RobLoP: Towards Robust Privacy Preserving Against Location Dependent Attacks in Continuous LBS Queries

Hongbo Jiang, *Senior Member, IEEE*, Ping Zhao, and Chen Wang, *Member, IEEE*

***Abstract*— With the increasing popularity of location-based services (LBS), how to preserve one's location privacy has become a key issue to be concerned. The commonly used approach k-anonymity, originally designed for protecting a user's *snapshot* location privacy, inherently fails to preserve the user from location-dependent attacks (LDA) that include the maximum movement boundary (MMB) attacks and maximum arrival boundary (MAB) attacks, when the user *continuously* requests LBS. This paper presents RobLoP, a robust location privacy preserving algorithm against LDA in continuous LBS queries. The key insight of RobLoP is to theoretically derive the constraints of both MMB and MAB in a uniform way. It provides a necessary condition of the pairwise user to be safely cloaked against LDA. On top of that, RobLoP first identifies those candidate users who can be cloaked with the requesting user. RobLoP then searches for a so-called strict point set including the candidate set and other auxiliary points, as a sufficient condition under which RobLoP can finally generate the cloaked region successfully. To the best of our knowledge, RobLoP is the first work that can preserve location privacy against LDA thoroughly and closely with a theoretical guarantee. The effectiveness and superiority of RobLoP to state-of-the-art studies are validated via extensive simulations on the real trucks data, the synthetic data, as well as the measured data collected by ourselves.**

***Index Terms*— Location privacy, k-anonymity, continuous LBS queries, location-dependent attacks.**

I. Introduction

With the current evolvement of mobile devices and wireless communication technologies, location-based services (LBS) have been enjoying growing popularity in recent years [1], [2], where mobile users can enjoy context-aware features (e.g., finding nearby restaurants from yelp, or monitoring real-time traffic from Google Maps) at all time and places, with their locations provided to the LBS server. On the downside, the location disclosure when LBS are provided often implies sensitive personal information such as one's life style or visited places, thereby raising severe privacy concerns [3]. Therefore, how to preserve location privacy has become a key issue in LBS [4]–[7].

In this paper, we focus on how to preserve location privacy against, when continuous LBS queries are issued with k-anonymity technique, the so-called location dependent attacks (LDA) [8]. In the following, we first briefly introduce k-anonymity and LDA, and then review existing k-anonymity-based privacy preserving methods against LDA, followed by our contributions.

A. k-Anonymity and LDA

k-Anonymity [9] employs a trusted anonymizer to blur a user's exact location into a large enough cloaked region (CR) geographically such that the user's location cannot be distinguished from at least $(k-1)$ other users in this CR. Beside the requirement of at least $(k-1)$ other users, k-anonymity always demands another important requirement, the so-called privacy area constraint denoted by $A_{\min}$. The smaller the area constraint is, the more vulnerable the query results are, because a random point in the query location region is closer to users' precise locations. That is why the k-anonymity mechanism requires that the area of the generated CR should be greater than $A_{\min}$ [10]–[12]. Fig. 1(a) shows an example of 2-anonymity, where two users u_1 and u_4 are cloaked in the CR $R_{u_1,t_{i+1}}$ (cf. the purple circle) at time t_{i+1}. In parallel, $\max(A_{u_1,\min}, A_{u_4,\min}) \leq A_{R_{u_1,t_i}}$ where $A_{R_{u_1,t_i}}$ is the area of the CR generated by the trusted anonymizer. Then LBS queries of u_1 and u_4 are finally aggregated and sent to the LBS server (achieving 2-anonymity) as shown in Fig. 1(a). It is possible that such a CR cannot be generated due to the requirements, and in that case, no query will be sent from the trusted anonymizer to the LBS server. Instead, the anonymizer gives back the negative messages in response to the users' requests.

One common limitation of most previous k-anonymity-based privacy preserving techniques [10], [11], [13], [14] is that they only deal with *snapshot* user locations (i.e., one-shot queries), while neglecting privacy disclosure when *continuous* LBS queries are launched (i.e., one's locations are continuously updated). Concretely speaking, as shown in Fig. 1(b), u_1 issues a LBS query at t_i (i.e., a one-shot query), and is cloaked with u_2 in CR R_{u_1,t_i}. Then u_1 issues another LBS query at t_{i+1}, and is cloaked with u_4. When the attacker has the prior

Manuscript received April 19, 2017; revised September 6, 2017, November 26, 2017, and February 3, 2018; accepted March 4, 2018; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor A. Kuzmanovic. Date of publication March 21, 2018; date of current version April 16, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant 61502192, Grant 61572219, Grant 61502193, Grant 61702204, and Grant 41701479, in part by the China Postdoctoral Science Foundation under Grant 2017T100556, and in part by the Fundamental Research Funds for the Central Universities under Grant 2016YXMS293 and Grant 2016JCTD118. *(Corresponding author: Chen Wang.)*

H. Jiang was with the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan 430074, China. He is now with the College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China (e-mail: hongbojiang2004@gmail.com).

P. Zhao and C. Wang are with the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan 430074, China (e-mail: pingzhao@hust.edu.cn; chenwang@hust.edu.cn).

Digital Object Identifier 10.1109/TNET.2018.2812851

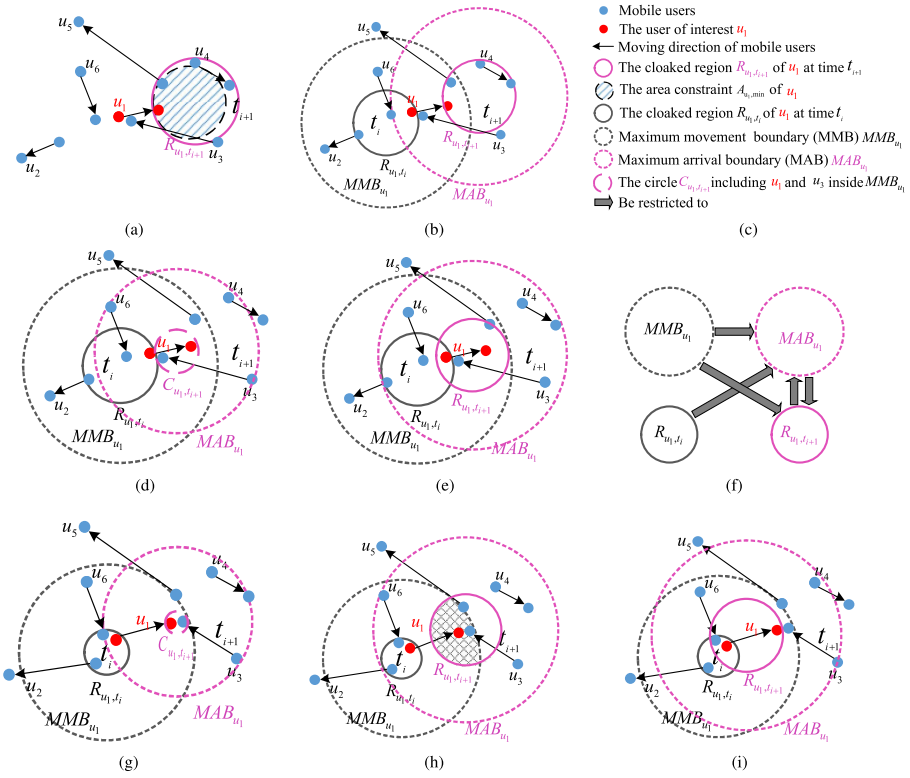

Fig. 1. Illustration of k-anonymity, LDA, challenges as well as solutions, $k = 2$. (d) (e) IClique successfully protects users' privacy. (g) (h) False-negative case in IClique. To be general, we assume the privacy area constraint $A_{u_1,\min}$ of u_1 is all the same. (a) An example of 2-anonymity. (b) An example of LDA. (c) Legend. (d) IClique against MMB attacks. (e) IClique against MAB attacks. (f) Constraints among MMB, MAB and CRs. (g) IClique against MAB attacks. (h) IClique against MAB attacks. (i) RobLoP against LDA.

knowledge of the last CR R_{u_1,t_i} and maximum movement speed v_{u_1} of the user u_1 (induced from various sources, e.g., speed-limit), he can infer u_1's maximum movement boundary (MMB, cf. the black dotted circle) MMB_{u_1} at t_{i+1} that extends R_{u_1,t_i} by a radius of $v_{u_1}(t_{i+1} - t_i)$, and further infer that u_1 must be located in the overlapped area of MMB and CR $R_{u_1,t_{i+1}}$ (cf. the small purple circle) at time t_{i+1}. Likewise, the attacker can infer u_1's maximum arrival boundary (MAB, cf. the purple dotted circle) MAB_{u_1} that extends $R_{u_1,t_{i+1}}$ by $v_{u_1}(t_{i+1} - t_i)$, and therefore deduce that u_1 must be located in the intersection region of MAB and CR R_{u_1,t_i} at time t_i. That is, u_1's location has been undesirably disclosed. The former is called MMB attacks and the latter is called MAB attacks. MMB and MAB attacks together are referred to as LDA [8].

