptive Nearest Neighborhood Cloaking (ANNC), Reciprocity condition

WiSec'10, March 22–24, 2010, Hoboken, New Jersey, USA.

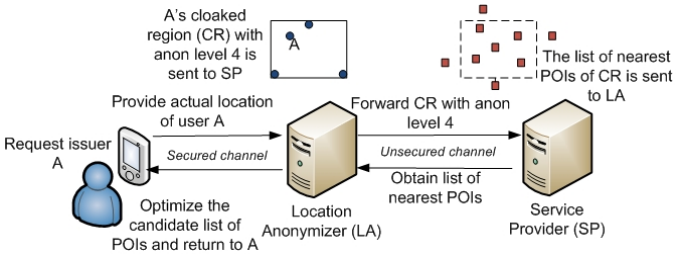

Figure 1: Spatial Query processing through Anonymizer

1. INTRODUCTION

According to projections from ABI Research Inc. [9] smart phones made up about 14% of all mobile devices shipped globally in 2008 and were expected to increase to more than 17% in 2009. Such proliferation of high-end touch-screen smart phones (iPhone, Google's G1 etc.) equipped with location-sensing capability has fueled the widespread popularity of Location-based services.

In this paper, we refer to LBS as internet services on handheld devices that use geo-locations of the users to provide search service, real-time information on locations such as weather or traffic report, or even share current location among friends and family members. iPhone social networking tools, BuddyFinder or search utilities such as EarthComber or urbanspoon are examples of such applications.

Dashboards of today's modern vehicular systems (Ford Motor Co. cars like Mercury and Lincoln) come pre-equipped with GPS navigation systems. Web map search utilities like Google Maps, with their enormous data centers and optimization techniques, are trying to provide very detailed information on points-of-interest (POI) in real time. So, spatial queries to LBS issued through mobile devices have become an indispensable part of people's day-to-day life.

Range query [7] and Nearest Neighbor (NN) query [11] are two common forms of spatial query. An example of Range query is: "Send me a list of gas stations and prices within a 10 miles radius", whereas, an NN query may look like: "Show me the nearest food court to the north of highway E-94".

It is evident that in any form of such query, the current location of the query requester has to be disclosed to the service provider and thus the query issuer is fraught with privacy risks.

An adversary, with the intent of obtaining the exact location of the user, if successful can perform the re-identification attack [2] by co-relating the most visited location with external knowledge such as public user profiles, WhitePages or any other form of publicly available phone and address directories. For example, frequent searches and visits to health-care service providers may disclose information about Alice's chronic illness which she didn't wish to share with her employer and in turn she

may be treated unfairly at her job. Likewise, Bob's search efforts for casino through LBS may reveal his gambling habit [7] and thus put his reputation in jeopardy. The inference attacks are also common on location traces [14], especially when the attacker obtains an individual's continuous spatial data in short intervals.

The privacy of the query issuer is preserved by hiding or obfuscating her exact location with a Cloaking Region (CR) [4, 5, 6] or Anonymous Spatial Region (ASR) [7] forwarded to the Service Provider (SP). Within a CR, along with the query issuer, there exists *k*-1 other users who have previously issued a query, or exist with the potential to do so. Most of the approaches [4-7] in the literature assign the task of CR generation to a trusted party known as the Location Anonymizer (LA). A brief architecture of the system is depicted at Figure 1.

As discussed, the *K*-anonymity [2] principle adopted in all the approaches [4-7] ensures that the CR chosen for a query offers the attacker a probability of re-identification not exceeding 1/*K*, *K* being the preferred anonymity level of a query issuer. In this paper, however, we argue that this is not always the case, especially when an adversary obtains several CRs in short intervals that include the query issuer inside different anonymity sets. We introduce this privacy concern of multi-query attacks in Section 2. This way, the exact location of the user can be nearly or completely compromised.

The peer-to-peer techniques found in the literature [8, 15, 13] eliminate the necessity of LA in order to generate CRs by considering mutual trusts among the entities [17]. Another recent Anonymizer-less technique proposed by Ghinita et al [16] is able to determine the approximate and exact nearest neighbors of the pre-shared points-of-interest (POI) using the private information retrieval. However, it requires the additional overhead of a list of POIs to be sent to the resource-constrained device before a query can be issued. Meanwhile, the choice of cryptographic protocol [10] is not yet very suitable for such an environment.

In summary, the problems with the existing approaches are:
a) They are subject to privacy attacks, because the same user is found in different CRs generated during short time spans.
b) Most approaches consider only a static snapshot and the actual location update cost for mobile users has been ignored.
c) The peer-to-peer techniques require a trust relationship be set up among the users before they can communicate.
d) Sending information on pre-cloaked regions over POIs or adopting cryptographic protocols exerts additional overhead on resource-constrained mobile devices.

The placement of the LA not only thwarts privacy violation attempts, but also removes additional overhead of filtering the candidate result set from resource-constrained mobile devices [5]. Although the spatial queries are mostly considered anonymous, the hardware address (e.g. MAC address) of the user is exposed during communication and thus, in reality, complete anonymity is not feasible. In that case, the LA at least provides safeguard against identity violation. The first communication can automatically subscribe the user into the LA's system and hence no additional operation is required.

Our proposed approach Adaptive Nearest Neighborhood cloaking (ANNC) involves the technique of generating CRs in such a way that the anonymity set used for a query are retained over a period of time. It tries to assign CRs on a 'first come, first served' basis through Nearest Neighbor search and retain them inside multiple binary tree structures for efficient look-up. However, a new user or a user that has moved out of the previously assigned CR may not be able to form a CR with its exact nearest neighbors. Thus, the new CR area may be constructed with degraded quality. By limiting the height of the trees and adopting CR area extension, ANNC tries to achieve a balance between query quality and privacy risk. Our approach and the contribution of our paper is as follows:
a) A novel cloaking approach is proposed that aims to prevent privacy attacks due to the shrinking and intersection of CRs.
b) An efficient data structure is presented to ensure the quality of the cloaking area and maintain reasonable update costs.
c) A detailed experimental evaluation of our approach demonstrates that it outperforms existing cloaking techniques in practice.

The rest of the paper is organized as: Section 2 presents the motivation of our approach, demonstrating the situations where existing approaches may fail. Section 3 lists the related works. Section 4 presents our approach and Section 5 provides brief mathematical analysis of it. Section 6 depicts the experimental findings. Finally, Section 7 presents our concluding remarks and future directions of this research.

2. MOTIVATION

In this section, we introduce the concept of a multi-query attack on an LBS user. However, we begin with the definition of reciprocity condition to better demonstrate the attack scenarios discussed later in this section. We consider an adaptation of the definition of reciprocity condition described in Kalnis et al's work [7] for multi-query attack.

Definition 1

The Reciprocity Condition: The original definition [7] states that a CR satisfies the reciprocity condition if a) it contains the requester and at least *k*-1 additional users and b) every user in the CR is also assigned the same CR for the given *k*.

Our adaptation for reciprocity condition necessitates that an LA retains the same CR for every user in an anonymity set of level k for the subsequent queries as long as the users remain inside that CR.

To better understand the situation, let us consider that the LA generates a *k*-CR for the user u_1 which also includes *k*-1 other users $\{u_2, \dots u_k\}$. The reciprocity condition enforces that if any other user apart from u_1, (i.e., a user having an index $1 < i \leq k$) issues any request with anonymity level k, the CR of the issuer must be the same as that of u_1.

Although not considered as a necessary condition for any of the cloaking techniques in the literature [4-7], the attacks demonstrated in this section emphasize the importance of this consideration.

