Journal of Zhejiang University-SCIENCE C (Computers & Electronics)
ISSN 1869-1951 (Print); ISSN 1869-196X (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

# EDA: an enhanced dual-active algorithm for location privacy preservation in mobile P2P networks*

Yan-zhe CHE†1, Kevin CHIEW†‡2, Xiao-yan HONG3, Qiang YANG4, Qin-ming HE1

(1*College of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China*)

(2*School of Engineering, Tan Tao University, Duc Hoa, Long An Province, Vietnam*)

(3*Department of Computer Science, University of Alabama, Tuscaloosa, AL 35487, USA*)

(4*College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China*)

†E-mail: pomme@zju.edu.cn; kevin.chiew@ttu.edu.vn

Received Sept. 20, 2012; Revision accepted Jan. 9, 2013; Crosschecked Apr. 18, 2013

**Abstract:** Various solutions have been proposed to enable mobile users to access location-based services while preserving their location privacy. Some of these solutions are based on a centralized architecture with the participation of a trustworthy third party, whereas some other approaches are based on a mobile peer-to-peer (P2P) architecture. The former approaches suffer from the scalability problem when networks grow large, while the latter have to endure either low anonymization success rates or high communication overheads. To address these issues, this paper deals with an enhanced dual-active spatial cloaking algorithm (EDA) for preserving location privacy in mobile P2P networks. The proposed EDA allows mobile users to collect and actively disseminate their location information to other users. Moreover, to deal with the challenging characteristics of mobile P2P networks, e.g., constrained network resources and user mobility, EDA enables users (1) to perform a negotiation process to minimize the number of duplicate locations to be shared so as to significantly reduce the communication overhead among users, (2) to predict user locations based on the latest available information so as to eliminate the inaccuracy problem introduced by using some out-of-date locations, and (3) to use a latest-record-highest-priority (LRHP) strategy to reduce the probability of broadcasting fewer useful locations. Extensive simulations are conducted for a range of P2P network scenarios to evaluate the performance of EDA in comparison with the existing solutions. Experimental results demonstrate that the proposed EDA can improve the performance in terms of anonymity and service time with minimized communication overhead.

**Key words:** Location-based service, Privacy preservation, Spatial cloaking, Mobile peer-to-peer networks
**doi:**10.1631/jzus.C1200267 **Document code:** A **CLC number:** TP393

## 1 Introduction

Location-based service (LBS) provides mobile users with various tailored and personalized services based on their location information. Due to the explosive deployment of smart mobile phones and the rapid growth of location determination technology (LDT), such as Cell-ID, A-GPS, EOTD (Tiwari *et al.*, 2011), mobile users benefit a lot from LBS in terms of traffic navigation, advertising recommendation, emergency service, social community, and entertainment (IETF, 2011). It is highlighted that almost three quarters (74%) of the smart phone users obtain real-time location-based information from LBS as of February 2012 (Kathryn, 2012). The examples of LBS include range query (Kalnis *et al.*, 2007), e.g., "show me a list of restaurants within 2 km distance from my current location", and nearest neighbor query (Xiong *et al.*, 2005), e.g., "where is

‡ Corresponding author

* Project (No. MOE-INTEL-11-06) supported by the MOE-Intel IT Research Fund of China

the nearest hospital?".

To access the services from an LBS provider (LSP), a user needs to reveal his private location information to the LSP, which may disclose both where he is/was and what he is/was doing. For instance, a location record can prove the fact that a user was in a certain hospital during a certain time, which will attract great interest of an insurance agent who will promote medical insurance to that user. Therefore, a user is often unwilling to disclose such information to any untrustworthy LBS servers due to the fact that a malicious adversary may obtain more about the user's private information, e.g., home address and user identity. As a matter of fact, it can even increase the criminal risk, e.g., kidnapping and domestic violence, in case where the criminals can gain access to the user's location information.

So far, many solutions have been proposed for preserving the users' location privacy (Gruteser and Grunwald, 2003; Gedik and Liu, 2005; Chow *et al.*, 2006; 2011; Liao *et al.*, 2006; Ghinita *et al.*, 2007a; Shokri *et al.*, 2011; Che *et al.*, 2012b). The most widely adopted technique is to deploy a trustworthy third party (centralized/decentralized) as a location anonymizing server (LAS) between users and LSPs. The LAS gathers sufficient location information and blurs them into a cloaked region that meets users' privacy requirements, such as $k$-anonymity (i.e., a user cannot be identified from the other $k-1$ users (Sweeney, 2002)), or the minimum cloaked region area $A_{\min}$ (i.e., a user needs to hide inside a region of minimum size $A_{\min}$).

However, in a mobile P2P network without an established communicating infrastructure, the deployment of solutions based on a trustworthy third party becomes inappropriate or even impractical. Thus, novel solutions without the participation of a third party have been proposed. The available solutions enable mobile users to cooperate with each other via P2P communication so as to blur their exact locations into a spatial cloaked region. Two implementation modes for the aforementioned solutions were proposed by Chow *et al.* (2006), i.e., on-demand and proactive modes. These two modes deploy different strategies in searching for anonymous candidate peers with long and unpredictable time of delay as well as low anonymization success rate in case users' privacy requirements are strict, e.g., requiring a very large $k$. Moreover, Chow *et al.* (2011) enhanced the existing algorithms to address the network partition problem and the 'center-of-cloaked-area' privacy attack.

Given this situation, in this work, we propose a dual-active algorithm (DA) as the basic approach. The major difference of DA as compared with other existing algorithms is that DA allows users to actively share their locations with each other. Moreover, our main idea is presented through an enhanced dual-active algorithm (EDA). It is built based on DA, yet it significantly reduces the communication overhead and evidently improves the quality of anonymization. Compared with existing algorithms, both DA and EDA increase the anonymizing speed and the anonymization success rate.

The key contributions made in this work are a set of methods that address two critical challenging problems in achieving location privacy in mobile P2P networks. First, our EDA addresses the communication overhead problem through two novel message transmission strategies, namely latest-record-highest-priority (LRHP) broadcasting and context negotiation. Second, EDA addresses the quality of location-based service problem (measured as the accuracy and latency of the responses) through the LRHP strategy and also user-driven location prediction. The technical overview of these methods is summarized as follows:

1. EDA adopts the LRHP strategy for broadcasting location records, which can lower the probability of disseminating some out-of-date location records. As a result, it can upgrade the quality of location records and also reduce the overall size of messages to be transmitted.

2. EDA allows mobile users to share locations with each other based on a prior negotiation result. This prevents duplicate or worthless location messages from being transmitted among users, and significantly reduces P2P communication overhead.

3. EDA allows a user to predict other users' locations based on their latest available locations information, i.e., other users' exact positions, moving speeds, and directions. Therefore, it can be used to solve the inaccuracy problem introduced by utilizing users' historical locations directly.

Our previous work (Che *et al.*, 2012b) briefly introduced the DA algorithm. In this paper, we not only describe the DA algorithm in detail but also propose EDA as an improved solution in terms of

less communication overhead. We evaluate the proposed DA and EDA with other existing algorithmic solutions through extensive simulation for a range of P2P network scenarios. The experimental results demonstrate the improvement of EDA in terms of communication overhead over DA for all the simulated scenarios.

## 2 Related work

The techniques proposed to preserve user location privacy can be categorized into two main classes, namely the pseudonyms and dummy location technique and the spatial cloaking technique, as reviewed in the following.

As a representative of the pseudonyms and dummy location technique, Landmark (Hong and Landay, 2004) allows a user to use the location of a certain landmark rather than sending his exact location to an LSP. Some other solutions enable a user to generate a set of fake locations and send it to an LSP so that the LSP cannot distinguish the user's exact location from these dummies, thus achieving the $k$-anonymity protection for the user (Yiu *et al.*, 2008; Shankar *et al.*, 2009; Wei *et al.*, 2012). Nevertheless, it consumes more network resources to transmit these dummy locations, and costs additional computing resources of the LSP for dealing with these fake locations.

For the spatial cloaking technique, the dedicated research efforts can be divided into three main directions (Chow and Mokbel, 2009), namely trusted third-party architecture, mobile P2P architecture, and hybrid architecture. For the first architecture, some solutions introduce a trustworthy third party serving as a middleman between a user and an LSP (Chow *et al.*, 2006; 2011; Ghinita *et al.*, 2007a; 2007b; Shokri *et al.*, 2011; Che *et al.*, 2012b). The third party collects users' real location information and blurs a cloaked region for each user based on these real locations and then communicates with those LSPs instead of the users. Since the third party has the full knowledge of users' location information as well as their requests, all the historical locations and users' interest could be revealed in case the third party is compromised by an adversary. Moreover, as the third party is the communication bottleneck, a denial-of-service (DoS) attack will frustrate the usability of the entire system.

Unlike the first centralized architecture, the mobile P2P architecture does not involve any third parties to anonymize users' location-related queries (Chow *et al.*, 2006; 2011; Ghinita *et al.*, 2007a; 2007b; Shokri *et al.*, 2011; Che *et al.*, 2012b). Chow *et al.* proposed two modes of spatial cloaking algorithms (Chow *et al.*, 2006) and three enhanced schemes (Chow *et al.*, 2011) in which users help each other generate the cloaked region. The on-demand mode executes only the candidates searching step whenever a user begins to send a query, whereas the proactive mode periodically executes that step. The information sharing scheme enables users to share their gathered location information, the historical location scheme addresses the network partition problem, and the cloaked area adjustment scheme prevents the 'center-of-cloaked-area' privacy attack. Our previous work (Che *et al.*, 2012b) introduced a dual-active mode, which reduces the anonymizing time and increases the anonymization success rate. Ghinita *et al.* (2007a; 2007b) proposed solutions to index users into a hierarchical network and built the spatial cloaked region based on the Hilbert space-filling curve. MobiCrowd (Shokri *et al.*, 2011) allows users to answer LBS queries of neighbor peers so that a querying user can preserve his location privacy from the LSP. Che *et al.* (2012a) proposed a semantics-aware location sharing framework based on the cloaking zone which takes the influence of semantic locations into consideration while generating cloaked regions.