Note that LDA, in a broad sense, can be regarded as spatio-temporal correlation attacks in the context of continuous LBS, which almost all existing location privacy preserving mechanisms may suffer from when continuous LBS queries are issued (detailed in Section IV-C). In our paper, LDA is referred to as, in a narrow sense, the spatio-temporal correlation attacks when location k-anonymity is used in continuous LBS as in the existing work [8].

B. Existing Efforts on Privacy Preserving Against LDA

MMB/MAB Constraints: To fight against MMB attacks, most studies exploited the practical so-called *MMB constraint*, where $R_{u_1,t_{i+1}} \subseteq MMB_{u_1}$, exemplified by Fig. 1(d) (denoted as $C_{u_1,t_{i+1}}$), which is a sufficient condition against MMB attacks [8], [15]–[18]. Likewise, the *MAB constraint* where $R_{u_1,t_i} \subseteq MAB_{u_1}$ exemplified by Fig. 1(e) is demanded for the resistance against MAB attacks [8].

MMB Attacks: Accordingly, a number of methods have strived to protect users' location privacy in response to MMB attacks. For instance, Cheng *et al.* [15] proposed to enlarge $R_{u_1,t_{i+1}}$ to cover R_{u_1,t_i}. Another solution in [16] strived to postpone the queries until the MMB grows large enough to fully contain $R_{u_1,t_{i+1}}$. Alternatively, Hashem *et al.* [17] proposed to generate the CR so that users are not in the overlapped region of the MMB and CR. Besides, Nguyen *et al.* [18] proposed to generate the CR contained inside the MMBs. Regardless, we surprisingly find few efforts in response to MAB attacks.

LDA Attacks: One attempt against MAB attacks is a follow-up study called IClique [8] performed in an ad hoc way. It considers against MMB attacks at first, by finding a circle $C_{u_1,t_{i+1}}$ (i.e., a candidate CR) including u_1 and no less than $(k-1)$ other users inside MMB (cf. the small purple dashed circle in Fig. 1(d)), satisfying the MMB constraint, similar to [18]. Next it turns to dealing with MAB attacks by gradually enlarging $C_{u_1,t_{i+1}}$ until its correlated MAB can cover R_{u_1,t_i} to satisfy the MAB constraint, which is shown in Fig. 1(e)).

Unfortunately, we capitalize that IClique with this ad hoc way leads to gratuitous "false-negative" cases. For example, when locations of u_1 and other users around are as shown in Fig. 1(g), the enlarged $C_{u_1,t_{i+1}}$ in Fig. 1(h) (i.e., $R_{u_1,t_{i+1}}$) cannot meet the u_1's area constraint $A_{u_1,\min}$ (i.e., the area of

the shadow area in Fig. 1(h) is less than $A_{u_1,\min}$). As such, u_1 cannot be cloaked by IClique. We reveal that this ad hoc way lacks *a thorough understanding of the inter-condition between MMB and MAB constraints*. As a matter of fact, u_1 could be cloaked (satisfying both MMB and MAB constraints) as long as having a proper CR as shown in Fig. 1(i) (we will elaborate the procedures to generate such a CR in Section III).

C. Challenges and Our Approach

In this paper we propose RobLoP, a Robust algorithm that can preserve Location Privacy against LDA, when continuous LBS queries are issued with k-anonymity. Our key idea is to deal with the inter-condition between MMB and MAB constraints in a uniform way, as mentioned in Section I-B. Though the basic idea sounds straightforward, we are facing the following challenges.

(1) It is non-trivial to identify at least $(k_{u_1}-1)$[1] other users who can be cloaked with u_1, considering both MMB and MAB constraints simultaneously, due to a chicken-and-egg problem: the specific privacy preserving requirements have to be deduced on the basis of the obtained $R_{u_1,t_{i+1}}$, while our aim is to generate $R_{u_1,t_{i+1}}$.

(2) It is not straightforward to generate the CR against LDA, even giving $(k_{u_1}-1)$ other users who can be cloaked with u_1, as there is no guarantee whether a circle covering those users is a valid CR or not.

To the best of our knowledge, RobLoP is the first work aiming at preserving location privacy against LDA in a uniform way. To address the first challenge, by the in-depth analysis deduced constraints among R_{u_1,t_i}, $R_{u_1,t_{i+1}}$, MMB, and MAB, RobLoP can avoid "false-negative" cases (cf. Theorem 1 in Section II-D). For the sake of practice we convert those constraints at time t_{i+1} to the constraints at time t_i (cf. Theorem 2 in Section III-A). As for the second challenge, we search for a so-called strict point set (cf. Section III-B), such that the smallest circle covering the strict point set can be used to generate a valid CR (cf. Section III-C). In addition, one benefit of considering the inter-condition between MMB and MAB constraints is that, according to Theorems 1 and 2, RobLoP is able to desirably mitigate the heavy load on the trusted anonymizer by excluding as many unqualified queries as possible (will be shown in Section V). As an intuitive example, u_3 in Fig. 1(i) at time t_i is far away from u_1, which implies that u_3's query should be excluded, and thus is not considered to be cloaked with u_1's query at time t_{i+1}.

The remainder of this paper is as follows. Section II gives an overview of our problem and solution. Section III describes the proposed algorithm, followed by some discussions in Section IV. Section V describes the performance evaluation. Finally Section VI concludes the paper.

II. Preliminary

A. System Architecture

The overall system architecture is shown in Fig. 2, which consists of three entities: a set of mobile users, the trusted anonymizer, and the LBS server.

[1]Here we assume the diversity that u_1 requests k_{u_1}-anonymity, u_2 requests k_{u_2}-anonymity, and so on.

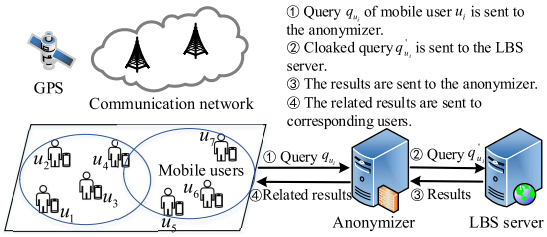

Fig. 2. System architecture.

Mobile Users: A user sends the LBS query $q = \{id, (x,y), (k, A_{\min}, d_t), C\}$ to the anonymizer, where id is the user's identity, (x,y) is the location of the user, C represents the query content, and $(k, A_{\min}, d_t)$ are privacy parameters. k means the CR must contain at least $(k-1)$ other users, $A_{\min}$ means the area of the CR should be larger than $A_{\min}$, and d_t is the maximum tolerable cloaking delay. Every user is allowed to set the parameters based on his own privacy preserving requirements.

Anonymizer: The anonymizer is assumed to be well-trusted, and the communication between users and the anonymizer is secure as in most prior studies [13], [19], [20].[2] Upon receiving the LBS query from a user (e.g., query q_{u_1} from u_1), the anonymizer seeks for no less than $(k_{u_1}-1)$ users around u_1, generates the CR where all users' location privacy is protected against LDA, and finally sends the cloaked/aggregated LBS query $q'_{u_1} = \{id, R, C\}$ to the LBS server.

LBS Server: The LBS server is generally considered semi-trusted [21], [22]. LBS server searches the results according to R and C, and sends them to the anonymizer. The anonymizer then forwards related results to the corresponding users.

Note that we mainly address those kinds of LBS applications where users send LBS queries containing their locations to the LBS server, and the LBS server searches results based on users' locations and returns results to users. For example, users query nearby restaurants from yelp, or request for real-time traffic from Google Maps, etc.

B. LDA Model

Consider any two successive queries of u_1 at time t_i and t_{i+1} ($i \in (1, 2, \ldots)$). A potential LDA adversary can be any party (e.g., other users or the LBS server), provided owing the following background information:

1) the centers and radiuses of the CRs R_{u_1,t_i} and $R_{u_1,t_{i+1}}$;
2) the maximum moving speed v_{u_1} of u_1 from t_i to t_{i+1}.

The attacker launches LDA by computing the intersection region of MMB and $R_{u_1,t_{i+1}}$ (resp. MAB and R_{u_1,t_i}) where u_1 is located at time t_{i+1} (resp. t_i). Specifically, either (1) $R_{u_1,t_{i+1}} \bigcap MMB_{u_1} \neq R_{u_1,t_{i+1}}$, or (2) $R_{u_1,t_i} \bigcap MAB_{u_1} \neq R_{u_1,t_i}$ will yield an LDA attack that could disclose the location privacy of u_1 [8], [15]. Take Fig. 1(b) for an instance, $R_{u_1,t_{i+1}} \nsubseteq MMB_{u_1}$, resulting in MMB attacks. Likewise, $R_{u_1,t_i} \nsubseteq MAB_{u_1}$, leading to MAB attacks.