The reciprocity condition has been adopted in Kalnis et al's work [7] by proposing a cloaking scheme called Hilbert Cloaking named after Hilbert space-filling curve [19]. Yet, the technique suffers from privacy attacks and degraded query quality which is discussed in the Related Works section.

2.1 Multi-query Attack

We define a multi-query attack as the one where an adversary tries to compromise the actual location of the query issuer with the help of a series of two or more spatial queries involving different cloaking regions. In this paper, the multi-query attack has been narrated in two different forms:
a) Shrink region attack; and b) Region intersection attack

Typically, the attack model on LBS considers the following fundamental knowledge assumptions [7] for an attacker, who may have access to the following information:

1. The method for determining cloaking regions
2. The dimensions for cloaking regions
3. The anonymity sets inside every CR -- this is a valid assumption, because in most cases the users are subscribed to the service provider and the queries appear as distinguishable.

2.1.1 Shrink Region Attack

As its name implies, the problem arises when two or more different subsequent queries cause shrinkage of a CR, while still representing the same anonymity sets, excluding just one user. As depicted in Kalnis et al's work [7], this problem is common for all the existing cloaking approaches such as Interval Cloak (IC) [4], New Casper [5], and Clique-Cloak [6]. The attack is illustrated through Figure 2(a)(b). It shows that a CR (denoted by dashed lines) covers six users , all having the equal probability of issuing a request. For example, a random user u_1 has issued a query with the anonymity level requirement k = 5. The CR denoted by a dashed line in the Figure satisfies the requirement. Yet, when other users with an index issue a query with the same anonymity level, the shrunk CR in Figure 2(b) is provided to the service provider. From the assumptions (knowledge of anonymity set), the attacker may conclude that the last request issuer is the user u_1 and hence his location is at the right corner of the CR. Subsequent queries issued by this set of users will confirm the guess by the attacker. This problem can only be addressed if the users in the same anonymity set would have used the same CR over time, i.e., a cloaking algorithm that conforms to the reciprocity condition.

2.1.2 Region Intersection Attack

Another common problem found in all the standard approaches is the region intersection attack. In this attack, the actual location of the user may be revealed through an intersection operation of anonymity sets from different queries that generate overlapping CRs. It can be best illustrated with Figure 2(c), in which there exists two CRs with anonymity level 3 denoted by anonymity sets and .

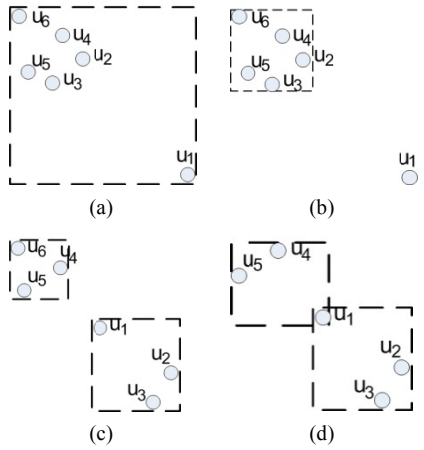

Figure 2. (a) (b) Shrink region attack (c) (d) Region intersection attack

When the user u_6 leaves, as shown in figure 2(d), the CRs are updated and a new CR consisting of the users also include u_1. Note that the user u_1 is in the intersection of the two CRs consisting of two sets of users and Thus, two subsequent requests from these two anonymity sets apparently reveal the actual location of the user u_1 to the attacker. The problem is, once again, not meeting the reciprocity requirement. To avoid this situation, the CRs should be defined for disjoint groups of user sets and, therefore, conform to the reciprocity requirements. ANNC conforms to the reciprocity condition and is immune from multi-query attack. Besides multi-query attack, another attack known as Center-of-CR attack [7] is discussed in the literature. The idea behind this attack is that if the CR is centered with the exact location of the query issuer, it will be readily compromised. Kanlis et. al's work [7] presents an interesting analysis on the probability of a person being found closest to the center of a CR generated by different cloaking techniques. Our proposed approach ANNC, being an adaptive version of randomized NNC, possesses inherent protection from this attack.

3. RELATED WORKS

As discussed earlier, the basic idea of cloaking is to render the location of the service requester or the query initiator indistinguishable from other k-1 users when forwarded to the SP, with k being the desired anonymity level of the requester.

The Spatial and Temporal Cloaking technique [4] or Interval Cloaking (IC) was the earliest to address the problem of location anonymization. This approach partitions the space into four equal squared regions or quadrants until all the users fit in separate quadrants. The data structure is known as quadtree and traversed top down until there are k users including the requester. The root of the subtree is then returned to the service provider by LA and the spatial region thus achieved is denoted as Cloaking Region (CR).

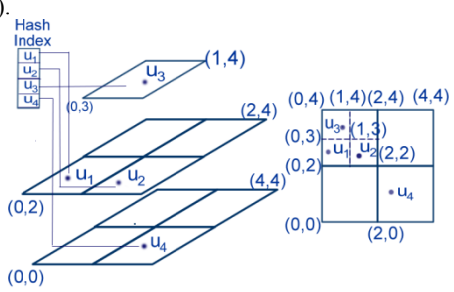

Figure 3. The spatial representation of quad-tree and New Casper

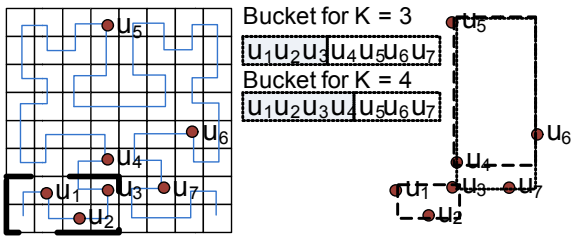

Figure 4. Hilbert Cloaking technique and NN Cloaking of the same users

The New Casper [6] follows the same notion of cloaking as IC's, where grids are organized in pyramid structure (as in Figure 3) to facilitate hierarchical traversal. However, the search for CR is performed in bottom-up fashion. New Casper also introduces query classification based on public and private data and demonstrates different requirements for anonymization. Casper tremendously reduces the CR area (almost half the size of IC's as found in the experimental results) and helps to gain efficiency in

candidate list filtering. Privacy Pyramid [18] provides an optimal choice between the top down and bottom up search to gain search efficiency. However, all the quadtree approaches [4, 5, 26] are expensive in terms of restructure cost and yet threatened by multi-query attack.

Nearest Neighborhood Cloaking (NNC) [7] focuses on minimizing the CR with a randomized neighbor selection approach. The number of users, including the requester, covered by the CR is equal to or greater than the anonymity level of the requester.

The Clique-Cloak [5] assigns Minimum Bounding Rectangle (MBR) for all users; if the users' MBR intersects, they are eligible to form a clique among themselves. The *k*-anonymity level of the request issuer is satisfied if he/she belongs to a *k*-cliqued region, otherwise the request issuance is suspended until the condition is satisfied. Thus the approach incorporates Temporal Cloaking along with Spatial Cloaking.