The hybrid architecture is a mixture of the former two. Zhang and Huang (2009) proposed a solution which allows users to either request cloaking service from a centralized trustworthy third party serving as an LAS, or carry out the task in a P2P manner which balances the workload among LAS and mobile users. However, this architecture suffers from the privacy protection flaws of the other two architectures, making it easier for an adversary to violate a user's privacy.

## 3 System model

In this section, we first describe the attack models for anonymization algorithms, and then state the research problem and describe the LBS system architecture in a P2P environment before presenting user privacy requirements.

### 3.1 Attack models

This section summarizes the attack models for anonymization algorithms which preserve mobile users' location privacy. In this study, we classify all the attacks into two main categories, namely server-side attacks and peer-side attacks.

For server-side attacks, we consider that some entities on the server side, e.g., an LSP who provides service based on user locations, or a third party who participates in the location anonymization process, cannot be trusted. This is because they can gather a lot of mobile users' precise location information in the classic LBS systems. With these private data, an LSP or a third party may use them for user-unexpected commercial purposes, e.g., personalized recommendations. Moreover, an adversary who can obtain these private data may utilize them in disclosing users' more sensitive information with malicious purposes, e.g., tracing users' trajectories, finding out users' habits and interest.

For peer-side attacks, we also assume that it is not wise for users to directly share their location information with other peers because some mobile peers cannot be fully trusted only based on their identities. In some cases, some malicious peers can secretly collect users' location information or can even actively initiate a DoS attack by sending a huge amount of dummy locations to a victim so as to block the victim's communication channel.

In this study, we define the trust model by assuming that the LSP and third parties are not trustworthy while the mobile peers who participate in the anonymization process can be trusted. Thus, our proposed algorithms aim to defend server-side attacks. The solutions against peer-side attacks, however, are left for further investigation in future work.

### 3.2 Research problem

The research problem of this study is to provide an effective and efficient solution to preserve user location privacy from being disclosed to the LSPs which should also weigh users' location privacy protection against the quality of LBS services.

### 3.3 System architecture

As shown in Fig. 1, the system contains two important components, mobile users and LSPs. The mobile users are carrying mobile devices with positioning functionality which can provide location information. They can (1) access LBS on the Internet through base stations or Wi-Fi access points and (2) communicate with other mobile users via wireless local area network (LAN) or ad hoc network routing protocols (Papadopouli and Schulzrinne, 2001), which enable outdoor users to communicate within a range of approximately 250 m. From time to time, these users need to access some location-based services and send their queries together with their location information, which can be their exact positions or a blurred cloaked region (CR) generated based on users' personalized privacy requirements.

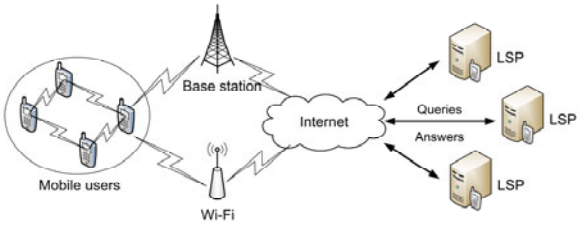


**Fig. 1 The system architecture**

On the other hand, an LSP is responsible for receiving queries from users and sending back answers to them. Once a user provides his exact location information to an LSP, some sensitive information may be disclosed by the LSP or an adversary who has compromised the LSP. For instance, the user's privacy (e.g., habits and interest) can be disclosed by analyzing his historical moving trajectory or can be violated by trading the valuable information to some malicious third parties. To preserve privacy, users often use CRs instead of their exact locations to achieve $k$-anonymity protection; i.e., one cannot distinguish a user from other $k-1$ users who are also located inside that region. On the other hand, to deal with queries based on blurred locations, an LSP can be equipped with a privacy-aware query processor (Chow *et al.*, 2009). Then the processor generates a group of candidate answers based on a region and sends the answers back to the user. The size of the candidate answer set depends mainly on the user's privacy requirements, as discussed in the following.

### 3.4 Privacy requirements

The user privacy requirements for accessing LBS can be classified into two categories:

1. Anonymous requirements. To support this type of requirements, two parameters, $k$-anonymity

and minimal size $A_{\min}$, can be used. Parameter $k$ indicates that a CR should cover at least $k$ anonymous users so that a user can achieve $k$-anonymity protection (i.e., without being known from other $k-1$ users). Generally, a larger $k$ can achieve better protection. Parameter $A_{\min}$ indicates the minimal acceptable size of the CR. Sometimes when the population is dense, $A_{\min}$ is more effective than $k$-anonymity. This is because achieving a strict $k$-anonymity protection is very easy in that situation.

2. Quality of service (QoS) requirements. It is also very important that users can enjoy LBS with better quality, e.g., accurate candidate answer sets and short latency towards receiving a service. Therefore, we deploy two parameters, $t$-latency and $A_{\max}$, to define the QoS requirements. Parameter $t$-latency is the longest tolerable time for generating a CR, and $A_{\max}$ limits the maximal acceptable size of the CR. Generally speaking, a very strict $k$-anonymity requirement may result in both a long time delay for gathering sufficient locations, which may exceed users' patience, and a larger size of CR to cover the anonymity, which indicates fewer accurate candidate answers and more communication to transmit these answers.

In summary, one cannot achieve good quality of anonymity and QoS at the same time. So, a trade-off between them needs to be investigated by tailored privacy preservation algorithms.

## 4 Spatial cloaking algorithm

This section presents our spatial cloaking algorithms for location privacy preservation in mobile P2P networks. First, we give an overview of the general algorithm. We then describe our basic idea in achieving privacy, namely the DA algorithm. Next, the algorithm is further enhanced in three detailed strategies, known as EDA, to address the overhead problem and the inaccuracy problem. Table 1 lists the notations to be used henceforth.

### 4.1 Algorithm overview

A spatial cloaking algorithm enables mobile users to cooperate with each other in sharing locations so as to generate a cloaked region that satisfies their anonymity and QoS requirements. In general, the spatial cloaking algorithm can be carried out through three steps as follows:

**Table 1 Notations and descriptions**

| Notation | Description |
|---|---|
| $k$ | Number of anonymous users for $k$-anonymity requirement |
| CR | A cloaked region |
| $A_{\min}$ | Minimal acceptable size of a CR |
| $A_{\max}$ | Maximal acceptable size of a CR |
| $t$-latency | Longest tolerable time for generating a CR |
| uid | Unique ID of a mobile user |
| hop | Routing distance between two hosts |
| $\alpha$ | Angle of a user's moving direction |
| initialTime | Generating time of the record |
| $P_{\mathrm{b}}(r)$ | Probability of broadcasting a record $r$ |
| $p$ | User-specified broadcasting probability |
| DNL | Direct neighbor list |
| CL | Candidate list |
| NL | Negotiation list |
| AL | Accept list related to an NL |
| MaxAcceptHop | Largest hop one can accept |
| MaxValidTime | Longest valid time for a record |
| RN | Ratio between the numbers of records sent by EDA and DA |
| RC | Ratio between the communication overheads caused by EDA and DA |

1. Candidates searching step. During this step, a mobile user gathers a list of mobile peers' location information, namely the candidate list (CL).

2. Cloaking step. Based on the gathered locations in the CL, the user then generates a CR according to his privacy profile, e.g., $k$-anonymity and $A_{\max}$.

3. Query processing step. In this step, the user sends a query with the CR instead of his exact location to an LSP for certain LBS. After receiving a set of candidate answers from the LSP, the user then picks out the desired ones.

Fig. 2 shows an example of algorithm execution in which all the mobile users, denoted by circles, are located in some grids of the map. Suppose that Alice, shown by the black circle in the middle of the map, wants to access a certain LBS. First, in the candidates searching step, she needs to gather enough candidate locations to fulfill her privacy requirement, e.g., $k$-anonymity, where $k$ equals 5. Hence, she begins by broadcasting a request to her direct neighbors m1, m2, and m3 (Fig. 2), asking for their locations. Then, these neighbors reply to her with their exact locations. However, since $k$ equals 5, the locations of three direct neighbors are insufficient. Therefore, she keeps sending requests to multi-hop users through the help of her direct neighbors who

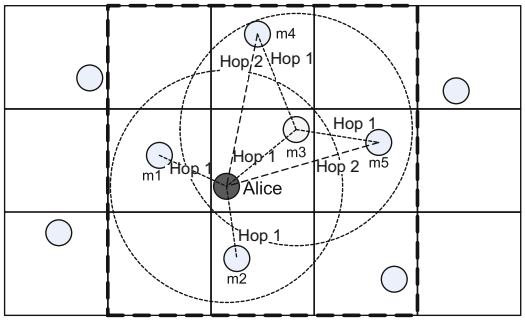


**Fig. 2 An execution demo of the spatial cloaking algorithm ($k$=5)**

will re-broadcast her request to other users. Second, when Alice has gathered enough candidates, she executes the cloaking step, which will blur the locations of these candidates into a CR by using diverse techniques. For the last step, one mobile user, e.g., m3, is randomly selected as a representative who will forward the query and CR instead of Alice to the target LSP. Equipped with a privacy-aware query processor, the LSP is able to reply to m3 with a set of candidate answers, which will then be forwarded to Alice. Finally, Alice picks out the desired answers from the answer set according to her exact location.