Our aim is to prevent LDA adversaries from disclosing u_1's location privacy via narrowing u_1's CRs with the help of MMB

[2]This is practical as many existing protocols such as SSL and TLS can be adopted to guarantee the secure communication between users and anonymizer.

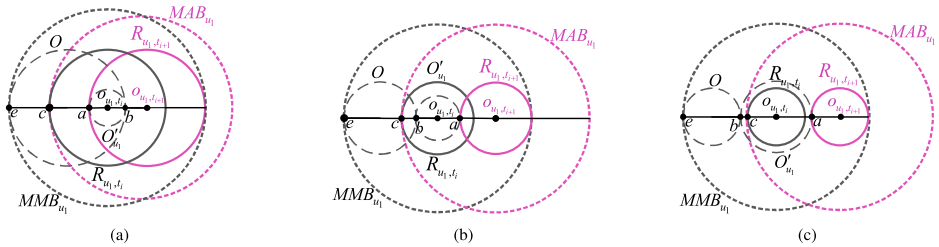

Fig. 3. Three cases of a specific user u_1, yielding three different privacy preserving requirements to make the CR locally safe. (a) Case 1. (b) Case 2. (c) Case 3.

and MAB. Note that here we only make a coarse-grained estimation of MMB and MAB ($v_{u_1}\Delta t_{u_1}$), without considering the particular constraint of road networks. The state-of-arts that study users' mobility in road networks [23], [24] can be directly applied to our work to obtain a more accurate estimation of the MMB and MAB, which is also validated in Section V-G.

C. Safe CR

Assume $u_1, u_2, \ldots, u_k$ requesting LBS are cloaked in a CR $R_{t_{i+1}}$ to form a k-anonymity at time t_{i+1}. According to the LDA model, we have:

Definition 1: $R_{t_{i+1}}$ is locally safe for a requesting user u_1, iff both (1) the MMB constraint $R_{t_{i+1}} \subseteq \mathit{MMB}_{u_1}$, and (2) the MAB constraint $R_{u_1,t_i} \subseteq \mathit{MAB}_{u_1}$, are satisfied.

Then to enable u_1 in $R_{t_{i+1}}$ to be indistinguishable from at least $(k_{u_1}-1)$ other users, it is required that all other users should also conform to both the MMB and MAB constraints. That is, we have:

Definition 2: $R_{t_{i+1}}$ is safe, iff for every user u_j ($j = 1, \ldots, k$), both $R_{t_{i+1}} \subseteq \mathit{MMB}_{u_j}$ and $R_{u_j,t_i} \subseteq \mathit{MAB}_{u_j}$ are satisfied. $R_{t_{i+1}}$ is called a safe CR.

Before obtaining a safe CR, at least $(k-1)$ qualified users should be identified in the first place. To that end, we explore in the following on what conditions a given CR is locally safe for a specific user u_1 (cf. Theorem 1), which is also the theoretical foundation leading to how RobLoP can practically identify the qualified users (cf. Theorem 2).

D. Privacy Preserving Requirements

Denote o_{u_1,t_i} (resp. $o_{u_1,t_{i+1}}$) and r_{u_1,t_i} (resp. $r_{u_1,t_{i+1}}$) the center and the radius of R_{u_1,t_i} (resp. $R_{u_1,t_{i+1}}$), and let $\Delta r_{u_1} = |r_{u_1,t_i} - r_{u_1,t_{i+1}}|$, $\Delta t_{u_1} = (t_{i+1} - t_i)$. We can derive the following:

Theorem 1: Given R_{u_1,t_i} (i.e., o_{u_1,t_i} and r_{u_1,t_i}), to ensure $R_{u_1,t_{i+1}}$ is locally safe for u_1, the following relations should be satisfied for $R_{u_1,t_{i+1}}$ (i.e., $o_{u_1,t_{i+1}}$ and $r_{u_1,t_{i+1}}$):

$$|o_{u_1,t_i}o_{u_1,t_{i+1}}| \le \min\{v_{u_1}\Delta t_{u_1} + \Delta r_{u_1}, v_{u_1}\Delta t_{u_1} - \Delta r_{u_1}\} \tag{1}$$

where $|\cdot|$ represents the Euclidean distance of two points.

Proof: According to the relationship between the maximum moving distance (i.e., $v_{u_1}\Delta t_{u_1}$) and r_{u_1,t_i}, there are three cases for a specific user u_1 (cf. Fig. 3 for instance). We consider privacy preserving requirements to make $R_{u_1,t_{i+1}}$ locally safe for u_1 in each case. For the ease of description, we introduce two auxiliary circles. The black dashed circle denoted by O which has the same size with R_{u_1,t_i}, is tangent to MMB_{u_1} at point e, and it intersects the lateral axis (at point b). The black dashed circle centering at o_{u_1,t_i} with the radius $|o_{u_1,t_i}b|$ is denoted by O'_{u_1}.[3] Based upon O and O'_{u_1}, we next show the privacy preserving requirements in each case. Without loss of generality, we assume that u_1 is moving right.

Case 1: $v_{u_1}\Delta t_{u_1} \le r_{u_1,t_i}$. That is, $o_{u_1,t_{i+1}}$ is within R_{u_1,t_i}, as shown in Fig. 3(a) where, apparently, $|ec| = |ca| = v_{u_1}\Delta t_{u_1}$. If $R_{u_1,t_{i+1}}$ is locally safe with respect to the MAB constraint, R_{u_1,t_i} should be in MAB_{u_1} (in Fig. 3(a), R_{u_1,t_i} is contained in MAB_{u_1}). To that end, $o_{u_1,t_{i+1}}$ can not be far away from o_{u_1,t_i}. Formally speaking, $|o_{u_1,t_i}o_{u_1,t_{i+1}}| \le r_{u_1,t_{i+1}} - |o_{u_1,t_i}a|$.

On the other hand, with respect to the MMB constraint, $R_{u_1,t_{i+1}}$ should be in MMB_{u_1} (in Fig. 3(a), $R_{u_1,t_{i+1}}$ should be within MMB_{u_1}). In this sense, the largest distance between $R_{u_1,t_{i+1}}$ and o_{u_1,t_i} should meet $|o_{u_1,t_i}o_{u_1,t_{i+1}}| + r_{u_1,t_{i+1}} \le r_{u_1,t_i} + v_{u_1}\Delta t_{u_1}$.

In summary, by geometric transformations, we can get

$$|o_{u_1,t_i}o_{u_1,t_{i+1}}| \le \min\{v_{u_1}\Delta t_{u_1} + \Delta r_{u_1}, v_{u_1}\Delta t_{u_1} - \Delta r_{u_1}\} \tag{2}$$

Case 2: $r_{u_1,t_i} \le v_{u_1}\Delta t_{u_1} \le 2r_{u_1,t_i}$. Fig. 3(b) shows an example where we have $|ec| = |ca| = v_{u_1}\Delta t_{u_1}$. Similar to Case 1, $R_{u_1,t_{i+1}}$ can not be far away from R_{u_1,t_i} such that $|o_{u_1,t_i}o_{u_1,t_{i+1}}| \le r_{u_1,t_{i+1}} + |o_{u_1,t_i}a|$.

With respect to the MMB constraint, $R_{u_1,t_{i+1}}$ should be in MMB_{u_1}, as shown in Fig. 3(b). Similar to Case 1, we have $|o_{u_1,t_i}o_{u_1,t_{i+1}}| + r_{u_1,t_{i+1}} \le r_{u_1,t_i} + v_{u_1}\Delta t_{u_1}$.

In summary, by geometric transformations, we can find

$$|o_{u_1,t_i}o_{u_1,t_{i+1}}| \le \min\{v_{u_1}\Delta t_{u_1} + \Delta r_{u_1}, v_{u_1}\Delta t_{u_1} - \Delta r_{u_1}\} \tag{3}$$

Case 3: $2r_{u_1,t_i} \le v_{u_1}\Delta t_{u_1}$. Please cf. Fig. 3(c) as an illustrative example. With the analysis similar to Cases 1 and 2, we have $|o_{u_1,t_i}o_{u_1,t_{i+1}}| + r_{u_1,t_{i+1}} \le r_{u_1,t_i} + v_{u_1}\Delta t_{u_1}$.