Exploiting the concept of the Hilbert space-filling curve [19], another cloaking technique has been devised by Kalnis et al [7,8] which guarantees spatial anonymity with reciprocity condition. The balanced binary tree indexed with Hilbert values (obtained through mapping of 2D points to 1D) helps to compute the start and end position of the anonymity level with almost no cost. Therefore, the users need not be physically stored in *K*-buckets. As an example, the left area of Figure 4, $\{u_1, u_2, u_3\}$ has Hilbert values {3, 5, 7} respectively. The CR for $\{u_1, u_2, u_3\}$ is shown by a dark dotted line. The buckets for k=3 and k=4 are shown in the Figure 4. An interesting observation is that consecutive queries with different anonymity levels will compromise the location of u_4. Let's consider u_2 issues a query with k=3 resulting in an anonymity set $\{u_1, u_2, u_3\}$, then u_6's query with k=3 results in an anonymity set $\{u_4, u_5, u_6\}$. However, when u_3 initiates a query with k=4, from the intersection of all the CRs the exact location of u_4 is compromised. Therefore, the reciprocity condition is maintained for only one anonymity level, in contrast to what the authors' claim. The only remedy to this problem is to provide multiples of buckets to preserve at least reciprocity for k=3. Again, the area of CR generated by Hilbert Cloaking may also degrade the quality of the query. It is also evident from Figure 4, where the CR regions generated by NNC by the anonymity sets $\{u_1, u_2, u_3\}$, $\{u_4, u_5, u_6\}$ are shown on the right. As in HC, u_7 has to share the area with $\{u_4, u_5, u_6\}$. The dashed rectangle on the right covering u_7 shows the new CR. However, in the case of HC, the CR will occupy more than half of the space shown left on the space-filling curve. The reason is the actual nearest neighbor of u_4 is u_7. But in the space filling curve, u_5 and u_6 come before u_7 to form a CR with anonymity level 3.

Hoh et al [20] provides an obfuscation technique for trajectory of continuous spatial data; this is acceptable when disclosing offline data and can not be applied to real time LBS. In Trusted Computing approach [21] the operator (also the LA) will only seek service and hand over actual location information by verifying the provider can not perform attacks.

4. OUR APPROACH

The discussions on the multi-query attack and quality issues in the previous sections help us to deduce the fundamental requirements for a cloaking algorithm intended to preserve privacy in spatial queries. A summary of the requirements are:

1. The CR should be able to satisfy the reciprocity condition
2. The LA should ensure as small a CR as possible for query efficiency
3. The cloaking approach should be a uniform one and must not assume anything about user distribution

Apparently the first two conditions together appear as the fundamental trade-off in designing a cloaking algorithm. Thus, none of the conditions can be enforced as necessities for the algorithm. The first condition, however, may become very vital to meet, especially at the lower anonymity levels when the CRs are small enough and subject to attack.

4.1 Adaptive Nearest Neighborhood Cloaking

Considering the requirements for designing an attack-resistant cloaking algorithm, we propose a novel cloaking scheme called Adaptive Nearest Neighborhood Cloaking (ANNC). The basic idea of ANNC is to develop disjoint sets of users dynamically with the requester's anonymity level. The disjoint sets are built with Nearest Neighbor (NN) search from the available neighbors (those who are not already included in other disjoint sets or CRs). The user anonymity sets are arranged in a binary tree (Figure 5(c)) and each node corresponds to the anonymity level of a CR (the user node itself has the anonymity level 1).

Definition 2

Cloaking Primitive (CP) level: This refers to the minimum anonymity level enforced by the system to conform to the reciprocity condition when disclosing CR to SP. For example, if the cloaking primitive level is k_{CP}, the system of n users will contain maximum $\lfloor n/k_{CP} \rfloor$ disjoint anonymity sets. Therefore, the following condition holds: $\bigcap_{i=1}^{\lfloor n/k \rfloor} f_i = \varphi\ (null)$, where f_i's are the anonymity sets with level k_{CP}.

Definition 3

Strict Reciprocity level (SRL): This refers to the maximum anonymity level of a disjoint set of users where the reciprocity condition is preserved. The roots of the binary trees formed by the disjoint anonymity sets are not allowed to grow beyond this point. The strict reciprocity anonymity level is referred to as k_{SRL}.

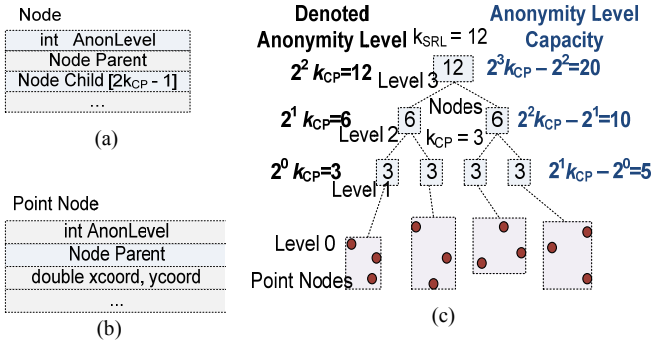

Figure 5. (a) internal Tree Node denoted simply as Node, (b) Point Node, (c) Denoted Level and Anonymity Level Capacity of Nodes inside the tree

4.1.1 Data structure

Node: The data structure Node is used for representing a mobile user (Point Node) as well as the composite disjoint set in the binary tree. As seen from Figure 5 (b, c), the user Node is represented as a leaf inside the tree and contains the coordinates in the euclidean space. There will be several such rooted trees in the system representing the hierarchical organization of the disjoint

sets of users. A very simplified structure presented here is a modified version of disjoint RTrees[12] intended for faster search and merging of CRs. The anonymity level of the root Node represents the maximum anonymity level that can be readily offered for a request of CR. Requests with higher anonymity levels require merging of CRs from other rooted trees, which is discussed in the ANNC CR Search section. As a special consideration, the Node that is the immediate parent of the leaf Nodes (Point Node) can contain up to $2k_{CP}$-1 children. That many children are allowed to accommodate new users who arrive inside the existing CRs of k_{CP} anonymity level.

4.1.2 Anonymity Level Capacity and Denoted Anonymity Level

The Anonymity Level Capacity denotes the maximum anonymity level allowed inside a Node. The denoted anonymity level of a Node residing in level i of a tree is $2^{i-1}k_{CP}$, $where\ 0 < i \leq \log_2(k_{SRL}/k_{CP})$. The maximum anonymity level of that Node will be $2^i k_{CP} - 2^{i-1}$. An example is shown in the Figure 5(c) with a tree of 4 levels (0-3), 0 being the leaf level containing Point Nodes (shown in dots). The Node at level 1 has the denoted anonymity level k_{CP} and the anonymity level capacity, $2k_{CP} - 1$. When this capacity is exceeded for a Node at level 1, the Node is split into two equal sized Nodes (with anonymity level k_{CP}) and the Node nearer to the sibling in the tree remains attached to the tree. This situation is depicted next.

4.1.3 Node Split

When a Node with anonymity level k_{CP} is populated with $2k_{CP}$ users, it is time to split the Node into two, each containing k_{CP} users. Of the two, the nearest Node remains attached to the tree. The other one is detached and forms a stand-alone tree. In the Figure 6 (a) the leftmost Node of the tree with k_{CP}=3 is populated with 6 users. From Figure 6 (b), we observe that the Node is split into two anonymity sets with the nearest neighbors. The new anonymity sets are $\{u_1,u_7,u_9\}$and$\{u_2,u_3,u_8\}$. Finally, the nearest CR of the anonymity set $\{u_4,u_5,u_6\}$ remains attached to the tree as a sibling. Additionally, the anonymity level of the parent Nodes along the way to the root are also updated.

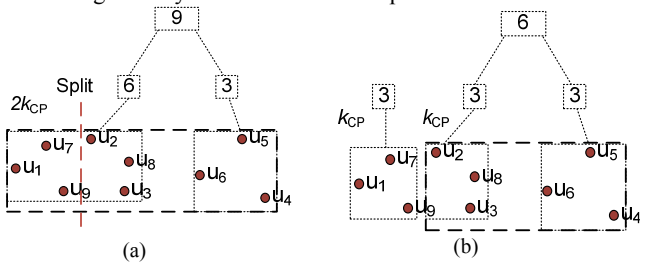

Figure 6. Node Split scenario: (a) the Node users count reach $2k_{CP}$, (b) the Node splits into two new nodes each with k_{CP} users

Importantly, no user's exact location is compromised through shrinkage of the region. The reciprocity requirement is preserved for the anonymity level k_{CP} (Section 5, Lemma 2).