The candidates searching step can be executed in either on-demand or proactive mode (Chow *et al.*, 2006). For on-demand mode, the search for candidates starts only at the query of a user. In contrast, in proactive mode candidates are searched for periodically and the CL is maintained for the user no matter whether the user sends a query or not. However, both modes suffer from unpredictable long delay in finding sufficient $k$-anonymity and low anonymization success rate, particularly when users have stringent privacy requirements. Hence, in our work, a dual-active mode has been proposed to overcome these shortcomings.

### 4.2 Dual-active mode

In this section,we introduce a novel strategy DA for the mobile candidates searching step. Instead of passively asking for the location information of other peers, a mobile user not only actively sends his location to others, but also actively broadcasts his gathered candidate locations.

**Definition 1** (Location record) The candidate peer location record, denoted as $\mathcal{R}_\ell$, is defined as a 5-tuple record as follows:

$$\mathcal{R}_\ell = (\text{uid}, x, y, \text{hop}, \text{initialTime}),$$

where uid is the mobile user's unique ID, $(x, y)$ represents the coordinates of the user's exact location, hop implies the routing distance between the record owner and receivers, and initialTime shows the generating time of this record.

Whenever a record is generated, 1 is assigned to its hop. This value is increased by 1 whenever it is broadcasted to the next-hop users. The parameter initialTime means the time when a record is generated by its owner. Users can define MaxAcceptHop to control whether to accept a received record; e.g., a user refuses a record if its hop value exceeds the predefined MaxAcceptHop. A user also defines MaxValidTime in his privacy profile to manage the valid period of each location record. For example, if the mean time between initialTime and current system time exceeds the MaxValidTime constraint, the record is expired and should be removed by its owner. Algorithm 1 presents the DA algorithm in pseudo code.

**Algorithm 1** Dual-active spatial cloaking algorithm

**Input:** mobile user $U$, privacy requirements $k$, $A_{\min}$, $A_{\max}$

**Phase I:** broadcasting phase
1: **for** each user $u_{\rm d}$ in $U$'s DNL **do**
2: generate $U$'s own location record $r$;
3: send $r$ to $u_{\rm d}$;
4: **for** each location record $r'$ in $U$'s CL **do**
5: increase hop of $r'$ by 1 and send $r'$ to $u_{\rm d}$;
6: **end for**
7: **end for**

**Phase II:** receiving phase
8: **for** each record $r$ received **do**
9: **if** $r$.hop $\geq$ MaxAcceptHop **then**
10: ignore $r$;
11: **else if** $\hat{r}$ exists in CL and $\hat{r}$.uid == $r$.uid **then**
12: check and update $\hat{r}$;
13: **else**
14: ignore $r$;
15: **end if**
16: **end for**

**Phase III:** updating phase
17: **for** each location record $r$ in CL **do**
18: **if** $r$ is expired **then**
19: ignore $r$;
20: **end if**
21: **end for**

For the DA algorithm, the candidates searching step is carried out by the following three phases:

1. Broadcasting phase. The mobile devices automatically execute this phase in the background periodically, e.g., 10 s per round. As a mobile device remains online, it can discover a list of direct neighbors (DNL) that are within the one-hop transmission range using the P2P communication technique. The user first sends his location record to his direct neighbors. In this location record, uid is the unique identifier of the user, $(x, y)$ represents the user's current location coordinates, hop is 1, and the initialTime is assigned with the current system time. After that, the user also sends all the records in his CL to his direct neighbors. Note that the hop value of each record to be sent should be increased by 1, which indicates that a record is broadcasted to the next hop.

2. Receiving phase. At the same time, each mobile user receives location records from his direct neighbors. Since these records are shared among users, it is possible that some users may hold the same copy of one record at the same time. Therefore, a user may receive many duplicate records from different neighbors or even the record that represents the user himself. We allow a user to accept, reject, or update these records based on their hop and initialTime values. For instance, a record with hop=10 will be ignored by a user who specifies his profile parameter MaxAcceptHop to be 8. Moreover, if a user receives a record in which the uid has already existed in his CL, he needs to compare the initialTime values of these two records and then keep the latest one.

3. Updating phase. The updating phase is executed periodically in the background in order to discard stale location records because they will affect the anonymous quality of the CR. For each record, we calculate the mean time between current system time and its initialTime. If the mean time exceeds the user-defined MaxValidTime constraint, we should drop that expired record.

The candidates searching process will not stop until the size of CL satisfies the $k$-anonymity requirement, which means that sufficient locations have been collected for CR generation. However, if the CR does not satisfy the size requirements (i.e., $A_{\min}$ and $A_{\max}$), it will roll back to the former step to regenerate the CR. For example, if a user has collected $k$ locations but these locations are too close to him, the size of the generated CR will be too small to meet the $A_{\min}$ constraint. In this case, the user should go back to the former step to collect more locations for re-generating the CR. At the same time, the total executing time of the algorithm, namely RunningTime, should not exceed the longest tolerable time, namely $t$-latency, for QoS consideration. Otherwise, it will be marked as an unsuccessful anonymization process.

### 4.3 Enhanced dual-active mode

From the above execution process, we can observe that the DA algorithm may exhaust the communication bandwidth by arbitrarily sharing location information among mobile users. Also, it has to endure the inaccuracy problem introduced by utilizing historical locations. As elaborated in the following, the EDA algorithm introduces three strategies to address these problems, namely (1) the LRHP strategy for message broadcasting, (2) message negotiation before location sharing, and (3) location prediction.

#### 4.3.1 LRHP strategy

The LRHP strategy will reduce the probability of broadcasting some low-quality location records. As noticed, it is possible that one record can be spread to a certain user far away from that location through multi-hop P2P broadcasting. However, the further the distance one record is forwarded to, the less its quality will be. This is because a CR, with maximal area limitation $A_{\max}$, cannot expand unboundedly in order to cover a distant position. So, the EDA algorithm deploys the LRHP strategy in the broadcasting phase, which evaluates the quality of a record by its hop value. A record with a smaller hop indicates that the location of the record is closer to the user, which also implies that it takes less time for the record to be forwarded to the user (which needs less multi-hop P2P communication). Therefore, a location record with hop=2 should have a higher probability of being broadcasted than a record with hop=6. Hence, we define the broadcasting probability as follows:

**Definition 2** (Broadcasting probability) Given an EDA location record $\mathcal{R}_{\text{eda}}$ received by a user, the probability for the user to broadcast this record, denoted as $P_{\text{b}}(\mathcal{R}_{\text{eda}})$, can be calculated by

$$P_{\text{b}}(\mathcal{R}_{\text{eda}}) = p^{\text{hop}}, \tag{1}$$

where $p$ $(0 \leq p \leq 1)$ is a user-specified parameter

and hop is equal to the hop value of $\mathcal{R}_{\text{eda}}$.

From the definition, we can see that if a user sets $p = 0$, then the user will not broadcast any records to his neighbors, whereas if the user sets $p = 1$, the user will forward every record he maintained to his neighbors. This is the same as in the DA algorithm.

#### 4.3.2 Negotiation process

The negotiation process aims at preventing users from sending duplicate records to each other. In the situation, after sharing locations with each other for a period of time, a group of nearby mobile users may have a very similar CL. Therefore, continuously sending some duplicate locations to each other will be of little help but only increase the communication overhead among them. So, we allow mobile users to perform a negotiation process during which users first select a group of uids following the LRHP strategy, and then put them into a list known as the negotiation list (NL). Next, they send the NL to their neighbors and receive a corresponding accept list (AL) in which those duplicate records' uids are dropped by their neighbors. Finally, by using a list of uids in the AL, both the sender and receiver have made decision on which records to send or to receive. We refer to the ratio of the AL size to the NL size as the acceptance ratio, as defined below:

**Definition 3** (Acceptance ratio) Given a user's NL and its corresponding AL sent back from one of his neighbors, we define the acceptance ratio of these records accepted to be sent, denoted as $q$, as

$$q = \frac{S(\text{AL})}{S(\text{NL})},$$

where $S(\text{AL})$ and $S(\text{NL})$ denote the sizes of the two lists, respectively. Given that a user has $m$ direct neighbors, we can also define the average acceptance ratio, denoted as $\bar{q}$, as

$$\bar{q} = \frac{\sum_{i=1}^{m} q_i}{m}.$$

In other words, during the negotiation process, there are only $q$ portion of records maintained in an NL which need to be transmitted to neighbors. We have $0 \leq q \leq 1$ since the size of AL is no more than that of NL. In practice, the value of $q$ may be less than 0.1 (observed in the experiments), indicating that a great number of reduplicative records are stopped from being sent.

The total amount of communication overhead between two users for the EDA algorithm, measured by the total length of messages to be transmitted, consists of three parts, namely (1) the length of the location records to be sent, (2) the length of NL, and (3) the length of AL. Compared with DA, EDA sends/receives only some negotiation messages in the negotiation process, instead of directly sending a group of location records. Since the negotiation messages only consist of uids of the records, the sum of the lengths of NL and AL is much smaller than the total length of these duplicate records. Therefore, EDA can decrease the overhead and save users' P2P communication bandwidth. Let $L(R_{\text{eda}})$ and $L(R_{\text{da}})$ be the total lengths of records to be sent by EDA and DA respectively, and $L(\text{NL})$ and $L(\text{AL})$ be the total lengths of NL and AL of EDA respectively. The following theorem will analyze the overall overhead decrease for the EDA algorithm:

**Theorem 1** Let the length of a location record be $\ell_{\text{r}}$ bytes, and the length of a corresponding negotiation message be $\ell_{\text{n}}$ bytes. Let RN be the ratio between the total numbers of location records sent by the two algorithms, i.e., $\text{RN} = \dfrac{L(R_{\text{eda}})}{L(R_{\text{da}})}$. Let RC be the ratio between the communication overheads of the two algorithms, i.e., $\text{RC} = \dfrac{\text{Overhead}_{\text{eda}}}{\text{Overhead}_{\text{da}}}$. We have the upper bound of RC as follows:

$$\text{RC} \leq \text{RN} + \frac{\ell_{\text{n}}}{\ell_{\text{r}}}(1 + \bar{q}), \tag{2}$$

where $\bar{q}$ is the average acceptance ratio.