At last, by geometric transformations, we have

$$|o_{u_1,t_i}o_{u_1,t_{i+1}}| \le \min\{v_{u_1}\Delta t_{u_1} + \Delta r_{u_1}, v_{u_1}\Delta t_{u_1} - \Delta r_{u_1}\} \tag{4}$$

Combining (2) $\sim$ (4), Theorem 1 holds. ■

Theorem 1 presents the privacy preserving requirements of $R_{u_1,t_{i+1}}$ being locally safe for a specific user u_1, but these requirements cannot be utilized to identify other users who

[3]Hereafter we name O'_{u_1} as the center circle.

can be cloaked with u_1 in a safe CR, due to a chicken-and-egg problem. Specifically, the privacy preserving requirements are deduced on the assumption that $R_{u_1,t_{i+1}}$ has been obtained, while our aim is to generate $R_{u_1,t_{i+1}}$. To address this problem, we propose to convert the constraints on $R_{u_1,t_{i+1}}$ in Theorem 1 to the constraints on CR at time t_i of each user around u_1 (cf. Theorem 2), such that the candidate users cloaked together with u_1 can be identified practically (detailed in Section III-A).

III. RobLoP Algorithm

The basic idea behind RobLoP is straightforward: for a requesting user u_1, RobLoP strives to generate a safe CR which contains at least $(k_{u_1}-1)$ other users cloaked with u_1, and meanwhile meets the privacy parameter $A_{u_1,\min}$. To that end, RobLoP mainly consists of four steps:

(a) **Identifying Candidate User Sets**. The first step of RobLoP is to exploit the constraints in Theorem 1 of a given pairwise user if they can be safely cloaked against LDA. On top of that, RobLoP identifies those candidate users cloaked with the requesting user. These candidate users along with the requesting user are grouped to form the so-called candidate user set.

(b) **Searching for Strict Point Sets**. For the candidate user set, it is not sufficient to guarantee that the smallest circle bounded by it is a safe CR. To address this issue, we purposely search for a so-called strict point set including all users and some other auxiliary points such that the smallest circle bounded by this set is a safe CR.

(c) **Generating the Safe CR**. A provably safe CR can be finally generated via the circle covering the strict point set. It is noted that u_1 may belong to more than one candidate user set, yielding multiple safe CRs. In this case, the largest safe CR meeting the area constraint is selected as the final output.

(d) **Updating Sub-candidate User Sets**. After generating the safe CR, users with expired queries or who are successfully cloaked are then deleted from the sub-candidate user sets. By doing so, sub-candidate user sets can be updated timely while retaining intermediate results for subsequent queries.

A. Identifying Candidate User Sets

To find at least $(k_{u_1}-1)$ users cloaked with u_1, we elaborately convert the constraints with $R_{u_1,t_{i+1}}$ (at time t_{i+1}) in Theorem 1, to the constraints with R_{u_1,t_i} of each user cloaked with u_1, as presented in the following theorem.

Theorem 2: If two users denoted by u_1 and u_2 can be cloaked in one safe CR, we have

$$T_{u_1,u_2} \le v_{u_1}\Delta t_{u_1} - |r_{u_1,t_i} - r_{u_2,t_i}| + v_{u_2}\Delta t_{u_2} \quad (5)$$

$$2\max\{r_{u_1,t_i}, r_{u_2,t_i}\} \ge |u_{1,t_{i+1}}u_{2,t_{i+1}}| \quad (6)$$

where T_{u_1,u_2} is the distance between the centers of R_{u_1,t_i} and R_{u_2,t_i}, and $|u_{1,t_{i+1}}u_{2,t_{i+1}}|$ represents the distance between u_1 and u_2 at time t_{i+1}.

Proof: See the Appendix. ∎

Theorem 2 provides a practical means to exclude the user who can not be cloaked with u_1. Intuitively, if u_1 is far away from u_2 at time t_i, Formula (5) could be violated. Likewise, if u_1 is far away from u_2 at time t_{i+1}, Formula (6) could be violated. We next focus on how to exploit Theorem 2. To that end, we give:

Algorithm 1 Searching the Candidate User Set

Input sub-candidate user sets Ω_{sub}, and a new querying user u_i

Output candidate user sets Ω

1: find out these users (denoted by u_j) that meet Formulas (5) and (6), set $\mathcal{U}_i' \leftarrow u_j$
2: **for** each sub-candidate user set $\Theta_{sub,j}$ in Ω_{sub} **do** find user u_l, $u_l \in \mathcal{U}_i'$, and $u_l \in \Theta_{sub,j}$, then $\mathcal{C}_j \leftarrow u_l$. Last $\mathcal{C}_j \leftarrow u_i$
3: **for** each $\mathcal{C}_j$ **do**
4: **if** cardinality of $\mathcal{C}_j$, $V(\mathcal{C}_j) \ge k_{\max}$ **then** $\Omega \leftarrow \mathcal{C}_j$
5: **return** Ω

Definition 3: Assume a user set $\{u_1, u_2, \ldots, u_j\}$ where any two users satisfy both constraints of Formulas (5) and (6). If $k_{\max} = \max(k_{u_1}, k_{u_2}, \ldots, k_{u_j}) \le j$, this set is a candidate user set of u_1 denoted by Θ, otherwise is u_1's sub-candidate user set, denoted by Θ_{sub}.

Here any pair of users in the candidate user set Θ and sub-candidate user set Θ_{sub} should satisfy the constraints of Formulas (5) and (6), because u_1's location privacy may be disclosed if any other user's location privacy in Θ and Θ_{sub} is disclosed. In this sense, Θ and Θ_{sub} are also the candidate user set and sub-candidate user set for any other user therein. Note that, there may be more than one candidate user set and sub-candidate user set for u_1.

According to Definition 3, we can easily deduce that, if a user u_{l_1} with each of the users $u_1, u_2, \ldots, u_{l_2}$ in Θ_{sub} meet the constraints of Formulas (5) and (6), the set containing $u_1, u_2, \ldots, u_{l_2}, u_{l_1}$ is also a sub-candidate user set. On this basis, we propose Algorithm 1 to search the candidate user sets whenever a user (u_i) issues a new LBS query. The first step is to find these users each of which and u_i meet the constraints of Formulas (5) and (6) (Line 1). Then we try to find u_i's sub-candidate user sets (Line 2). Last we search for the candidate user sets according to $k_{\max}$ (Lines 3-4).

Take Fig. 4 as an example, $k_{u_j} = 3, j = (1, \ldots, 5)$. When u_1 queries LBS at time t_1, there is no user waiting to be cloaked (cf. Fig. 4(a)). So the sub-candidate user set $\Theta_{sub} = \{u_1\}$, and u_1 has to wait to be cloaked until its query is expired as the cardinality of Θ_{sub} is less than k_{u_1} (i.e., $V(\Theta_{sub}) < k_{u_1}$). Then another user ($u_2$) requests for LBS at t_2. We first identify whether u_2 and u_1 meet Formulas (5) and (6) (cf. Fig. 4(b)). If so, $\Theta_{sub} = \{u_1\}$ is updated to be $\Theta_{sub} = \{u_1, u_2\}$ (cf. Fig. 4(c)). As $V(\Theta_{sub}) < \max(k_{u_1}, k_{u_2})$, both u_1 and u_2 have to wait. Then another user u_3 queries LBS at t_3. We first check whether u_3 and u_1, u_2 meet the constraints in Formulas (5) and (6) (cf. Fig. 4(c)), and update $\Theta_{sub} = \{u_1, u_2\}$ to $\Omega_{sub} = (\{u_1, u_2\}, \{u_1, u_3\})$ where $\Theta_{sub,1} = \{u_1, u_2\}$ and $\Theta_{sub,2} = \{u_1, u_3\}$. Similarly, as $V(\Theta_{sub,2}) < \max(k_{u_1}, k_{u_3})$, u_1, u_2 and u_3 have to wait. Thereafter u_4 queries LBS at t_4 (cf. Fig. 4(d)). We first identify whether u_4 and u_1, u_2, u_3 meet Formulas (5) and (6), and update Ω_{sub} to $\Omega_{sub} = (\{u_1, u_2\}, \{u_1, u_3, u_4\})$ $\Theta_{sub,1} = \{u_1, u_2\}$ and $\Theta_{sub,2} = \{u_1, u_3, u_4\}$. Since $V(\Theta_{sub,2}) \ge \max(k_{u_1}, k_{u_3}, k_{u_4})$, so set $\{u_1, u_3, u_4\}$ is a candidate user set

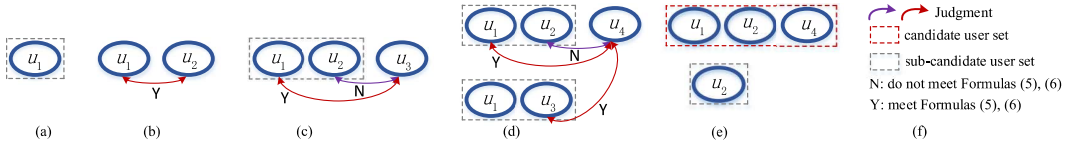

Fig. 4. Illustration of Algorithm 1, with $k_{u_j} = 3, j = (1, 2, \ldots, 5)$. (a) $\Theta_{sub} = \{u_1\}$ at time t_1. (b) $\Theta_{sub} = \{u_1, u_2\}$ at time t_2. (c) $\Omega_{sub} = (\{u_1, u_2\}, \{u_1, u_3\})$ at time t_3. (d) $\Omega_{sub} = (\{u_1, u_2\}, \{u_1, u_3, u_4\})$ at time t_4. (e) $\Theta_{sub} = \{u_2\}, \Theta = \{u_1, u_3, u_4\}$.

of u_1, u_3, and u_4, and we further update Ω_{sub} to $\Omega_{sub} = (\{u_2\})$ (cf. Fig. 4(e)).