4.1.4 Growth of trees

In order to achieve search efficiency in ANNC, the growth of the trees with higher anonymity level roots is necessary. The growth of a tree takes place when a new user comes in or some existing user issues a query with an anonymity level not found inside the tree to which the user is attached. It is not guaranteed that the tree will always be grown under such circumstances. However, sometimes compromising with QoS would be necessary to grow the trees and thus achieve search efficiency. The growth is allowed up to level k_{SRL}.

To better understand the situation, let's look at the example demonstrated in Figure 7. Here, four anonymity sets are shown, each with anonymity level k_{CP} = 3. Let us suppose u_3 issued a query with anonymity level 6. The smallest CR containing u_3 with anonymity level 6 would be the union of the CRs containing anonymity sets $\{u_1, u_2, u_3\}$and$\{u_4, u_5, u_6\}$. Both the CRs are the roots of anonymity level 3 and they are combined to grow a tree rooted with a Node of level 6. Then consider that the user u_8 has issued a new query with level 6. Although the anonymity sets $\{u_4, u_5, u_6\}$ and $\{u_7, u_8, u_9\}$ constitute the smallest CR area, they can not be combined to grow a tree. The reason is the CR with $\{u_4, u_5, u_6\}$ is already attached to another tree. From Figure 7(a), it is evident that the next available nearest CR would be the one with anonymity set $\{u_{10}, u_{11}, u_{12}\}$. The light dotted line denotes the probable CR. The choice of whether these trees would be combined to grow a new tree depends on what degree of QoS the application designer wants. The quality of the query is governed by a parameter called *Area Tolerance factor, M.*

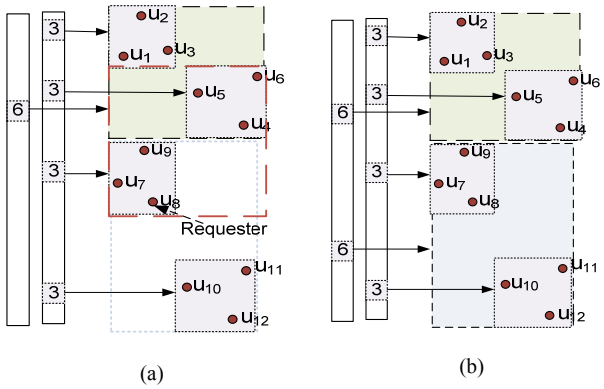

Figure 7. Growth of trees: in (a) the user u_8 made a query with anonymity level 6, but the tree can not be grown; (b) Area tolerance M has allowed to grow the tree, so two trees with anonymity level 6 are found

Area Tolerance factor, M: Before two tree roots with denoted anonymity level $k/2$ can be combined to grow a tree of level k, as a measure of quality control, the union of CRs is checked to see if the area exceeds $M * Area_k$. Here, $M >= 1$ is denoted as the Area Tolerance factor and $Area_k$ = Minimum Area of CR with anonymity level k containing the query Node.

4.1.5 New User Addition

A new user is interpreted as the user who has arrived in the system at a new location and has issued a query. The task of LA is to now find an appropriate CR for the new user and attach the Node to the appropriate tree. Most often the exact nearest neighbors are found occupied by other anonymity sets. So selection of neighbors that are not the actual nearest neighbors would degrade the quality of a CR. The system tries to preserve the QoS by temporarily assigning a CR (contains the actual nearest neighbors) and avoids constructing a tree with the current free neighbors. We present three new user insertion scenarios that are exceptions to the regular tree construction scenario involving actual nearest neighbors.

Case 1: Let us consider that the new user has arrived inside an existing CR of k_{CP} level (which is also less than $2k_{CP}$-1). The user

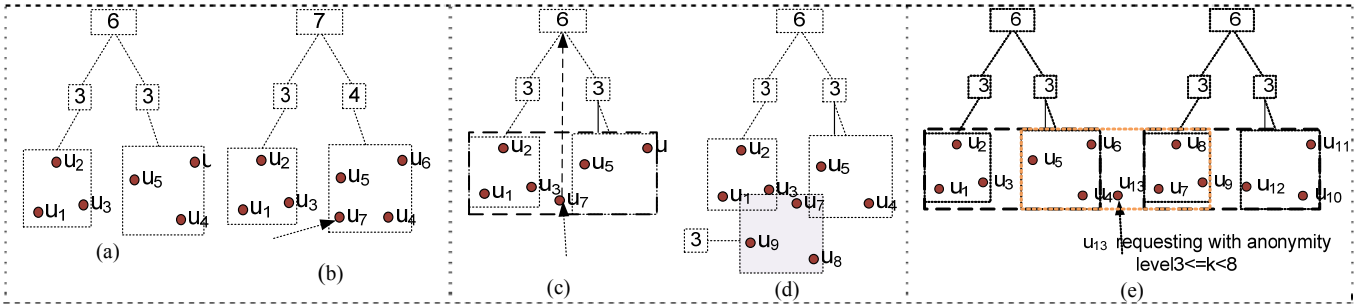

Figure 8. New user update: (a) A tree with 6 users. (b) New user u_7 came along inside the primitive cloaking region (k=3) of the nearest neighbor u_5. (c) u_7 came inside larger CR (k=6) of the nearest neighbor u_3. (d) new users u_8 and u_9 came along and formed CR with k_{CP} (e) The user u_{13} can not be merged inside larger CRs of the nearest neighbor u_4. The CR with anonymity level <8 is chosen (dotted line) by merging the CRs of nearest neighbors of two disjoint sets (u_4 and u_7)

can be attached to the parent Node of the anonymity set of the CR. For example, in Figure 8 (b) u_7 came along with anonymity level requirement 3, inside the CR of {u_4, u_5 ,u_6 }. u_7 is allowed to share the CR with them and the anonymity level is increased along the path to the root (Algorithm 1 lines 5 to 7).

Case 2: When a new user (u_7) does not land on the CR of the neighbors (u_3 or u_5) (Figure 8(c)), the neighbors' higher anonymity levels are checked to see if the new user falls inside that CR (Algorithm 1 lines 4 to 8). In Figure 8(d), the user u_7 is temporarily accommodated to the u_3's CR of anonymity level 6. When the new users u_8 and u_9 appear, the new anonymity set is formed as {u_7, u_8, u_9}. The temporary assignment is suspended (Algorithm 1 lines 9 to 12). It is important to note that there is a minor violation of the reciprocity condition since u_7 is now assigned to two different sets, but this is done for the sake of preserving QoS. In that case, a CR of even higher anonymity level can be chosen to protect the user from any potential attack.

Case 3: This is the case when the user cannot be assigned even a higher anonymity level CR of the nearest neighbor. Consider the example shown in Figure 8(e). The new user (u_{13}) with anonymity level 3 does not land inside the CRs (even with anonymity level 6) of the neighbors (u_4 or u_7). Importantly, u_4 and u_7 reside in different anonymity sets. Therefore, the system assigns a CR of anonymity level 6 to the user u_{13}, by merging two CRs of k=3 of the nearest neighbors u_4 and u_7 as in Figure 8 (c). CR of higher anonymity levels can be searched and combined in the same way.

4.1.6 ANNC CR Search

The ANNC CR Search is required to obtain a CR with the desired anonymity level. Sometimes growing of trees is performed (Algorithm 1 line 11, 20) provided that the tree root's anonymity level has not reached k_{SRL}. Figure 9 (a) demonstrates a search scenario, where the query requester is shown inside the leftmost CR. The tree is rooted with anonymity level 12 which is also the k_{SRL} of the system. The desired anonymity level of the user is 18. So, the nearest Node with anonymity level 6 is spotted (shown in black dotted lines). The union of the CRs constitutes to the minimum area of CR with anonymity level 18 and hence this area is returned. The *searchNode* method mentioned in the Algorithm 1 (line 19, 31) performs the task.