**Proof** First, we can deduce that $\ell_{\text{r}} > \ell_{\text{n}}$ because the negotiation message contains only the uid part of a record. Second, from the definition of RC, we have

$$\text{RC} = \frac{\text{Overhead}_{\text{eda}}}{\text{Overhead}_{\text{da}}} = \frac{L(R_{\text{eda}}) + L(\text{NL}) + L(\text{AL})}{L(R_{\text{da}})}.$$

Given that a user has $n$ location records and $m$ direct neighbors, we know that DA sends $mn$ records and EDA sends $mn \cdot \text{RN}$ records. Then we have $L(R_{\text{da}}) = \ell_{\text{r}} mn$ and $L(R_{\text{eda}}) = \ell_{\text{r}} mn \cdot \text{RN}$. For EDA, the worst case for the NL is that the uids of all $n$ records are put into the NL (the same influence as setting $p = 1$). Thus, we have $L(\text{NL}) \leq \ell_{\text{n}} mn$ and $L(\text{AL}) \leq \ell_{\text{n}} mn\bar{q}$, where $\bar{q}$ is the average ratio of the AL size to the NL

size. Finally, we have

$$\begin{aligned}\mathrm{RC} &= \frac{L(R_{\mathrm{eda}})+L(\mathrm{NL})+L(\mathrm{AL})}{L(R_{\mathrm{da}})}\\ &\le \frac{\ell_{\mathrm{r}}mn\cdot\mathrm{RN}+\ell_{\mathrm{n}}mn+\ell_{\mathrm{n}}mn\bar{q}}{\ell_{\mathrm{r}}mn}\\ &= \frac{\ell_{\mathrm{r}}\cdot\mathrm{RN}+\ell_{\mathrm{n}}(1+\bar{q})}{\ell_{\mathrm{r}}}\\ &= \mathrm{RN}+\frac{\ell_{\mathrm{n}}}{\ell_{\mathrm{r}}}(1+\bar{q}).\end{aligned}$$

As a result, we can use RC to quantitatively evaluate the performance of the EDA algorithm in terms of communication overhead against the DA algorithm. To calculate the value of RC, we should first estimate the value of RN. By definition, we have $\mathrm{RN}=\dfrac{L(R_{\mathrm{eda}})}{L(R_{\mathrm{da}})}=\dfrac{N_{\mathrm{eda}}}{N_{\mathrm{da}}}$ where $N_{\mathrm{eda}}$ and $N_{\mathrm{da}}$ denote the numbers of location records sent in one broadcasting process of EDA and DA, respectively. Owing to the LRHP strategy, the records with small hop are more likely to be sent. Therefore, RN is closely related to the percentage of records with different hops in the CL. In what follows, we investigate three types of classic distributions for the records with different hop values, i.e., uniform distribution, exponential distribution, and Poisson distribution. We also show the examples of RN values under different distributions.

**Theorem 2** Supposing that the number of records in CL with different hop values follows uniform distribution, then $\mathrm{RN_u}$ (subscript 'u' means uniform distribution) can be calculated as

$$\mathrm{RN_u}=\begin{cases}\dfrac{(1-p^H)p\bar{q}}{(1-p)H}, & 0\le p<1,\\ \bar{q}, & p=1,\end{cases}\tag{3}$$

where $p$ is the broadcasting probability, $H$ is the MaxAcceptHop value, and $\bar{q}$ is the average acceptance ratio.

**Proof** Given a user who has $n$ location records stored in his CL and that the numbers of records with different hop values are equal, we can calculate the total number of records to be shared by EDA (before the negotiation process) as

$$\frac{n}{H}p^1+\frac{n}{H}p^2+\cdots+\frac{n}{H}p^H=\sum_{h=1}^{H}\frac{n}{H}p^h.$$

If $p=1$, the value equals $n$, which is exactly the same as in DA. In the negotiation process, given that a user has $m$ direct neighbors with corresponding values of $q$, we have

$$N_{\mathrm{eda}}=\sum_{h=1}^{H}\frac{n}{H}p^h\cdot\sum_{i=1}^{m}q_i,\quad p\ne 1,$$

and $N_{\mathrm{da}}=nm$. Thus, $\mathrm{RN_u}$ can be calculated as

$$\begin{aligned}\mathrm{RN_u} &= \frac{\sum_{h=1}^{H}\frac{n}{H}p^h\cdot\sum_{i=1}^{m}q_i}{nm}\\ &= \frac{\sum_{h=1}^{H}p^h\cdot\sum_{i=1}^{m}q_i}{Hm}\\ &= \frac{(1-p^H)pm\bar{q}}{(1-p)Hm}\\ &= \frac{(1-p^H)p\bar{q}}{(1-p)H}.\end{aligned}$$

If $p=1$, we have

$$\mathrm{RN_u}=\frac{n\sum_{i=1}^{m}q_i}{nm}=\frac{m\bar{q}}{m}=\bar{q}.$$

For example, if we have $p=0.8$, $H=10$, and $\bar{q}=8\%$ then we have $\mathrm{RN_u}=2.85\%$. It indicates that EDA sends only 2.85% of the records as required in DA. If we let $\ell_{\mathrm{r}}=64$ bytes and $\ell_{\mathrm{n}}=4$ bytes, by Theorem 1 we have the upper bound of RC, 9.6%. This means that EDA costs only 9.6% of the communication overhead required by DA, showing a significant decrease as expected.

**Theorem 3** Supposing that the number of records in CL with different hop values follows exponential distribution, then $\mathrm{RN_e}$ (subscript 'e' means exponential distribution) can be calculated as

$$\mathrm{RN_e}=\begin{cases}\lambda\bar{q}p\dfrac{1-(\mathrm{e}^{-\lambda}p)^H}{\mathrm{e}^{\lambda}-p}, & 0\le p<1,\mathrm{e}^{\lambda}\ne p,\\ \lambda\bar{q}H, & 0\le p<1,\mathrm{e}^{\lambda}=p,\\ \bar{q}, & p=1,\end{cases}\tag{4}$$

where $p$ is the broadcasting probability, $H$ stands for the MaxAcceptHop value, $\bar{q}$ is the average acceptance ratio, and $\lambda$ is the rate parameter of the distribution.

**Proof** Given exponential distribution, we can calculate the number of records at certain hop $=h$ as $n\lambda\mathrm{e}^{-\lambda h}$. If $0\le p<1$, then the total number of records to be shared by EDA (before the negotiation process) can be calculated as

$$n\lambda\mathrm{e}^{-\lambda}p+n\lambda\mathrm{e}^{-2\lambda}p^2+\cdots+n\lambda\mathrm{e}^{-H\lambda}p^H=n\lambda\sum_{h=1}^{H}(\mathrm{e}^{-\lambda}p)^h.$$

Given that a user has $m$ direct neighbors, then we have

$$N_{\mathrm{eda}} = n\lambda \sum_{h=1}^{H} (\mathrm{e}^{-\lambda} p)^h \cdot \sum_{i=1}^{m} q_i, \quad p \neq 1,$$

and $N_{\mathrm{da}} = nm$. Thus, $\mathrm{RN_e}$ can be calculated as

$$\begin{aligned} \mathrm{RN_e} &= \frac{n\lambda \sum_{h=1}^{H} (\mathrm{e}^{-\lambda} p)^h \cdot \sum_{i=1}^{m} q_i}{nm} \\ &= \lambda \bar{q} \sum_{h=1}^{H} (\mathrm{e}^{-\lambda} p)^h \\ &= \begin{cases} \lambda \bar{q} p \dfrac{1 - (\mathrm{e}^{-\lambda} p)^H}{\mathrm{e}^{\lambda} - p}, & \mathrm{e}^{\lambda} \neq p, \\ \lambda \bar{q} H, & \mathrm{e}^{\lambda} = p. \end{cases} \end{aligned}$$

When $p = 1$, we have $N_{\mathrm{eda}} = nm\bar{q}$. Then,

$$\mathrm{RN_e} = \frac{nm\bar{q}}{nm} = \bar{q}.$$

For example, if we let $p = 0.8$, $H = 10$, $\bar{q} = 8\%$, and $\lambda = 0.31$ for exponential distribution (we observe that the mean value of the distribution is approximately equal to 3.2 in the experiments), then we have $\mathrm{RN_e} = 3.51\%$. We notice that $\mathrm{RN_e}$ is slightly greater than $\mathrm{RN_u}$ (which is 2.85%) in this case. This is because the number of records with small hop values under exponential distribution is greater than that under uniform distribution, leading to a slightly larger number of records to be shared. If we let $\ell_{\mathrm{r}} = 64$ bytes and $\ell_{\mathrm{n}} = 4$ bytes, then by Theorem 1 we know that the upper bound of RC is 10.26%.

**Theorem 4** Supposing that the number of records in CL with different hop values follows Poisson distribution, then $\mathrm{RN_p}$ (subscript 'p' means Poisson distribution) can be calculated as

$$\mathrm{RN_p} = \begin{cases} \mathrm{e}^{-\lambda} \bar{q} \sum_{h=1}^{H} \dfrac{(\lambda p)^h}{h!}, & 0 \leq p < 1, \\ \bar{q}, & p = 1, \end{cases} \tag{5}$$

where $p$ is the broadcasting probability, $H$ stands for the MaxAcceptHop value, $\bar{q}$ is the average acceptance ratio, and $\lambda$ is the rate parameter of the distribution.