Next, we study the time complexity of Algorithm 1. Denote the number of sub-candidate user sets in Ω_{sub} by $N_o(\Omega_{sub})$, and the number of sets $\mathcal{C}_j$ by $N_s(\mathcal{C}_j)$. The two for-loop iterations in Line 2, and Lines 3-4 incur $O(N_o(\Omega_{sub}))$, $O(N_s(\mathcal{C}_j))$ computation cost. In Line 1, the algorithm has to search for the qualified users from all the querying users at the worst case. Thus Line 1 incurs $O(N)$ time complexity at most (N is the number of querying users). In summary, the time complexity of Algorithm 1 is $O(N)$, since $N_o(\Omega_{sub}) \ll N$ and $N_s(\mathcal{C}_j) \ll N$.

B. Searching for Strict Point Sets

The candidate user set guarantees that a safe CR covering this set exists, but the smallest circle bounded by the set may not be a safe CR. Please cf. Fig. 16(a) as an illustrative example. We assume that the privacy requirement of u_1 and u_2 is $k_{\max} = \max(k_{u_1}, k_{u_2}) = 2$, and $\Theta = \{u_1, u_2\}$ is a candidate user set. According to Theorem 1, the safe CR for u_1 and u_2 at time t_{i+1}, $R_{u_1,u_2,t_{i+1}}$ should meet: $R_{u_1,u_2,t_{i+1}} \cap O'_{u_1} \neq$, and $R_{u_1,u_2,t_{i+1}} \cap O'_{u_2} \neq$ (O'_{u_1} and O'_{u_2} are center circles). However, the smallest circle bounded by locations of u_1 and u_2 at time t_{i+1} obviously does not intersect with O'_{u_1}, and thus is not a safe CR.

Before digging into the safe CRs, in this subsection, we focus on searching for the so-called strict point set based on which the safe CR can be generated. Briefly speaking, this point set is the vertex set of a convex hull where the convex hull of a plane point set is the smallest convex polygon that covers all the points in the point set [25], [26]. It has been proved, in [25] and [26], that there is one and only one convex hull for a given plane point set.

For the candidate user set, all locations constitute a plane point set and there exists one corresponding convex hull. As above-mentioned, it is not sufficient to guarantee that a circle covering all candidate users within it can be a safe CR. To address this problem, we propose the strict point set (derived from an updated convex hull) such that the circle covering the set is locally safe. Recall that in the proof of Theorem 1, in Case 1, the safe CR should contain the center circle O'_{u_1}, and in Cases 2 and 3, the safe CR must intersect with O'_{u_1}. Accordingly, we define the strict point set as follows.

Definition 4: Assume a candidate user set $\Theta = \{u_1, u_2, \ldots, u_n\}$, with the corresponding plane point set and convex hull $\mho$ and H. To be general, we assume $v_{u_j}\Delta t_{u_j} \leq r_{u_j,t_i}$ ($j \in (1, \ldots, l)$, $l < n$) (i.e., Case 1), and $v_{u_j}\Delta t_{u_j} > r_{u_j,t_i}$ ($j \in (l+1, \ldots, n)$) (i.e., Cases 2 and 3). Then, the Strict Point Set Υ is the set of the vertexes on convex hull H updated with the following point p:

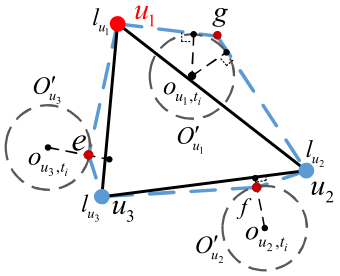

Fig. 5. Searching for the strict point set.

$$\forall \text{ user } u_j,\ j \in (1, \ldots, l)$$
$$|po_{u_j,t_i}| = r_{u_j,t_i} - v_{u_j}\Delta t_{u_j} \tag{7}$$
$$\forall \text{ user } u_j,\ j \in (l+1, \ldots, n)$$
$$|po_{u_j,t_i}| = r_{u_j,t_i} - (2r_{u_j,t_i} - v_{u_j}\Delta t_{u_j})$$
$$\arg\min |pH| \tag{8}$$

where $|pH|$ is the minimum distance between p and H.

Intuitively, points on the center circle O'_{u_j} in case of Case 1 (defined in Formula (7)) and the point on the center circle O'_{u_j} nearest to the convex hull H (defined in Formula (8)) are added to the plane point set $\mho$ to update the convex hull H. Then the vertices of the updated convex hull H is the strict point set. Please cf. Fig. 5 as an illustrative example, where u_1, u_2, u_3 constitute a candidate user set, with their locations as a plane point set $\mho = \{l_{u_1}, l_{u_3}, l_{u_2}\}$. The convex hull of $\mho$ is H=$\{\overrightarrow{l_{u_1}l_{u_3}}, \overrightarrow{l_{u_3}l_{u_2}}, \overrightarrow{l_{u_2}l_{u_1}}\}$. Assume $0 < v_{u_1}\Delta t_{u_1} \leq r_{u_1,t_i}$, $r_{u_2,t_i} \leq v_{u_2}\Delta t_{u_2} \leq 2r_{u_2,t_i}$, $2r_{u_3,t_i} \leq v_{u_3}\Delta t_{u_3}$. Points e and f on O'_{u_3} and O'_{u_2} (i.e., the $|o_{u_3,t_i}e|$-radius (resp. $|o_{u_2,t_i}f|$-radius) circle centering at o_{u_3,t_i} (resp. o_{u_2,t_i}) in Fig. 5) are nearest ones to H. As such, e, f are added to plane point set $\mho$. Similarly, points on center circle O'_{u_1} ($|o_{u_1,t_i}g|$-radius circle centering at o_{u_1,t_i}) are also added to $\mho$. Thereafter, we accordingly update H with e, f, and points on center circle O'_{u_1}, and the updated convex hull is $H = \{\overrightarrow{el_{u_3}}, \overrightarrow{l_{u_3}f}, \overrightarrow{fl_{u_2}}, \overrightarrow{l_{u_2}g}, \overrightarrow{gl_{u_1}}, \overrightarrow{l_{u_1}e}\}$. As such, the strict point set is $\Upsilon = \{l_{u_1}, l_{u_3}, l_{u_2}, e, f, g\}$.

C. Generating the Safe CR

In this subsection, we focus on generating the safe CR from a given strict point set, as elaborated in Algorithm 2.

Theorem 3: Given a strict point set Υ, there exists only one minimum circle C covering Υ. Besides, if there exist only two points p_{τ_1} and p_{τ_2} ($\tau_1 \neq \tau_2$) on the boundary of C, the diameter of C, $d_C = |p_{\tau_1}p_{\tau_2}|$; If there exist only three points p_{τ_1}, p_{τ_2} and p_{τ_3} ($\tau_1 \neq \tau_2 \neq \tau_3$) on the boundary of C, the triangle $\triangle p_{\tau_1}p_{\tau_2}p_{\tau_3}$ is not an obtuse triangle.

Proof: 1) We first prove there exists only one minimum circle covering the strict point set Υ.

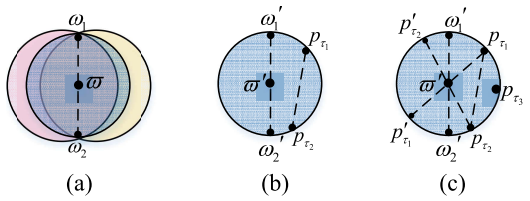

Fig. 6. Illustrative examples for (a) the existence of the minimum circle, (b) the case where only two points on the minimum circle, and (c) the case where only three points on the minimum circle.

Assume there are two minimum circles covering Υ (the pink and yellow circles in Fig. 6(a)), and they intersect at ω_1 and ω_2. So a $\frac{|\omega_1\omega_2|}{2}$-radius circle centering at point ϖ (denoted by C) can cover the overlapped region of the two minimum circles, and thus cover the strict point set Υ. Obviously, the circle C is smaller than the two minimum circles. Therefore, there is only one minimum circle C covering Υ.

2) Next, we turn to prove when only two or three points in Υ are on C's circumference, these points satisfy the constraints defined in this theorem. Denote the $\frac{|\omega_1'\omega_2'|}{2}$-radius circle centering at point ϖ' as C' (cf. Fig. 6(b)).