CR Area Optimization in Local Search: Optimization is achieved in Local search by looking at the actual anonymity level of the Nodes. ANNC assigns a CR Node not just based on the denoted anonymity level on the Node. In Figure 9 (b), a user on the leftmost CR issues a query with anonymity level 11. Although k_{CP} = 3, all the Nodes inside the tree have reached the anonymity level capacity. The root Node (represented with anonymity level 12) contains 20 users. If the actual anonymity level of the Nodes were not checked, the CR that belongs to the root Node would have been returned. But ANNC search returns the union of three leftmost CRs with the total anonymity level 15 (shown in dashed line in Figure 9 (b)).

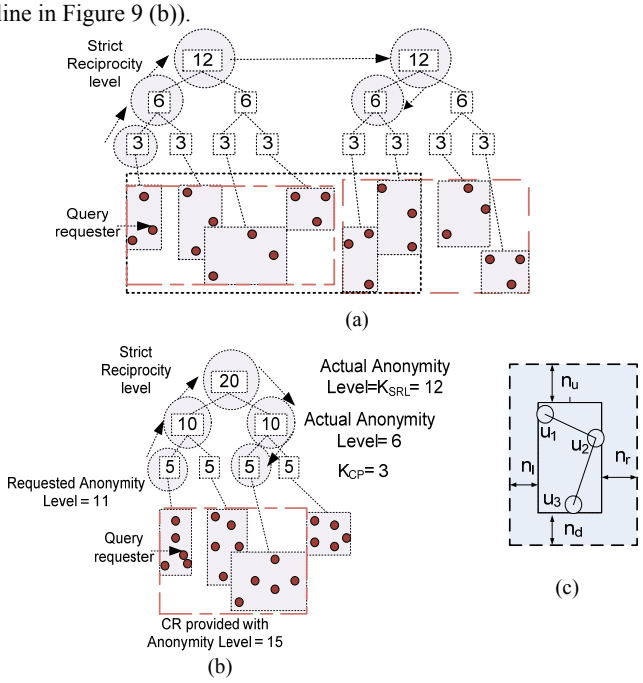

Figure 9. (a) ANNC Search for CR with k_{CP} 3, k_{SRL} 12 and requested Anonymity Level 18; (b) Local search optimization for CR with Anonymity Level 11; (c) ANNC CR Area Extension

4.1.7 ANNC delete

The ANNC delete procedure is performed when a Node has gone out of a CR with level k_{CP} or is not interested to perform any query in the near future. Figure 10 demonstrates such a scenario. In Figure 10 (b), the user u_2 has already left the CR with k_{CP}. So the anonymity levels are updated all the way to the root. When the user u_1 *or* u_3 issues a query with level 3, the CR with level 5 will be assigned to them. In Figure 10 (c), all the users have left, leaving only one CR with k_{CP}. So the root is also destroyed and the right sibling becomes disjoint again without violating the reciprocity condition. The anonymity set {u_4, u_5, u_6} will be waiting for the nearby CR of level k_{CP} to unite.

Algorithm 1: Pseudocode for ANNC CR Search

input Query issuer p, required anonymity level k, system parameters: k_{CP}, k_{SRL} , M

output a CR containing p with anonymity level $\geq k$

1. NN = Actual Nearest Neighbors' list; $Nodes[k]$ = A list of all $Node$s with level k; l = A list of $PointNode$s (current user locations);
2. IF p not attached to any tree THEN
3. $\{nn_i, 1 < i \leq k_{CP}\} \leftarrow$ Find k_{CP} nearest neighbors of p not yet attached to any tree
4. IF The neighbor lists $nn \neq NN$ THEN
5. IF ($u \leftarrow$ Find the nearest $Node$ with anonymity level k_{CP} that contains p) is $not\ null$ THEN
6. Add p to u and increase the anonymity level along the way to the root
7. Split Node if $u.level = 2k_{CP}$ and $u \leftarrow$ Node containing p
8. Return CR of u
9. ELSE
10. Add p and nn_i, $1 < i \leq k_{CP}$ to the list l
11. IF CR area of points in list $l <$ CR area of the smallest $Node$ with $2k_{CP}$ covering p THEN
12. $u \leftarrow$ Build a new $Node$ with anonymity level k_{CP} with the points in l
13. ELSE IF CR area of a Node t with $2k_{CP}$ covering p THEN
14. $u \leftarrow t$
15. ELSE
16. $u \leftarrow$ Union of $Node\ s$ with anonymity level k_{CP} containing neighbors of p
17. ENDIF
18. ENDIF
19. ENDIF
20. ENDIF
21. IF Anonymity level $u.level < k$ and k_{SRL} THEN
22. $u_1 \leftarrow$ Find the nearest non-attached Node of u with anonymity level $u.level - k$
23. IF Union of CR area of u and u_1> M*(smallest CR with level, $u.level - k$ containing p) THEN
24. $u_2 \leftarrow$ Combine u_1, u and grow a new tree with anonymity level $2 * u.level$
25. Add u_2 to $Nodes[\ 2^i k_{CP}]$, where $2^{i-1}k_{CP} \leq h \leq 2^i k_{CP}$, $0 \leq i \leq \log_2(k_{SRL}/k_{CP})$
26. Return u_2
27. ELSE
28. Return "no such CR can be found"
29. ENDIF
30. ENDIF
31. $k_2 \leftarrow u.level - k$
32. IF $k_2 > k_{SRL}$ THEN
33. $\{U_i, 1 \leq i \leq \lfloor k_2/k_{SRL} \rfloor\} \leftarrow$ Find the nearest $Nodes[k_{SRL}]$ in order s.t. union of the CR areas by U_i and u constitutes to the minimum area
34. $t \leftarrow$ Find the nearest Node of u with anonymity level k_2 mod k_{CP} from the tree rooted with $U_j, j = \lceil k_2/k_{SRL} \rceil$
35. Return the CR area formed by the union of $Nodes\ u, \{U_i, 1 \leq i \leq \lfloor k_2/k_{SRL} \rfloor\}$ and t
36. ELSE
37. $t \leftarrow$ Find the nearest Node of u with anonymity level $\geq k_2\ mod\ k_{SRL}$
38. Return the union of the CR area of u and t
39. ENDIF
40. Return CR of u

4.1.8 CR Area Extension

The NNC [6] optimizes the area of a CR by placing the users on the periphery. Extending the area of a CR can be useful in two ways: a) movements by the user will have reduced impact on CR update costs, and the same CR can be reused for some time even if the users moved; b) attacks on the CR border will be minimized due to the reduced probability of finding the users on the border of the CR.