**Proof** Given that a user has $n$ location records stored in his CL and that the number of records follows Poisson distribution, we can calculate the number of records at certain hop $= h$ as $n\dfrac{\lambda^h \mathrm{e}^{-\lambda}}{h!}$. If $0 \leq p < 1$, then the total number of records to be shared by EDA (before the negotiation process) can be calculated as

$$\begin{aligned} &\frac{n\lambda \mathrm{e}^{-\lambda}}{1!} p^1 + \frac{n\lambda^2 \mathrm{e}^{-\lambda}}{2!} p^2 + \cdots + \frac{n\lambda^H \mathrm{e}^{-\lambda}}{H!} p^H \\ &= \sum_{h=1}^{H} \frac{n\mathrm{e}^{-\lambda} \lambda^h p^h}{h!} = n\mathrm{e}^{-\lambda} \sum_{h=1}^{H} \frac{(\lambda p)^h}{h!}. \end{aligned}$$

Given that a user has $m$ direct neighbors, we have

$$N_{\mathrm{eda}} = n\mathrm{e}^{-\lambda} \sum_{h=1}^{H} \frac{(\lambda p)^h}{h!} \cdot \sum_{i=1}^{m} q_i, \quad p \neq 1,$$

and $N_{\mathrm{da}} = nm$. Thus, $\mathrm{RN_p}$ can be calculated as

$$\begin{aligned} \mathrm{RN_p} &= \frac{n\mathrm{e}^{-\lambda} \sum_{h=1}^{H} \dfrac{(\lambda p)^h}{h!} m\bar{q}}{nm} \\ &= \mathrm{e}^{-\lambda} \bar{q} \sum_{h=1}^{H} \frac{(\lambda p)^h}{h!}. \end{aligned}$$

When $p = 1$, we have $N_{\mathrm{eda}} = nm\bar{q}$. Then $\mathrm{RN_p}$ is

$$\mathrm{RN_p} = \frac{nm\bar{q}}{nm} = \bar{q}.$$

For example, if we let $p = 0.8$, $H = 10$, $\bar{q} = 8\%$, and $\lambda = 3.2$ for Poisson distribution (observed in the experiments), then we have $\mathrm{RN_p} = 3.8\%$. Given that $\ell_{\mathrm{r}} = 64$ bytes and $\ell_{\mathrm{n}} = 4$ bytes, we know that the upper bound of RC is 10.55%. This is also a significant improvement of EDA performance in terms of less communication overhead. The experimental results in Section 5.2 show that the number of records follows Poisson distribution in our experiments.

#### 4.3.3 Location prediction

The use of the historical location records can bring some advantages, e.g., reducing communication overhead between peers and solving the network partition problem (Chow *et al.*, 2011). However, directly using these stale locations without considering user movement (Che *et al.*, 2012b) will definitely cause an inaccuracy problem when generating a CR. For example, user $U_{\mathrm{a}}$ received a record from user $U_{\mathrm{b}}$ who was very close to $U_{\mathrm{a}}$ in position about 10 s before. Supposing that $U_{\mathrm{b}}$ is leaving fast away from $U_{\mathrm{a}}$, the CR, which was generated with the old location of $U_{\mathrm{b}}$, may not contain enough anonymity due to the fact that $U_{\mathrm{b}}$ is currently outside that CR.

The purpose of location prediction is to solve the inaccuracy problem caused by using these out-of-date locations. Chow *et al.* (2011) proposed a strategy which calculates a circular area as the peer's possible location at current time with a radius of meantime × speed, where meantime indicates the time between the location record generation and the current time, and speed is the estimated maximum possible speed of the peer. In this work, EDA enables a user to predict other users' current locations in a different way. In this process, EDA updates the coordinates $(x, y)$ based on their moving speeds and directions, which can be obtained from a positioning device like the Global Positioning System (GPS). Hence, we define the EDA location record by extending Definition 1 as follows:

**Definition 4** (EDA location record) The EDA location record, denoted as $\mathcal{R}_{\text{eda}}$, is defined as a 7-tuple record:

$$\mathcal{R}_{\text{eda}} = (\text{uid}, x, y, \text{hop}, \text{initialTime}, \text{speed}, \alpha), \quad (6)$$

where the first five parameters are the same as stated in Definition 1 and the last two parameters, speed and $\alpha$, represent the moving speed and direction of the user, respectively.

Since we have obtained a peer's speed and moving direction, we can predict his location coordinates $(x, y)$ at current time by the following formulas: $x = x + \text{speed} \cdot \cos\alpha$ and $y = y + \text{speed} \cdot \sin\alpha$. Compared with the former prediction method (Chow *et al.*, 2011), we notice that both location prediction strategies have their respective pros and cons. The former method is more conservative and safer, which can guarantee the region to contain as many as $k$ anonymous users. It can achieve a more secure $k$-anonymity protection at the expense of a relatively large region area as well as the additional candidate answers to transmit. On the other hand, our method may cause a location predicting error by using the stale movement condition which may vary with time. However, the advantage is that, we can narrow down a user's position to a small range with a high probability that does not require the arbitrary expansion of the size of the CR, hence resulting in a smaller candidate answer set and thus reducing the communication overhead. Moreover, we enable users to control the side effect of inaccurate location prediction by setting a proper value to the user-defined constraint MaxValidTime. For example, if MaxValidTime is set to be a small value, say 4 s, the location predicting error may be a relatively small and negligible value compared with the total size of the CR.

#### 4.3.4 Algorithm specification

Algorithm 2 presents the EDA algorithm in pseudo code. With the above three strategies, EDA enhances the three phases from DA. The details of the phases are described as follows.

**Algorithm 2** Enhanced dual-active spatial cloaking algorithm

**Input:** mobile user $U$, privacy requirements $k$, $A_{\min}$, $A_{\max}$

**Phase I:** broadcasting phase

add $U$'s own uid into the NL;
**for** each record $r$ in $U$'s CL **do**
  have a probability $p^{\text{hop}}$ to add $r$'s uid into NL;
**end for**
**for** each user $u_{\text{d}}$ in $U$'s DNL **do**
  send the NL to $u_{\text{d}}$;
  wait and receive the AL from $u_{\text{d}}$;
  **for** each uid in the AL **do**
    prepare $r$ according to uid;
    send $r$ to $u_{\text{d}}$;
  **end for**
**end for**

**Phase II:** receiving phase

wait and receive NL from a user $u_{\text{d}}$;
**for** each uid in $U$'s own NL **do**
  **if** uid does not exist in CL **then**
    add uid to AL;
  **end if**
**end for**
send AL to user $u_{\text{d}}$;
wait and receive a set of $r$'s from $u_{\text{d}}$;
**for** each record $r$ received **do**
  **if** $r$.hop $\geq$ MaxAcceptHop **then**
    ignore $r$;
  **else**
    add $r$ to CL;
  **end if**
**end for**

**Phase III:** updating phase

**for** each candidate record $r$ in CL **do**
  **if** $r$ is expired **then**
    remove $r$;
  **else**
    update $r$'s $(x, y)$ based on speed and $\alpha$ values;
  **end if**
**end for**

1. Broadcasting phase. A mobile user does not directly broadcast the records to neighbors but first prepares an NL instead. The user adds his own uid into the NL. Subsequently, for each record inside his CL, the user calculates the probability to add the record's uid into the NL by Eq. (1) in Definition 2. After preparing the NL, the user sends it to all his direct neighbors. As the NL contains only a set of uids, its largely reduced size makes it more practicable to be transmitted via P2P communication as compared with the DA algorithm which directly transmits these full-size records. When one neighbor replies to the user with an AL which contains a set of uids accepted for transmitting, the user then sends corresponding records to the neighbor, according to the record's unique uid. Note that the hop value of any record to be sent should be added by 1, indicating that the record is broadcasted to a further routing hop.

2. Receiving phase. This phase runs automatically at the background. During this stage one user listens to all his direct neighbors, each of whom may send an NL. It then generates an AL based on the received NL. For example, upon receiving an NL from user Alice, user Bob now has a list of candidate uids and then searches for each uid from his CL. If one uid does not exist in the CL, which means that Bob does not have the location record, he then adds it to the AL. The final AL will be sent back to Alice. After that, Alice sends to Bob a group of records according to the AL. For each record Bob receives, if the hop value satisfies the MaxAcceptHop constraint, he then saves the record to his CL or ignores it otherwise.

3. Updating phase. In this phase, we first remove those out-of-date location records by checking their existing time, which is similar to that in DA, and update these valid locations based on their speed and $\alpha$ parameters. The updated coordinates $(x, y)$ should be $x = x+\text{speed}\cdot\cos\alpha$ and $y = y+\text{speed}\cdot\sin\alpha$, respectively. Parameter $\alpha$ is the angle of the user's moving direction in a 2D coordinate system; e.g., if the user is moving northward, then $\alpha$ should be 90°.

### 4.4 Complexity analysis

Based on the detailed descriptions in Algorithms 1 and 2, the time complexity of either DA or EDA can be divided into three parts according to the three processing phases of the algorithms. Suppose that the average number of neighbors in the DNL (list of direct neighbors) is $m$ for each user and that the average number of records in the CL is $n$. In general, we have $m < n$ for most of the cases because $m$ is limited by the capable transmission range of one-hop P2P communication. From Algorithm 1, we know that: (1) in the broadcasting phase, it takes $O(mn)$ time for each user to send $n$ records to each of the $m$ neighbors, (2) in the receiving phase, it requires $O(mn)$ time to receive $n$ records from each neighbor, and (3) in the updating phase, it needs $O(n)$ time to go through and update all the records in CL. Therefore, the overall time complexity of DA is $O(mn)$.