(a) Only two points p_{τ_1} and p_{τ_2} in Υ are on C''s circumference. If $|p_{\tau_1}p_{\tau_2}| \neq |\omega_1'\omega_2'|$ (cf. Fig. 6(b)), C' can be narrowed in the direction that is perpendicular to the line $\overline{p_{\tau_1}p_{\tau_2}}$. Thus, C' is not the minimum circle (i.e., $C' \neq C$). If $|p_{\tau_1}p_{\tau_2}| = |\omega_1'\omega_2'|$, C' cannot be narrowed and thus is the minimum circle (i.e., $C' = C$). In summary, when only two points p_{τ_1} and p_{τ_2} in Υ are on the circumference of the minimum circle C, the diameter of C meets $d_C = |p_{\tau_1}p_{\tau_2}|$.

(b) Only three points p_{τ_1}, p_{τ_2} and p_{τ_3} in Υ are on C''s circumference. If $|p_{\tau_1}p_{\tau_2}| = |\omega_1'\omega_2'|$, according to case (a) above, C' is the minimum circle (i.e., $C' = C$), and $\triangle p_{\tau_1}p_{\tau_2}p_{\tau_3}$ is a right triangle. Otherwise (cf. Fig. 6(c)), (i) if $p_{\tau_3} \in$ arc $\overset{\frown}{p'_{\tau_1}p_{\tau_2}p'_{\tau_2}}$, C' can be narrowed in the direction that is perpendicular to the line $\overline{p'_{\tau_1}p'_{\tau_2}}$, and thus $C' \neq C$; (ii) if $p_{\tau_3} \in$ arc $\overset{\frown}{p'_{\tau_1}p'_{\tau_2}}$, C' cannot be narrowed (i.e., $C' = C$), and $\triangle p_{\tau_1}p_{\tau_2}p_{\tau_3}$ is an acute triangle. In summary, when $C' = C$, $\triangle p_{\tau_1}p_{\tau_2}p_{\tau_3}$ is not an obtuse triangle.

In summary, Theorem 3 holds. ■

Theorem 3 shows that the minimum circle covering the strict point set is determined by the most distant two points, or the three points consisting an acute-angled triangle.

Moreover, according to the definition of safe CR (cf. Definition 2), we have:

Theorem 4: The minimum circle covering each strict point set is a safe CR.

Proof: Assume a candidate user set $\Theta = \{u_1, u_2, \ldots, u_n\}$, the strict point set for Θ is Υ, and the corresponding minimum circle is C. According to the definition of safe CR, to prove that C is a safe CR, we will prove C satisfies both MMB and MAB constraints simultaneously. Namely, for $\forall$ user u_j ($j \in (1, \ldots, n)$), $C \subseteq$ ***MMB***$_{u_j}$ and $R_{u_j,t_i} \subseteq$ ***MAB***$_{u_j}$.

(1) First, we prove C satisfies MMB constraint, i.e., for $\forall$ user u_j ($j \in (1, \ldots, n)$), $C \subseteq$ ***MMB***$_{u_j}$.

(a) The locations of users $u_1, u_2, \ldots, u_n$, according to Definition 3, are in ***MMB***$_{u_j}$ ($j \in (1, \ldots, n)$).

(b) The strict point set for Θ is obtained through adding points on the center circles to the set of users' locations (cf. Definition 4), and these points are in ***MMB***$_{u_j}$, ($j \in (1, \ldots, n)$) (proved in Theorem 2).

Thus, combing (a) and (b), points in the strict point set Υ are in ***MMB***$_{u_j}$ ($j \in (1, \ldots, n)$). Furthermore, the largest distance from o_{u_j,t_i} (i.e., the center of ***MMB***$_{u_j}$) to the minimum circle C constricted by Υ is $(r_{u_j,t_i} + v_{u_j}\Delta t_{u_j})$ (i.e., the radius of ***MMB***$_{u_j}$). Therefore, C satisfies MMB constraint.

(2) Second, we prove C satisfies MAB constraint, i.e., for $\forall$ user u_j ($j \in (1, \ldots, n)$), $R_{u_j,t_i} \subseteq$ ***MAB***$_{u_j}$.

(a) According to Theorem 1, to satisfy the MAB constraint, in Case 1, the CR should include the center circle O'_{u_1}, and in Cases 2 and 3, the safe CR must intersect with O'_{u_1}.

(b) According to Definition 4, at least one point on the center circle O'_{u_1} in Case 1 and all points on the center circle O'_{u_1} in Cases 2 and 3 are included in the strict point set Υ, thereby in the minimum circle C.

In summary, the corresponding MAB for C includes R_{u_j,t_i}, ($j \in (1, \ldots, n)$). Namely, C satisfies MAB constraint.

To sum up, Theorem 4 holds. ■

Algorithm 2 depicts the main idea of generating safe CR. In Algorithm 2, Line 2 dominates the computation cost, incurring $O(N_o(\Upsilon))$ cost ($N_o(\Upsilon)$ is the number of points in Υ).

Algorithm 2 Generating the Safe CR

Input Υ

Output the minimum circle C (radius C_r, center C_{cen})

1: find out two points $p_{\hat{\tau_1}}, p_{\hat{\tau_2}}$, such that $|p_{\hat{\tau_1}}p_{\hat{\tau_2}}| = \arg\max\{|p_{\tau_1}p_{\tau_2}|\}$, then $C_r = |p_{\hat{\tau_1}}p_{\hat{\tau_2}}|/2$, $C_{cen} = ((p_{\hat{\tau_1}}.x + p_{\hat{\tau_2}}.x)/2, (p_{\hat{\tau_1}}.y + p_{\hat{\tau_2}}.y)/2)$.

2: **if** $p_j \notin C$ **then**

3: find out three points $p_{\hat{\tau_1}}$, $p_{\hat{\tau_2}}$, $p_{\hat{\tau_3}}$, such that $\triangle p_{\hat{\tau_1}}p_{\hat{\tau_2}}p_{\hat{\tau_3}}$ is an acute-angled triangle, $\forall\ p_i \in \Upsilon$, then $C_r = |p_{\hat{\tau_1}}p_{\hat{\tau_2}}||p_{\hat{\tau_1}}p_{\hat{\tau_3}}||p_{\hat{\tau_3}}p_{\hat{\tau_2}}|/\Gamma(p_{\hat{\tau_1}}, p_{\hat{\tau_2}}, p_{\hat{\tau_3}})$[3], $C_{cen} = ((p_{\hat{\tau_1}}.x+p_{\hat{\tau_2}}.x+p_{\hat{\tau_3}}.x)/3, (p_{\hat{\tau_1}}.y+p_{\hat{\tau_2}}.y+p_{\hat{\tau_3}}.y)/3)$.

4: **return** C

Note that there may be several safe CRs for the requesting user u_1, since there may be several candidate user sets of u_1. Each of the obtained safe CRs above only satisfies $k_{\max}$, but may not meet $\max(A_{\min})$.[4] Thus, next we focus on selecting one safe CR that meets both $k_{\max}$ and $\max(A_{\min})$ as the final output. When we output the CR with maximum area, more users' location privacy can be protected. But the quality of LBS (QoS) degrades with the size of CR. Thus it is a trade-off between QoS and the privacy preservation, and both the CR with minimal area and the CR with maximum area can be selected as the final output, with their respective pros and cons. In the following, we select the CR with maximum area as the final output, to provide stronger privacy preservation.

Specifically, (1) if there exists more than one safe CR whose area (denoted by $Area$) is larger than $\max(A_{\min})$, we select the safe CR with maximum area as the final output; (2) if not, then in each safe CR whose $Area$ larger than u_1's $A_{\min}$, we delete the users whose $A_{\min} > Area$ one by one until the updated $\max(A_{\min})$ less than the updated $Area$. If there

[3] $\Gamma(a,b,c) = \sqrt{(a+b+c)(a+b-c)(a-b+c)(-a+b+c)}$

[4] Assume users u_1, u_2, …, u_n are cloaked in one CR, and then $\max(A_{\min}) = \max\{A_{u_1,\min}, A_{u_2,\min}, \ldots, A_{u_n,\min}\}$.

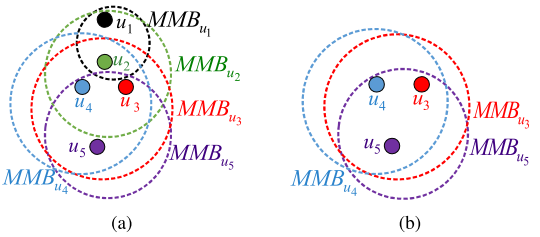

Fig. 7. Illustration of updating candidate user sets. (a) Before updating. (b) After updating.