ANNC sets a maximum limit, δ for the area extension of a CR at level k_{CP} and the extensions for 4 dimensions are randomly selected (not shown in the algorithm). As seen from Figure 9 (c),

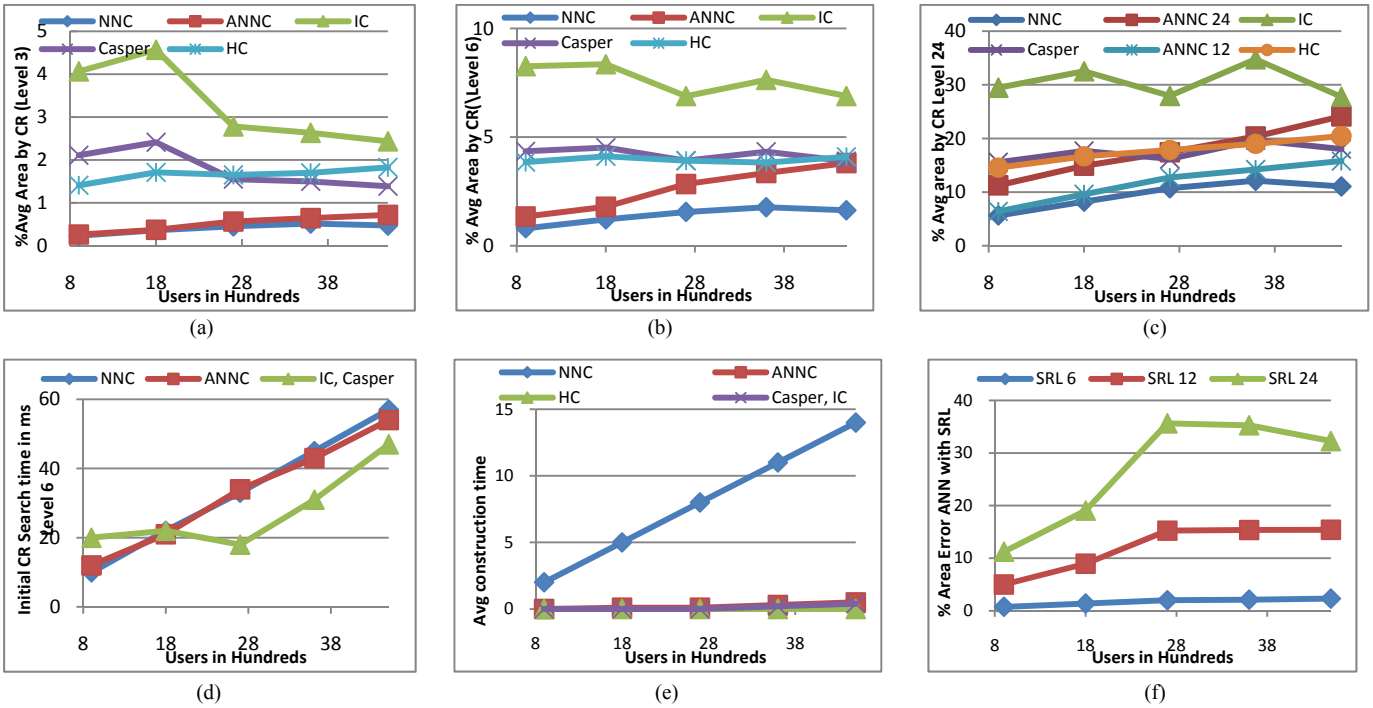

Figure 11. Average CR area generated by different Cloaking approaches with: (a) Anonymity Level 3; (b)Anonymity Level 6; (c) Anonymity Level 24 with ANNC's strict reciprocity level 6,12; (d) initial CR search time; (e) Average tree construction time for a query; (f) % Error for ANNC with different k_{SRL}s

for four the extended dimensions of a CR, the following condition holds: $(n_u + n_d) \times (n_l + n_r) \leq \delta$.

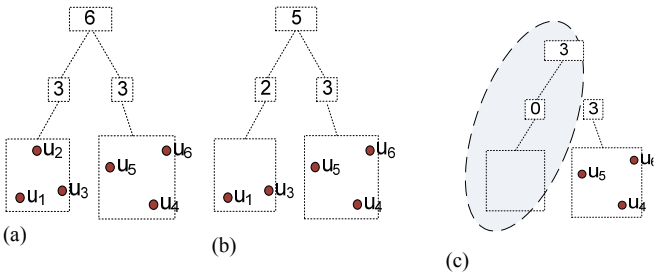

Figure 10. (a) The initial situation; (b) u_2 has moved out of the CR, the anonymity levels updated along the path; (c) all the users left, the tree root is deleted

5. Analysis of the Approach

This section lists a number of lemmas regarding preservation of the reciprocity requirement and the size of the offered anonymity level of a CR. All proofs presented here consider that the average number of users inside the CR of cloaking primitive is k_{CP}.

Lemma 1: In ANNC, the minimum anonymity level preserved for reciprocity condition is k_{CP}.

Proof: Let us consider the anonymity level of a CR, $k_1 = k_{CP}$. Again, let us take anonymity sets f_1 and f_2 consisting of k_1 and k_2 users respectively.

Intuitively, the proof is trivial for $k_1 = k_2$. From the property of ANNC: if $f_1 \neq f_2, then f_1 \cap f_2 = \emptyset$. So, $\forall u \in f_1, \forall v \in f_2 : CR(u, k_1) \neq CR(v, k_2)$, where CR stands for Cloaking Region. The anonymity level $k_2 = k_1 = k_{CP}$ is preserved for the reciprocity condition.

If $k_2 > k_1$, the ANNC requires $k_2 = ck_1, c \in I_+$. In that case, there will be two different scenarios. First, let us consider the situation, where, $\forall u \in f_1, \forall v \in f_2 : v \notin f_1$ or $f_1 \cap f_2 = \emptyset$. if u and v are the only query issuers of the anonymity sets f_1 and f_2, the anonymity levels $k_1 and\ k_2$ are preserved for u and v. Again, when $\forall u \in f_1, \exists v \in f_2 : v \in f_1$ i.e., $f_1 \subset f_2$, since the query issued by u has already disclosed the CR for the anonymity set f_1, the preserved anonymity level for both the users u and v will be $k_1 = k_{CP}$.◊

Lemma 2: In ANNC, the maximum anonymity level preserved for reciprocity condition for all users is k_{SRL}.

Proof: By contradiction, we consider that there exists an anonymity level $k_2 > k_{SRL}$ for which the reciprocity condition can be preserved for all users. Let us suppose the maximum anonymity level for the queries issued so far by any user is $k_{max} \leq k_2$. We consider two users, u and v from the same anonymity set f, are issuing queries. Furthermore, $|f| = k_{SRL}$. Intuitively, $\forall u, v \in f, |f| = k_{SRL}, k_2 > k_{SRL} : CR(u, k_{SRL}) = CR(v, k_{SRL}) \not\Rightarrow CR(u, k_2) = CR(v, k_2)$. Here, CR denotes the cloaking region for a user with desired anonymity level. Hence, the reciprocity condition cannot be preserved for $k_2 > k_{SRL}$ and can at most be preserved for k_{SRL} for all users. ◊

Lemma 3: In ANNC, the anonymity levels of the offered cloaking regions are always multiples of k_{CP}.

Proof: The proof can be divided into two parts based on the value of user's desired anonymity level. The first part considers that the user made a request with anonymity level $k_1 > k_{SRL}$, and in the second case $k_1 \leq k_{SRL}$.