For EDA, from Definition 3, we know the acceptance ratio $q \leq 1$ because some duplicate records in NL will be filtered out. From Algorithm 2, we know that: (1) in the broadcasting phase, EDA first takes $O(n)$ time to traverse the CL to generate an NL, and then it requires $O(p^h qmn)$ time to send each one of the $m$ neighbors with $p^h qn$ records, where $p$ is the broadcasting probability and $h$ is the mean hop value of the records, (2) in the receiving phase, EDA takes $O(p^h qmn)$ time to receive $p^h qn$ records from each neighbor, and (3) in the updating phase, it requires $O(n)$ time, which is the same as in DA. Therefore, the overall time complexity of EDA is $O(p^h qmn)$. Since we have $p^h \leq 1$ and $q \leq 1$, we can conclude that the time complexity of EDA is no more than that of DA.

### 4.5 Discussion

Both DA and EDA allow users to control the algorithm executing process and performance goals by setting a group of parameters in the personal profiles. They are $k$, MaxValidTime, MaxAcceptHop, $t$-latency, $A_{\min}$, and $A_{\max}$.

As mentioned above, the MaxValidTime parameter controls the period of validity for each stale location record and the MaxAcceptHop parameter controls the furthest P2P communication among mobile peers. The assignment for these user-specified parameters, e.g., MaxAcceptHop and MaxValidTime, will logically result in different inferences on the performance of EDA. If we assign a large value to MaxValidTime, a location record can be used for a long time. Although using the stale location records will cause the inaccuracy problem of a CR, it promotes the anonymization success rate, reduces the anonymizing time, and solves the network partition

problem. Similarly, if we set a large value to MaxAcceptHop, users can gather more location records from long hop-distance users, which can increase the anonymization success rate. However, it costs more network resource to transmit these records. Moreover, if users assign a small value to $A_{\min}$, these records may be useless for CR generation. A tradeoff needs to be made between convenience and the quality of anonymization by assigning proper values to these parameters.

Given the unique limitations of the mobile environment, our algorithms face several challenges (Mokbel and Chow, 2006). One of these challenges is the limited battery power of mobile devices, which is believed to be closely related to the executing time of an algorithm, which can be partially measured by the anonymizing time, and the overall communication overhead. The comparison among different algorithms in terms of these two aspects will be made using a set of experiments in the next section. On the other hand, the mobile users can disconnect themselves from the network frequently for saving energy or reducing network failure, which may affect the functioning of our algorithms. However, in this study, we assume that the mobile users do not suffer from this problem and we believe this issue is worth further investigation in the future work.

## 5 Performance evaluation

This section evaluates the proposed DA and EDA algorithms in comparison with three existing, closely related algorithms as benchmarks, i.e., on-demand, proactive algorithms (Chow *et al.*, 2006) and IS-HL (which is the acronym for the combination of information sharing scheme and historical location scheme) algorithms (Chow *et al.*, 2011). We compare these algorithms with respect to four key performance metrics. (1) Anonymization success rate. This is the metric to measure whether an algorithm satisfies a user's $k$-anonymity requirement by gathering the locations of enough users within a given time. It is the ratio of the number of users who successfully generate the cloaked region to the total number of users. Its value should be within $[0, 1]$ and the greater the value, the better the performance. (2) Average anonymizing time per query. It measures the response time of the anonymizing process for each algorithm. It is defined as the average mean time between a user initiating a query and finishing generating a CR for the query. The shorter the time it requires, the better the performance it achieves. (3) Average communication overhead per query. It measures the total length of all the messages caused by a query's anonymizing process. It consists of all the messages sent and received by a user during a period of simulation time. (4) Cloaked region size. It measures the average size of the cloaked region generated by the algorithms. This also indicates the quality of the cloaked region since the larger the region, the more the candidate answers that can be generated by the LBS provider, and the more the computing and communicating resources of the mobile devices that will be spent on dealing with these additional candidate answers.

### 5.1 Experimental settings

Based on the realistic road infrastructure of the San Francisco Bay area, we construct our data by using the generator of network-based moving objects (Brinkhoff, 2002), which simulates the movement for a group of mobile users varying from 2000 to 5000.

For each scenario of the experiments, the simulation time is 200 s during which each mobile user randomly selects a certain moment to launch a query.

Without losing generality, we consider the processing time for a message, including receiving and sending times, to be 100 ms for all mobile devices.

For user requirement, the $k$-anonymity varies within a range of $[10, 70]$ (default as [20, 50]) and the transmission range is between 100 and 200 m, randomly set for each user.

For all the experiments, the default values for MaxAcceptHop and MaxValidTim are 10 s and 8 s, respectively, and the default value of $p$ for the EDA algorithm is 0.8.

For communication overhead measurement, we suppose that the length of either a P2P communication message or a location record is 64 bytes, and parameter uid in a record is 4 bytes, which will be used in calculating the negotiation message.

All algorithms are implemented by Java and run on a personal computer with Intel Core2 Quad 2.4 GHz CPU and 1.96 GB RAM. The specification of simulation parameters is summarized in Table 2.

**Table 2 Experimental parameter specification**

| Parameter | Value |
|---|---|
| User population | 2000–5000 |
| Simulation time | 200 s |
| Message processing time | 100 ms |
| $k$-anonymity | [10, 40] to [40, 70] |
| Transmission range | 100–200 m |
| MaxAcceptHop | 6–12 |
| MaxValidTime | 4–10 s |
| $p$ | 0.7–1 |
| Message length | 64 bytes |

### 5.2 Location records under Poisson distribution

Theorem 4 assumes that the numbers of records with different hop values follow Poisson distribution. In this experiment, we simulate a scenario of 3000 users sharing and receiving records for 200 s. We observe that the average number of records that a user maintains is about 187. We also count the record number of each hop. Fig. 3 gives the numeric results, showing that the distribution of the records is in conformity with Poisson distribution. From Fig. 3, we also calculate the rate parameter ($\lambda$) of Poisson distribution, which equals 3.2.

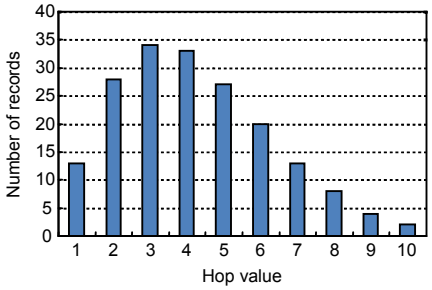


**Fig. 3 Location records under Poisson distribution**

### 5.3 Scalability analysis

We first evaluate the scalability of the algorithms along with the increasing number of mobile users from 2000 to 5000 (Fig. 4). Fig. 4a shows that the on-demand and proactive algorithms achieve the same anonymization success rate for all scenarios. This is because they gather the same size group of anonymity, which directly affects the anonymization success rate. On the other hand, IS-HL, DA, and EDA promise a higher success rate because they all utilize some stale locations generated a short time before, making it easier for users to fulfill their $k$-anonymity requirements. We notice that the results of IS-HL are slightly higher than those of on-demand and proactive algorithms (since some historical locations can be used in IS-HL) but a little lower than those of DA and EDA (since IS-HL is built on on-demand mode, lower than dual-active mode). We also notice that the results of EDA are slightly lower than those of DA. This is because EDA prevents some of the records with large hop from being further broadcasted, whereas the DA algorithm does not. So, DA can gather more anonymity.

Fig. 4b shows that the anonymizing time for each query decreases with the growth of user population. This is because a larger population density allows a user to gather enough anonymity more quickly. We notice that DA and EDA require less time than the others. This is because these two algorithms can actively share locations and use some stale records, making it quicker to meet their $k$-anonymity requirements.

As for the communication overheads, Fig. 4c shows that IS-HL achieves the best performance, followed by EDA, in terms of reducing communication overhead. We can see that DA costs the most for all scenarios while EDA costs only about 10% of DA's overhead, which proves the improvement of EDA over DA in terms of communication overhead.

In Fig. 4d, we find that the region sizes generated by different algorithms follow DA < on-demand = proactive < EDA < IS-HL. While generating regions, all of DA, EDA, and IS-HL can use some stale locations. However, DA does not consider user movement, and thus can use closer stale locations, resulting in a smaller region. In contrast, both EDA and IS-HL predict user movement and update new locations instead of the stale ones, and thus relatively few nearby locations are available, resulting in a larger region. Due to the different predicting techniques as described in Section 4.3.3, the size of IS-HL is slightly larger than that of EDA.

### 5.4 Impact of $k$-anonymity

Fig. 5 shows the performances of the algorithms with the increasing $k$-anonymity privacy requirements from [10, 40] to [40, 70]. Note that as the $k$-anonymity requirement gets stringent, all five algorithms have to gather more locations to generate CRs, which will decrease the success rate step by step (Fig. 5a). This requires more running time for all the algorithms (Fig. 5b) and larger region size (Fig. 5d).