Algorithm 3 Update Sub-Candidate User Sets

Input sub-candidate user sets Ω_{sub}, and a successfully cloaked user u_i
Output sub-candidate user sets Ω_{sub}
1: **for** each $\Theta_{sub,j} \subset \Omega_{sub}$ **do**
2: **if** $u_i \in \Theta_{sub,j}$ **then** update $\Theta_{sub,j}$ by removing u_i
3: **for** each updated $\Theta_{sub,j}$ **do**
4: **for** each updated $\Theta_{sub,j'} (j' \neq j)$ **do**
5: **if** $\Theta_{sub,j'} \subseteq \Theta_{sub,j}$ **then** remove $\Theta_{sub,j'}$ from Ω_{sub}
6: **if** $\Theta_{sub,j} \subseteq \Theta_{sub,j'}$ **then** remove $\Theta_{sub,j}$ from Ω_{sub}
7: **return** Ω_{sub}

exists one or more safe CR whose cardinality $vol \geq k_{\max}$, we select the safe CR with maximum area as the final output; (3) otherwise, in each updated safe CR whose cardinality $k_{u_1} < vol < k_{\max}$, we remove the users whose $k > vol$ one by one until $k_{u_1} \leq k_{\max} \leq vol$, and repeat step (1). Note that if the cardinality (resp. the area) of each safe CR is less than u_1's k_{u_1} (resp. $A_{\min}$), all users in these safe CRs are not successfully cloaked, and have to wait for being cloaked before their queries are expired.

D. Updating Sub-Candidate User Sets

Whenever users are successfully cloaked or users' queries are expired, these users will be deleted from the anonymizer. Deleting users means removing the area belonging to MMBs of these users from the overlapped area of MMBs of all users in a sub-candidate user set. For example, in Fig. 7(a), assume users u_1, u_2 are in a candidate user set $S_{2,1}$, and u_2, u_3, u_4 (resp. u_3, u_4, u_5) in a sub-candidate user set $S_{2,2}$ (resp. $S_{2,3}$). When u_1, u_2 are successfully cloaked, or queries of u_1 and u_2 are expired, $S_{2,1}$ will be deleted; u_2 will be deleted from $S_{2,2}$ (i.e., the area belonging to the MMB of u_2 will be removed from the overlapped area of MMBs of u_2, u_3, u_4). So, $S_{2,2}$ will be updated to the one including u_3, u_4, and the overlapped area will be updated to the intersection region of MMBs of u_3,u_4. Since the updated overlapped area covers the overlapped area of MMBs of u_3, u_4, u_5, thus the updated $S_{2,2}$ will be deleted, and only one sub-candidate user set $S_{2,3}$ remains, as shown in Fig. 7(b).

The main idea of this step is shown in Algorithm 3. The first for-loop iteration in Line 1 incurs $O(N_o(\Omega_{sub}))$ computation cost. In Lines 3-6, the time complexity is $O(N_o(\Omega_{sub,u_i})^2)$, where Ω_{sub,u_i} is the set of the sub-candidate user sets involving u_i. Since $N_o(\Omega_{sub,u_i}) < N_o(\Omega_{sub})$, the worst-case time complexity of Algorithm 3 is $O(N_o^2(\Omega_{sub}))$.

IV. Discussions

A. Privacy Preservation

We first theoretically prove the privacy preservation provided in RobLoP, and we get the following results.

Theorem 5: RobLoP can protect users' location privacy against LDA over the entire paths of users.

Proof: The generated CR $R_{u_1,t_{i+1}}$ is proved to be a safe CR in Theorem 4, when we consider any two successive queries of the user u_1 at time t_{i+1} and t_i. Therefore, we only need to prove CR $R_{u_1,t_{i+1}}$ is also a safe CR even when we consider u_1's queries at time t_{i+1} and any a timestamp, e.g., t_{i-3}.

Assume u_1 issues LBS queries at time $t_0, t_1, \ldots, t_i, t_{i+1}, \ldots, t_n$. To be general, we consider two queries of u_1 at time t_{i+1} and $t_{i-\kappa}$, $\kappa \in (0, 1, 2, \ldots, i-1)$.

According to Theorem 4, the CRs generated in RobLoP $R_{u_1,t_1}, R_{u_1,t_2}, \ldots, R_{u_1,t_n}$ are safe CRs when we consider u_1's queries at time t_0 and t_1, t_1 and t_2, $\ldots$, $\ldots$, t_{n-1} and t_n separately. Thus according to the definition of safe CR (cf. Definition 2), the following holds to meet MMB and MAB constraints:

$$\begin{cases} r_{u_1,t_i} + v_{u_1}(t_{i+1} - t_i) \geq r_{u_1,t_{i+1}} + \mid o_{u_1,t_{i+1}} o_{u_1,t_i} \mid \\ r_{u_1,t_{i-1}} + v_{u_1}(t_i - t_{i-1}) \geq r_{u_1,t_i} + \mid o_{u_1,t_i} o_{u_1,t_{i-1}} \mid \\ \vdots \\ r_{u_1,t_{i-\kappa}} + v_{u_1}(t_{i-\kappa+1} - t_{i-\kappa}) \\ \quad \geq r_{u_1,t_{i-\kappa+1}} + \mid o_{u_1,t_{i-\kappa+1}} o_{u_1,t_{i-\kappa}} \mid \end{cases} \tag{9}$$

$$\begin{cases} r_{u_1,t_{i+1}} + v_{u_1}(t_{i+1} - t_i) \geq r_{u_1,t_i} + \mid o_{u_1,t_{i+1}} o_{u_1,t_i} \mid \\ r_{u_1,t_i} + v_{u_1}(t_i - t_{i-1}) \geq r_{u_1,t_{i-1}} + \mid o_{u_1,t_i} o_{u_1,t_{i-1}} \mid \\ \vdots \\ r_{u_1,t_{i-\kappa+1}} + v_{u_1}(t_{i-\kappa+1} - t_{i-\kappa}) \\ \quad \geq r_{u_1,t_{i-\kappa}} + \mid o_{u_1,t_{i-\kappa+1}} o_{u_1,t_{i-\kappa}} \mid \end{cases} \tag{10}$$

Adding the $(\kappa+1)$ formulas in Formula (9) up, we have

$$r_{u_1,t_{i-\kappa}} + v_{u_1}(t_{i+1} - t_{i-\kappa}) \geq r_{u_1,t_{i+1}} + \mid o_{u_1,t_{i+1}} o_{u_1,t_{i-\kappa}} \mid \tag{11}$$

Namely, the CR $R_{u_1,t_{i+1}}$ meets the MMB constraint when we consider two queries at time t_{i+1} and $t_{i-\kappa}$.

Similarly, adding the $(\kappa+1)$ formulas in Formula (10) up, we can get

$$r_{u_1,t_{i+1}} + v_{u_1}(t_{i+1} - t_{i-\kappa}) \geq r_{u_1,t_{i-\kappa}} + \mid o_{u_1,t_{i+1}} o_{u_1,t_{i-\kappa}} \mid \tag{12}$$

Namely, the CR $R_{u_1,t_{i+1}}$ meets the MAB constraint when we consider two queries at time t_{i+1} and $t_{i-\kappa}$.

In summary, the CR $R_{u_1,t_{i+1}}$ is a safe CR even when we consider any two queries at time t_{i+1} and $t_{i-\kappa}$.

Therefore, Theorem 5 holds. ■

B. Time Complexity

This subsection investigates the computation overhead in RobLoP. According to the detailed introduction of the proposed algorithm, we get the following results:

Theorem 6: The time complexity in RobLoP is at most $O(N, N_o^2(\Omega_{sub}))$, where N is the number of users querying LBS, and $N_o(\Omega_{sub})$ is the number of sub-candidate user sets.

Proof: In the first step, as analyzed in Section III-A, the time complexity of Algorithm 1 is $O(N)$.

In second step, all querying users are contained in the candidate user sets in the worst case, and thus the points defined in Definition 4 corresponding to all these users are selected. Therefore, the time complexity is $O(N)$ in the worst case.

As analysed above, the computation cost in generating safe CR is $O(N_o(\Upsilon))$. Then selecting one of the safe CRs with maximum area as the final output. During this process, the computation cost is $O(N_o(\Omega))$. Since each Θ is mapped to a strict point set, $N_o(\Upsilon) = N_o(\Omega)$. Therefore, the time complexity in the third step is $O(N_o(\Omega))$.

As analyzed in Section III-D, the time complexity in the last step is $O(N_o(\Omega_{sub})^2)$.

As $N_o(\Omega_{sub}) \ll N$ and $N_o(\Omega) \ll N$, therefore the worst-case time complexity in this work is $O(N, N_o^2(\Omega_{sub}))$.