In the first case, the user will be assigned a CR of anonymity level $(m+1)k_{CP}$, where $m = \lfloor k_1 / k_{CP} \rfloor$, thus achieving the bound $mk_{CP} < k_0 \leq (m+1)k_{CP}$. Here, k_0 is the offered anonymity level to the user. In the second case, the CR to be

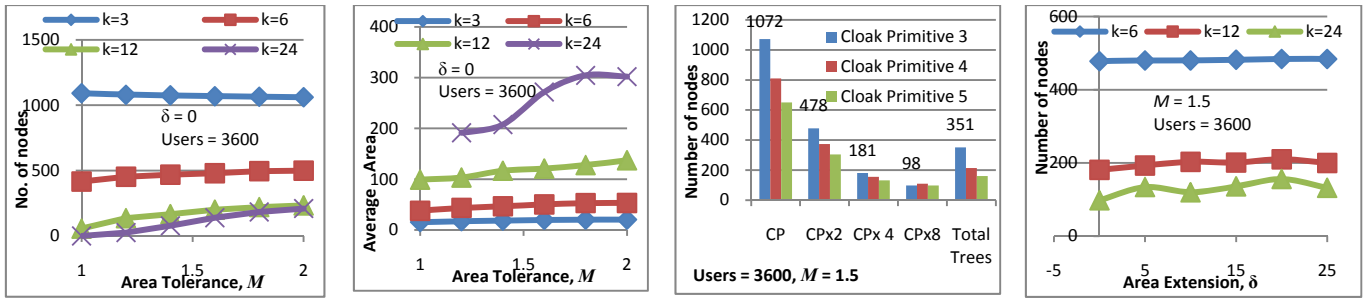

Figure 12. (a) Number of CRs with different anon levels with varying Area tolerance (b) Average area occupied by CR of anon levels with varying Area tolerance (c) Comparison of number of roots generated with k_{CP} = 4, 5 and 6(d) Number of CRs of diff. anon levels with varying Area extension

generated for offered anonymity levels are in the order of $2^i k_{CP} \leq k_{SRL}, i \in I_+$. Therefore, the offered CR will have anonymity level $2^i k_{CP}, i \in I_+$ where, $m = \lfloor k_1/k_{CP} \rfloor$ and $2^{i-1} < m \leq 2^i, i > 1$, thus achieving the bound $2^{i-1}k_{CP} < k_0 \leq 2^i k_{CP}, i > 1$ and $\forall i, i > 1: 2^i k_{CP} \leq k_{SRL}$ holds. So the anonymity levels are multiples of k_{CP}. ◊

5.1 Search Complexity

We perform the best and worst case analysis of ANNC search assuming the average number of users inside the CR of cloaking primitive is k_{CP}.

Worst case scenario: In this case, all the user in the system are considered to have performed at least a query with anonymity level $\leq k_{CP}$. The maximum height of the tree in the system is $2^0 k_{CP} = 1$. There can be at most $\lceil n/k_{CP} \rceil$ trees in the system. Therefore, if any of the users requests a CR with anonymity level $k > k_{CP}$, a search is performed among at most $\lceil n/k_{CP} \rceil$ trees in the system. So $O(\lceil n/k_{CP} \rceil)$ is the upper bound for search in ANNC.

Best case scenario: In this scenario, it is considered that the system was able to grow a maximum number of trees with height $\log_2 k_{SRL} + 1$ (fully grown trees up to k_{SRL} users). The number of trees in the system would be at most $\lceil n/k_{SRL} \rceil$. Therefore, an ANNC search will require $\lceil n/k_{SRL} \rceil$ searches for trees and a top-down search of $\log_2(k_{SRL}/k_{CP})$ at most. We find the worst case complexity in the best case scenario of ANNC search as $O(\lceil n/k_{SRL} \rceil + \log_2(k_{SRL}/k_{CP}))$.

6. Experimental Evaluation

The evaluation section describes the experiments conducted to examine the relative performance among different approaches. The experimental findings support the claim that our approach performs almost as good as NNC in terms of query optimization. On the other hand, it thwarts any attack arising from violation of reciprocity condition as well as achieves increased searching efficiency to handle moving objects in real time.

We developed prototypes for the existing and proposed cloaking techniques in Java 1.6 in order to perform comparative analysis among them. The experiments were conducted on a machine with hardware configuration Intel Processor 1.7 Ghz, 1.5 GB Memory and OS Windows XP SP2. The spatial data used in the evaluation were taken from North America data set [20] which originally consisted of 15K points. The original dataset was scaled to 100 while performing all the analysis. Among them we considered 5K points and they were fed to the algorithms as user points in 2D space. That made a convenient static snapshot for all the algorithms and helped to compare cloaking efficiency in terms of the average CR area, initial construction time, and overall CPU time with different anonymity levels. Number of candidate lists [5] has not been considered as a measure of evaluation as it bears the same meaning as the average CR area optimization.

Table 1. Summary of System Parameters

Name	Meaning
Cloaking Primitive Level, k_{CP}	Minimum anonymity level to enforce reciprocity condition
Strict Reciprocity Level, k_{SRL}	Maximum anonymity level to enforce reciprocity condition
Area Tolerance, M	Area Tolerance limit to preserve QoS
Area Extension , δ	Maximum allowable extension of a CR

6.1 Comparison among LA's Performance

Our proposed approach, ANNC, has been compared against approaches IC, Casper, NN and HC. The objective of ANNC is to safeguard mobile devices from attacks and, at the same time, maintain query quality as close to NNC's as possible. Let's take a closer look at the results of the comparative analysis.

Figure 11 (a) (b) (c) demonstrates the average CR area generated per query by 4 different approaches mentioned earlier. The queries are originated at random from all the users. The numbers of the users are varied in hundreds (900-4500). Apparently, from the result, NNC and ANNC outperform others, while HC and Casper achieve efficiency over IC. For ANNC, we set M = 1.5, δ=0.0 and k_{SRL} =24 in the average area calculation shown in Figure 11 (a) (b). In Figure 11 (c), two different k_{SRL}s (12 and 24) have been considered for ANNC while searching for the average CR area of level 24. Naturally, ANNC with k_{SRL}= 6 achieves almost the same efficiency as NNC. From Figure 11 (d), it is evident that NNC demonstrates the poorest search efficiency with respect to time. The average search times per query shows that the other three approaches significantly outperforms NNC. This is because other approaches organize the users in binary and quadtree structures, thus achieving reduced search costs.

6.2 ANNC's Performance

Let us turn to the individual performance of ANNC. Figure 11(d) shows the initial CR search times of different approaches. In the case of NNC, the search time for a query constitutes to the k Nearest Neighbor search. The ANNC takes up as much time as NNC's during the initial construction phase and constructs distinct rooted trees to achieve search efficiency for future queries. It performs as well as IC for subsequent queries as seen from the Figure 11 (e). In case of IC and Casper, the quad tree construction

Table 2. Summary of Comparison Among different Cloaking Approaches

Cloaking Technique	Initial construction	Search	CR Area	Anonymizer	Reciprocity condition Met?	Shrink Region Attack?	Region Intersection Attack?
Quadtree IC [4]	$O(n\log_4 n)$	$O(\log_4 n)$	Large	Yes	No	Yes	Yes
Quadtree Casper[5]	$O(n\log_4 n)$	$O(\log_4 n)$	Large	Yes	No	Yes	Yes
NN Cloak[7]	$O(nlog_2 n)$	$O(nlog_2 n)$	Small	Yes	No	Yes	Yes
Hilbert Cloak [7]	$O(n^2+mapping)$	$O(1)$	Large(Varies)	Yes	For single anonymity level	Yes	Yes
PrivacyGrid [18]	$O(n\log_4 n)$	*Average* $O(\log_4 n)$	Larger	Yes	No	Yes	Yes
PRIVÉ [8]	$O(n^2)$	$O(1)$	Small	No	No[1]	Yes[1]	Yes[1]
Chow et al [13]	$O(n^2)$	$O(1)$	Small	No	No[1]	Yes[1]	Yes[1]
Hashem et al[15]	$O(n^2)$	$O(1)$	Medium	No	No[1]	Yes[1]	Yes[1]
Ghinita et al[16]	N/A[2]	N/A[2]	Large	No	No[3]	No[3]	No[3]
ANNC	*Worst:* $O([n/k_{CP}])$	*Best:* $O(n/k_{SRL}+\log_2(k_{SRL}/k_{CP})$	Small	Yes	Yes	No	No

1. By compromising any of the peers, the attacker can obtain location information of the all the users (in the group)
2. Depends on the number of POIs. There is an additional overhead of determining nearest POIs and higher computation cost that involves PIR.
3. Adoption of PIR hides what information is requested from LBS. Therefore, no information about the query is revealed

time has been considered for the comparison. Unlike NNC, ANNC's search time diminishes significantly with time.