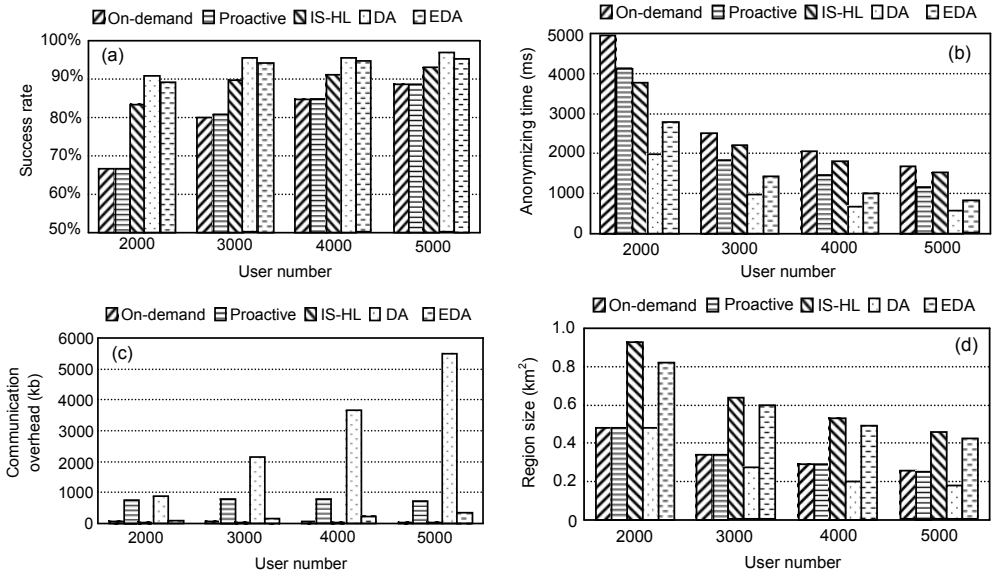


**Fig. 4 Performance evaluation against different number of users: (a) anonymization success rate; (b) average anonymizing time; (c) average communication overhead; (d) average region size**

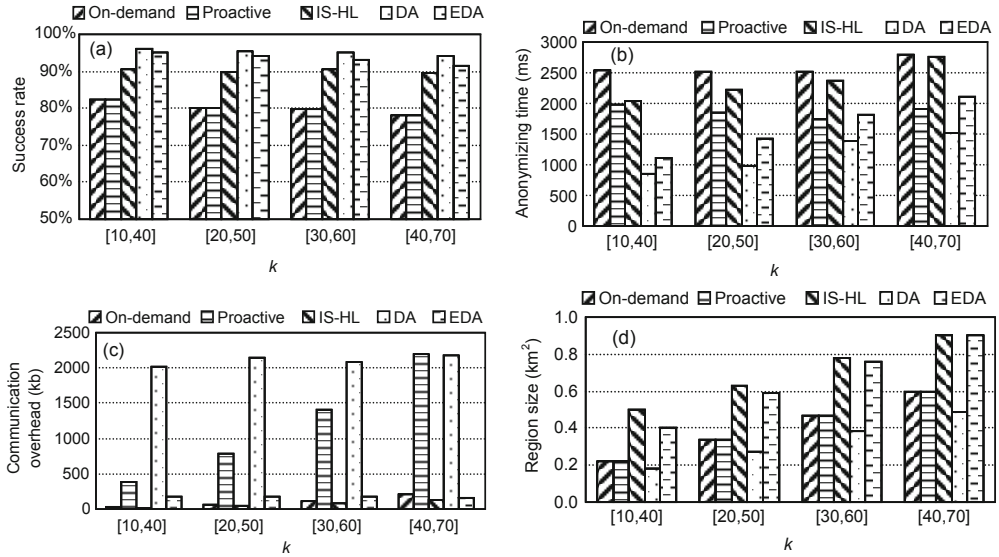


**Fig. 5 Performance evaluation against different $k$-anonymity: (a) anonymization success rate; (b) average anonymizing time; (c) average communication overhead; (d) average region size**

As a result, more communication overhead is required (Fig. 5c). The results also demonstrate that, although EDA has lower success rate and longer anonymizing time than DA, it costs much less (about only 10%) communication overhead. This also verifies Theorem 4.

### 5.5 Impact of privacy parameters

Fig. 6 shows the impact of parameter MaxAcceptHop, which is increased from 6 to 12. It is expected that as MaxAcceptHop increases, a user can send and receive more location records whose hop values are relatively large. As a result, the anonymization success rate increases and the anonymizing time decreases for both DA and EDA. However, more communication overhead is required to transmit these records with larger hop values. As expected, we notice that EDA has lower success rate and longer anonymizing time but less overhead than DA.

The following group of experiments demonstrates the influence of the parameter MaxValid-

Time, which grows from 4 to 10, on the three algorithms, IS-HL, DA, and EDA (Fig. 7). When MaxValidTime gets larger, we can infer that users can utilize more out-of-date locations for CR generation, which can increase the anonymization success rate and reduce the anonymizing time (Figs. 7a and 7b). Nevertheless, to transmit these records, communication overhead increases steadily with the increase of MaxValidTime (Fig. 7c). Moreover, the region size slightly increases with the MaxValidTime since users can use more locations that have a longer distance from themselves (Fig. 7d).

We also evaluate the performance of EDA with respect to parameter $p$, which affects the

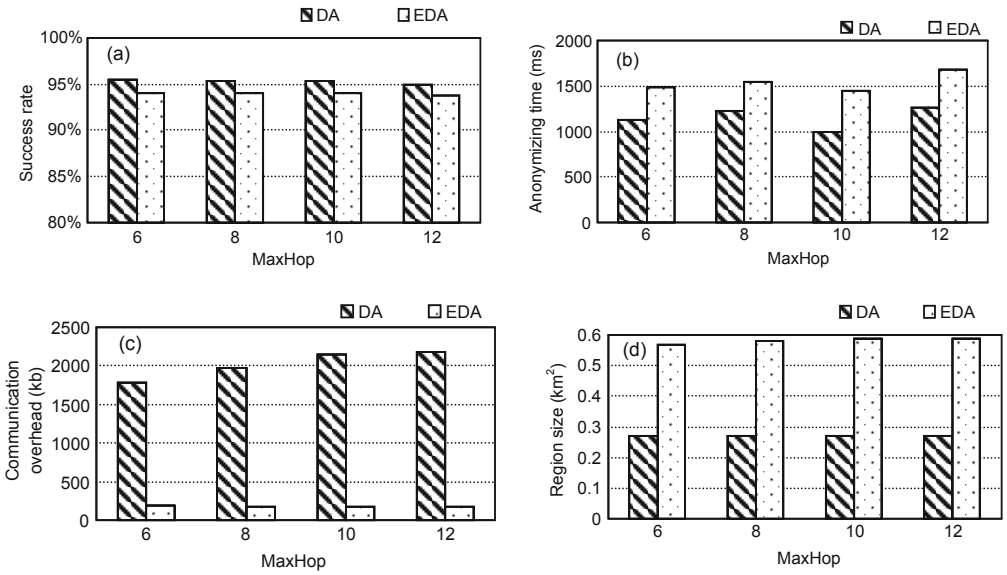


**Fig. 6 Performance evaluation against different MaxHop values: (a) anonymization success rate; (b) average anonymizing time; (c) average communication overhead; (d) average region size**

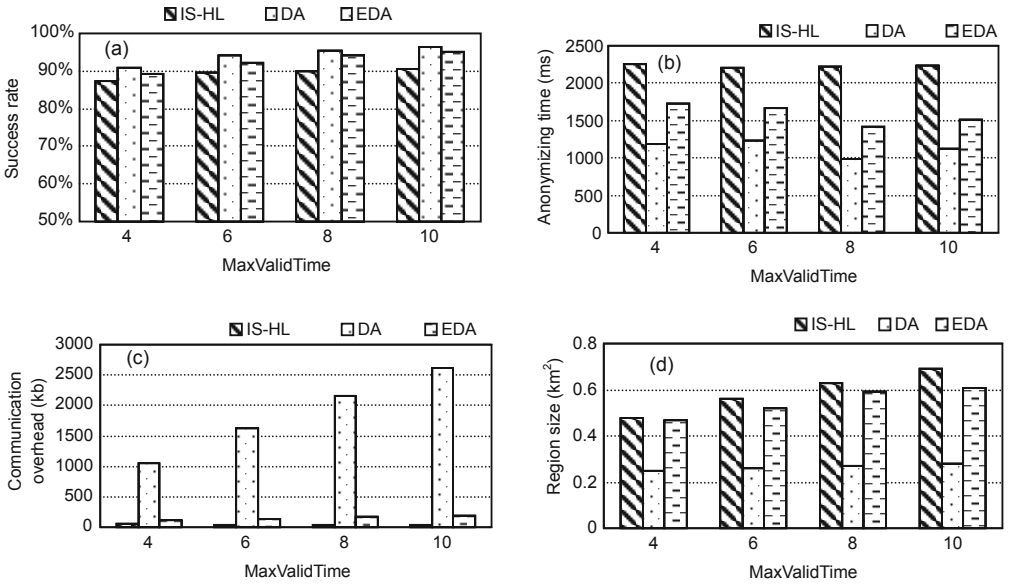


**Fig. 7 Performance evaluation against different MaxValidTime values: (a) anonymization success rate; (b) average anonymizing time; (c) average communication overhead; (d) average region size**

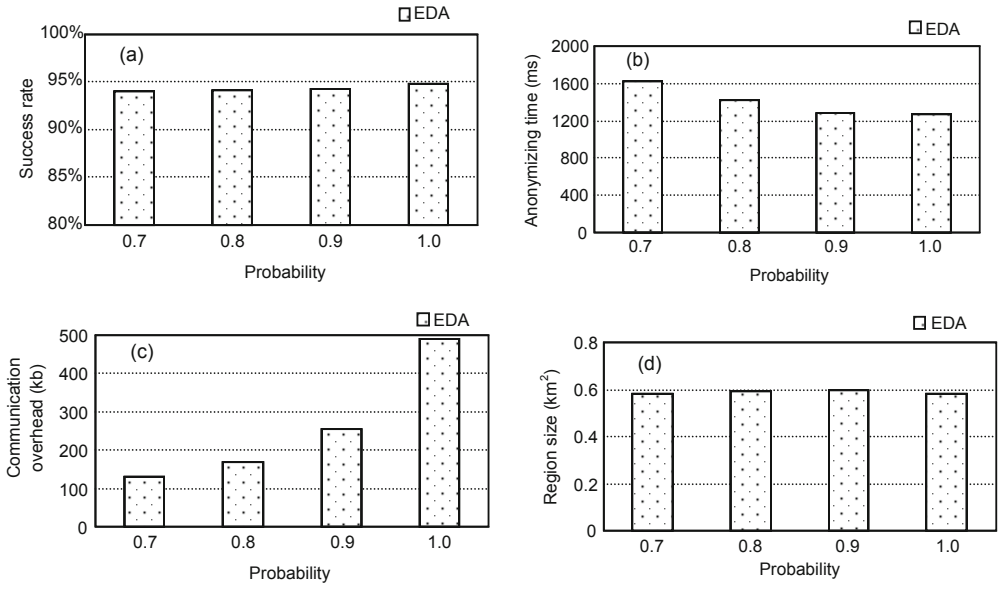


**Fig. 8 Performance evaluation against different probabilities: (a) anonymization success rate; (b) average anonymizing time; (c) average communication overhead; (d) average region size**

broadcasting probability for the LRHP strategy. In the experiment, we increase the value of $p$ from 0.7 to 1 (By Definition 2, $p = 1$ means that users will always broadcast every record to others). Fig. 8 shows the results. It is expected that as $p$ increases, the communication overhead increases (Fig. 8c).

## 6 Conclusions

In this paper we have proposed dual-active and enhanced dual-active algorithms for preserving users' location privacy while using LBS in a mobile P2P network environment. As compared with other solutions, we have made the following contributions:

1. With the LRHP strategy, EDA has been able to lower the probability of broadcasting useless location records and thus to reduce communication overhead and promoting the quality of anonymization.

2. EDA has been able to significantly reduce the overhead of communications among users by minimizing transmission of duplicate messages in the negotiation process.

3. With user-driven location prediction, EDA has been able to solve the inaccuracy problem, which is caused by using historical locations.

4. Experimental results have verified the effectiveness and efficiency of our EDA algorithm.

For the next stage of study, we plan to implement the proposed solutions on real systems and investigate new solutions for defending peer-side attacks and dealing with the challenging issues of mobile P2P environments.

## References

Bamba, B., Liu, L., Pesti, P., Wang, T., 2008. Supporting Anonymous Location Queries in Mobile Environments with PrivacyGrid. Proc. 17th Int. Conf. on World Wide Web, p.237-246. [doi:10.1145/1367497.1367531]

Brinkhoff, T., 2002. A framework for generating network-based moving objects. *GeoInformatica*, **6**(2):153-180. [doi:10.1023/A:1015231126594]

Che, Y., Chiew, K., Hong, X., He, Q., 2012a. SALS: Semantics-Aware Location Sharing Based on Cloaking Zone in Mobile Social Networks. Proc. 1st ACM SIGSPATIAL Int. Workshop on Mobile Geographic Information Systems.

Che, Y., Yang, Q., Hong, X., 2012b. A Dual-Active Spatial Cloaking Algorithm for Location Privacy Preserving in Mobile Peer-to-Peer Networks. Proc. IEEE Wireless Communications and Networking Conf., p.2098-2102. [doi:10.1109/WCNC.2012.6214137]

Chow, C.Y., Mokbel, M.F., 2009. Privacy in location-based services: a system architecture perspective. *SIGSPATIAL Spec.*, **1**(2):23-27. [doi:10.1145/1567253.1567258]

Chow, C.Y., Mokbel, M.F., Liu, X., 2006. A Peer-to-Peer Spatial Cloaking Algorithm for Anonymous Location-Based Service. Proc. 14th Annual ACM Int. Symp. on Advances in Geographic Information Systems, p.171-178. [doi:10.1145/1183471.1183500]

Chow, C.Y., Mokbel, M.F., Aref, W.G., 2009. Casper*: query processing for location services without

compromising privacy. *ACM Trans. Database Syst.*, **34**(4):1-45. [doi:10.1145/1620585.1620591]

Chow, C.Y., Mokbel, M.F., Liu, X., 2011. Spatial cloaking for anonymous location-based services in mobile peer-to-peer environments. *Geoinformatica*, **15**(2):351-380. [doi:10.1007/s10707-009-0099-y]

Gedik, B., Liu, L., 2005. Location Privacy in Mobile Systems: a Personalized Anonymization Model. Proc. 25th IEEE Int. Conf. on Distributed Computing Systems, p.620-629. [doi:10.1109/ICDCS.2005.48]

Ghinita, G., Kalnis, P., Skiadopoulos, S., 2007a. MobiHide: a Mobilea Peer-to-Peer System for Anonymous Location-Based Queries. Proc. 10th Int. Symp. on Advances in Spatial and Temporal Databases, p.221-238. [doi:10.1007/978-3-540-73540-3_13]

Ghinita, G., Kalnis, P., Skiadopoulos, S., 2007b. PRIVE: Anonymous Location-Based Queries in Distributed Mobile Systems. Proc. 16th Int. Conf. on World Wide Web, p.371-380. [doi:10.1145/1242572.1242623]

Gruteser, M., Grunwald, D., 2003. Anonymous Usage of Location-Based Services Through Spatial and Temporal Cloaking. Proc. 1st Int. Conf. on Mobile Systems, Applications and Services, p.31-42. [doi:10.1145/1066116.1189037]

Hong, J.I., Landay, J.A., 2004. An Architecture for Privacy-Sensitive Ubiquitous Computing. Proc. 2nd Int. Conf. on Mobile Systems, Applications, and Services, p.177-189. [doi:10.1145/990064.990087]

Internet Engineering Task Force (IETF), 2011. An Architecture for Location and Location Privacy in Internet Applications. Available from www.rfc-editor.org/rfc/rfc6280.txt

Kalnis, P., Ghinita, G., Mouratidis, K., Papadias, D., 2007. Preventing location-based identity inference in anonymous spatial queries. *IEEE Trans. Knowl. Data Eng.*, **19**(12):1719-1733. [doi:10.1109/TKDE.2007.190662]

Kathryn, Z., 2012. Three-Quarters of Smartphone Owners Use Location-Based Services. Technical Report, Pew Research Center's Internet & American Life Project, Pew Research Center.

Liao, J., Qi, Y.H., Huang, P.W., Rong, M.T., Li, S.H., 2006. Protection of mobile location privacy by using blind signature. *J. Zhejiang Univ.-Sci. A*, **7**(6):984-989. [doi:10.1631/jzus.2006.A0984]

Mokbel, M.F., Chow, C.Y., 2006. Challenges in Preserving Location Privacy in Peer-to-Peer Environments. Proc. 7th Int. Conf. on Web-Age Information Management Workshops. [doi:10.1109/WAIMW.2006.8]

Mokbel, M.F., Chow, C.Y., Aref, W.G., 2006. The New Casper: Query Processing for Location Services without Compromising Privacy. Proc. 32nd Int. Conf. on Very Large Data Bases, p.763-774.

Papadopouli, M., Schulzrinne, H., 2001. Effects of Power Conservation, Wireless Coverage and Cooperation on Data Dissemination among Mobile Devices. Proc. 2nd ACM Int. Symp. on Mobile Ad Hoc Networking and Computing, p.117-127. [doi:10.1145/501416.501433]

Shankar, P., Ganapathy, V., Iftode, L., 2009. Privately Querying Location-Based Services with SybilQuery. Proc. 11th Int. Conf. on Ubiquitous Computing, p.31-40. [doi:10.1145/1620545.1620550]

Shokri, R., Papadimitratos, P., Theodorakopoulos, G., Hubaux, J.P., 2011. Collaborative Location Privacy. Proc. IEEE 8th Int. Conf. on Mobile Adhoc and Sensor Systems, p.500-509. [doi:10.1109/MASS.2011.55]

Sweeney, L., 2002. $k$-Anonymity: a model for protecting privacy. *Int. J. Uncert. Fuzz. Knowl.-Based Syst.*, **10**(5):557-570. [doi:10.1142/S0218488502001648]

Tiwari, S., Kaushik, S., Jagwani, P., Tiwari, S., 2011. A Survey on LBS: System Architecture, Trends and Broad Research Areas. Proc. 7th Int. Workshop on Databases in Networked Information Systems, p.223-241. [doi:10.1007/978-3-642-25731-5_18]

Wei, W., Xu, F., Li, Q., 2012. MobiShare: Flexible Privacy-Preserving Location Sharing in Mobile Online Social Networks. INFOCOM, p.2616-2620. [doi:10.1109/INFCOM.2012.6195664]

Wu, X., Liu, J., Hong, X., Bertino, E., 2008. Anonymous geo-forwarding in manets through location cloaking. *IEEE Trans. Parall. Distr. Syst.*, **19**(10):1297-1309. [doi:10.1109/TPDS.2008.28]

Xiong, X., Mokbel, M.F., Aref, W.G., 2005. SEA-CNN: Scalable Processing of Continuous $k$-Nearest Neighbor Queries in Spatio-Temporal Databases. Proc. 21st Int. Conf. on Data Engineering, p.643-654. [doi:10.1109/ICDE.2005.128]

Yiu, M.L., Jensen, C.S., Huang, X., Lu, H., 2008. SpaceTwist: Managing the Trade-offs among Location Privacy, Query Performance, and Query Accuracy in Mobile Services. Proc. 24th Int. Conf. on Data Engineering, p.366-375. [doi:10.1109/ICDE.2008.4497445]

Zhang, C., Huang, Y., 2009. Cloaking locations for anonymous location based services: a hybrid approach. *GeoInformatica*, **13**(2):159-182. [doi:10.1007/s10707-008-0047-2]