The comparative performance of ANNC with different k_{SRL}'s is shown in Figure 11 (f). The percentage (%) error has been computed as the increase of % average area over NNC. The error can go as high as 37% in the case of k_{SRL}=24. So the choice of k_{SRL} is very crucial while trying to achieve the balance between the area optimization and search efficiency. A good choice for this data set would be k_{SRL}=12 which fares well while maintaining the error around 14% most of the time.

The study with the Area Tolerance parameter, M came up with the obvious results. As shown in Figure 12 (a), the number of Nodes for anonymity level 3 constructed from 3600 users was steadily diminishing, while the number of Nodes with higher anonymity levels were on a significant rise. For example, with anonymity level 24, no Nodes were constructed at M = 0, but at M = 2.0 there were 210 Nodes. The observation is repeated in Figure 12 (b) with the average area for different anonymity levels.

We can see from the figure that there was a sharp increase in the average area for anonymity level 24 from Tolerance 1.4 to 1.8 (50%), but after that it became steady. The lower anonymity levels 3 and 6 are not affected much. The choice of Tolerance should aim at balancing between search efficiency (number of Nodes) and quality of the CR area. Figure 12 (c) presents a comparative performance of the system with different cloaking primitives. The number of constructed Nodes and roots were observed for up to $4k_{CP}$ level with Area Tolerance 1.5 and Area Extension 0.0. Interestingly enough, the number of Nodes with anonymity level 24 (98) was found less than that of anonymity level 32 (103). The total number of roots in the system for anonymity level 3, 4 and 5 were 351, 214 and 161 respectively.

The results of the Area Extension parameter can be seen from Figure 12 (d). The extensions were allowed from 0.0 to 30.0. However, no significant increase in the number of Nodes has been observed. Nodes with the higher anonymity level 24 demonstrate some irregular trend, rising and falling, whereas lower anonymity level Nodes are on a steady rise. This is because the anonymity level capacity of the Nodes with k_{CP} limits the Nodes' ability to cover as many users as possible. Finally, our ongoing experiment covers a study of restructure costs (change in the number of roots) imposed by ANNC due to random movements of the users with varying speeds.

7. Discussions and Guidelines

From the experimental facts, it is evident that ANNC is a better choice over all the existing approaches. To summarize: a) ANNC guarantees resistance from multi-query attack, whereas all other existing approaches (NNC, IC, Casper, HC) fail to safeguard against the threats. b) The search space is significantly reduced due to the adaptive data structure. c) The quality of the CR is nearly as good as the NNC's, provided a good choice of k_{SRL} is made.

ANNC provides utmost flexibility in balancing the QoS and reciprocity requirement. The purpose of k_{SRL} is to provide global search efficiency and an extended choice of local search.

As the value of k_{SRL} goes up, the performance of ANNC begins to suffer. The Area Tolerance limit, M and Area Extension, δ play a significant role in performance. Higher values of M would result in a degraded QoS but achieve significant search efficiency. Therefore, the choice of k_{SRL} and M should be made carefully, balancing the QoS and search efficiency. Meanwhile, the choice of δ should be made considering the relative speed of the mobile users.

8. Conclusion and Future directions

The industry is constantly aiming at taking handheld devices like PDAs and cell phones to the next frontier of technology by accommodating more processing and storage capability. Battery power, however, stands as the bottleneck to long hours of continuous operation for these devices. With faster new gadgets and the requirements for trusting a third party LA, researchers have become very keen on investigating LBS models [8, 15, 16, 10] that eliminate the LA. In this model, the server may need to send a data structure on selected POIs (stored as Voronoi

Tessellation with POIs) to the user prior to issuing a query. The issues with this approach is that the density control of the POIs is arbitrary at the provider's end, and the additional overhead of filtering the result set is now the query issuer's task. Future direction of privacy research in LBS is to find suitable data structures and secured protocol to support the very nature of resource-constrained devices.

References

[1] Beresford, A., and Stajano, F. 2003. Location Privacy in Pervasive Computing. IEEE PERVASIVE computing: Volume 2 Issue 1, 2003, 46-55.

[2] Sweeney, L. 2002. *K*-anonymity: a model for protecting privacy. International Journal on Uncertainty, Fuzziness and Knowledge-based Systems, 10 (5), 557-570.

[3] Bettini, C., Mascetti, S., Wang, X. S., and Jajodia, S. 2007. Anonymity in Location-based Services: Towards a General Framework. In Proceedings of Mobile Data Managment, 2007, 69-76.

[4] Gruteser, M., and Grunwald, D. 2003. Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking. In Proceedings of MobiSys 2003, 31-42.

[5] Gedik, B., and Liu, L. 2005. Location-Privacy in Mobile Systems: A Personalized Anonymization Model. In Proceedings of ICDCS 2005, 620-629.

[6] Mokbel, M. F., Chow, C. and Aref, W. G. 2006. The new casper: query processing for location services without compromising privacy. In Proceedings of VLDB 2006, 763-774.

[7] Kalnis, P., Ghinita, G., Mouratidis, K., and Papadias, D. 2007. Preventing Location-Based Identity Inference in Anonymous Spatial Queries. IEEE Transactions on Knowledge and Data Engineering, Vol 19 No. 12 (Dec 2007), 1719- 1733.

[8] Ghinita, G., Kalnis, P., and Skiadopoulos, S. 2007. PRIVÉ: Anonymous Location-Based Queries in Distributed Mobile Systems. In Proceedings of WWW 2007, 371-380.

[9] Smart phone leads market growth: http://www.pcworld.com/businesscenter/article/158697/smart_phones_lead_market_growth.html

[10] Zhong, G., and Hengartner, U. 2008. Toward a Distributed k-Anonymity Protocol for Location Privacy. In Proceedings of WPES 2008, 33-37.

[11] Xiong, X., Mokbel, M. F., and Aref, W. G. 2005. SEA-CNN: Scalable Processing of Continuous K-Nearest Neighbor Queries in Spatio-Temporal Databases. In Proceedings of ICDE 2005, 643–654.

[12] Theodoridis, Y. The R-Tree-Portal. http://ww.rtreeportal.org.

[13] Chow, C., Mokbel, M., and Liu, X. 2006. A peer-to-peer Spatial Cloaking Algorithm for Anonymous Location-based Services. In Proceedings of ACM-GIS 2006, 171–178.

[14] Krumm, J. 2007. Inference Attacks on Location Tracks. In Proceedings of Pervasive, 2007, 127-143.

[15] Hashem, T., and Kulik, L. 2007. Safeguarding Location Privacy in Wireless Ad-hoc Networks. In Proceedings of Ubicomp, 2007, 372-390.

[16] Ghinita, G., Kalnis, P., Khoshgozraran, A., Shahabi, C., and Tan, K. 2008. Private Queries in Location Based Services: Anonymizers are not Necessary. In Proceedings of SIGMOD 2008, 121 -132.

[17] Haque, M., and Ahamed, S. I. 2007. An Omnipresent Formal Trust Model (FTM) for Pervasive Computing Environment. In Proceedings of COMPSAC 2007, 49-56.

[18] Bamba, B., Liu, L., Pesti, P., and Wang, T. 2008. Supporting Anonymous Location Queries in Mobile Environments with PrivacyGrid. In Proceedings of WWW 2008, 327 – 246.

[19] Hilbert Curve: http://en.wikipedia.org/wiki/Hilbert_curve

[20] Hoh, B., and Gruteser, M. 2005. Protecting Location Privacy Through PathConfusion. In Proceedings of SecureComm 2005, 194–205.

[21] Hengartner, U. 2008. Location Privacy based on Trusted Computing and Secure Logging. In Proceedings of SecureComm 2008.