Overall, Theorem 6 holds. ■

C. Limitation of k-Anonymity Alternatives

Many alternatives for k-anonymity have been proposed, e.g., approaches that employ geo-indistinguishability (including mix zone [27], dummy [28], etc.), private information retrieval protocols (PIR) [29], and differential privacy [30]. These techniques, however, may also suffer from the spatio-temporal correlation attacks in the context of continuous LBS.

To be more concrete, mix zone is vulnerable to timing attacks (i.e., temporal correlation attacks) [31] and transition attacks (i.e., spatial correlation attacks) [32].

Dummy queries are also susceptible to spatio-temporal correlation attacks in continuous LBS scenarios, as dummy querying users cannot credibly imitate the mobility of living people (i.e., the spatio-temporal correlation among locations), which can be easily identified by attackers [33].

Differential privacy itself cannot be directly applied to continuous LBS. The latest research [34] adapts differential privacy to the LBS scenario, and takes temporal correlations among locations into consideration. But the spatial correlation among users' locations are ignored, which can be used by attackers to further disclose users' location privacy.

PIR allows a user to retrieve information in private from database, but it is not applicable to the continuous LBS scenario where the server frequently searches for results in cipher text space for the user, as the server can infer what exact information this user has requested from the different number of retrievals on data blocks [35].

retrieve information in private from database in the absence that the server knows what exact information this user has requested. But it suffers from heavy overhead and limited operation in ciphertext space.

In summary, all these methods cannot completely protect users' location privacy when directly applied to continuous LBS. Our work focuses on refining k-anonymity so that it is applicable to continuous LBS, thereby protecting users' location privacy against LDA.

V. Performance Evaluation

A. Experimental Setup

We evaluate the performance of RobLoP through extensive experiments by leveraging synthetic, Athens trucks dataset [36], the measured dataset, and real-world dataset, loc-Gowalla [37]. The synthetic data traces are generated by Thomas Brinkhoff Network-based Generator of Moving Objects [38], which outputs the moving objects of the road map in Oldenburg County. In Athens trucks dataset, $150,000$ trajectory segments (each with 500 location updates indicating 500 LBS queries) are randomly selected from the original trajectories to represent $150,000$ users. We measure 6 movement trajectories by Samsung Galaxy S5 equipped with a 2.5G Hz CPU, 2G RAM, and Android 4.4 OS, and randomly select 36 trajectory segments (each with 15 location updates) from the original trajectories to represent 36 users. Loc-Gowalla dataset collects 6.4 million check-ins (i.e., locations) from Feb. 2009 and Oct. 2010, consisting of $196,591$ nodes (i.e., users).

In addition, we compare the performance of RobLoP to other three algorithms: IClique [8], RobLoP*. RobLoP* is a simplified version of RobLoP without considering MMB and MAB constraints (thus it is similar to the original k-anonymity scheme). Note that RobLoP* cannot prevent a user's privacy against LDA; it is only used for comparisons when additional constraints are taken into account. We do not compare our work with other privacy preserving methods, such as mix zone [39] and differential privacy [30], since they cannot be directly used to protect users' location privacy against LDA.

All the experiments are implemented in C++ and conducted on a desktop PC with an Intel Core i5 2.79G Hz processor and 4G RAM. The queries containing locations and privacy parameters are set as the input. The output are a series of safe CRs or the notice that a certain query is expired. When a user conducts the LBS query for the first time, his/her MMB is considered to be infinite. The MMBs induced by his follow-up queries are considered as the maximum movement distance. The default parameters are set as follows: $A_{\min}$ is randomly chosen from $[0.005\%, 0.01\%]$, indicating the ratio of the system area, $d_t = 0.1$s, query interval is 60s, k_{u_j} is randomly chosen from $[2, 10]$, the number of users is $150,000$, and the speed setting is medium.

We propose the following metrics to quantify the privacy preservation, the quality of LBS (i.e., the data utility), and the overhead.

- **The average success rate s_r**, which is average percentage of users successfully cloaked. Here, we consider a user, e.g., u_1, is successfully cloaked when u_1 is cloaked in a CR including no less than $(k_{u_1} - 1)$ other users and with a size larger than $A_{u_1,\min}$, and is protected against LDA simultaneously. s_r quantifies the privacy preservation.
- **The cumulative distribution function (CDF) of the area of the CRs Ar_{cd}**, which quantifies the quality of LBS and the data utility as the CRs (instead of the exact locations) will result in inaccurate querying results from the LBS sever.
- **The average cloaking time t_c**, which is the average time spent in identifying candidate user sets, searching for the strict point sets, and generating the safe CR.
- **The average processing time t_p**, which is the average time spent in waiting for being cloaked after requesting LBS. Both t_c and t_p quantify the overhead.

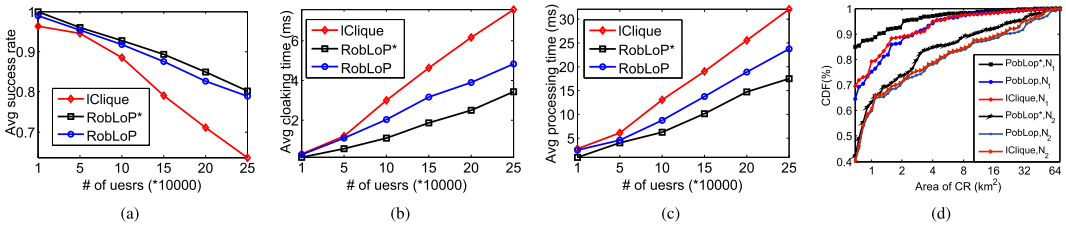

Fig. 8. Impact of the number of users. The number of users $N_1 = 100000$ and $N_2 = 200000$. (a) Average success rate. (b) Average cloaking time. (c) Average processing time. (d) CDF of the area of CRs.

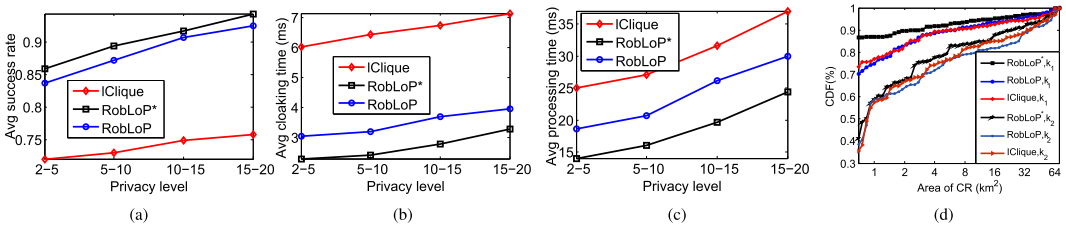

Fig. 9. Impact of privacy parameter k. k_1 and k_2 indicate that k is set to be $[2, 5]$ and $[5, 10]$. (a) Average success rate. (b) Average cloaking time. (c) Average processing time. (d) CDF of the area of CRs.

B. Performance Varies With User Numbers

We first investigate the impact of the number of users. More users, coupled with increasing t_p, lead to more queries expired. Therefore, s_r in Fig. 8(a) decreases with increasing number of users. It can be observed that RobLoP yields a higher s_r than IClique, and the superiority of RobLoP is even more prominent as the number of users increases. The reason is that IClique would cause "false-negative" cases, as we mentioned in Section I-B.

Figs. 8(b) and 8(c) indicate that, t_c and t_p increase with the number of users, because more users to be cloaked will intuitionally increase t_c and t_p. Additionally, t_c and t_p in RobLoP are obviously more robust to the number of users than those in IClique, especially when the number of users is more than $150,000$. The reasons are that (1) RobLoP desirably excludes unqualified users and only deals with the rest little amount of data, as we mentioned in Section I-C; (2) Sub-candidate user sets are updated timely for subsequent queries (cf. Step (d) in Section III-D). Furthermore, compared with RobLoP*, t_c and t_p in RobLoP are slightly larger than those in RobLoP*, implying a little sacrifice for protecting against LDA in RobLoP.

Fig. 8(d) shows the CDF of the area of CRs. It can be observed that the size of CRs decreases with the number of users. That is because algorithms can cloak nearer neighbors together when more users query LBS. In addition, the size of CRs in both IClique and RobLoP is larger than that in RobLoP*, since IClique and RobLoP consider more constraints when cloaking users's queries. Lastly, the size of CRs in RobLoP is a bit larger than that in IClique, as RobLoP outputs the CR with maximum area while IClique outputs the CR with minimal area. In this sense, it is acceptable since RobLoP provides privacy protection for more users (cf. Fig. 8(a)).

C. Performance Varies With Privacy Parameters

Parameter $\boldsymbol{k}$*:* In this part, our first interest is to check whether the privacy parameter k has an effect on the performance. So, in Fig. 9, we fix k at a range of 5 and increase both the lower and upper bound. It can be observed that s_r of each algorithm slightly increases with k. The reason is that both RobLoP and IClique have to perform more times to generate safe CR for users' queries when users issue smaller k. Thus it takes more time, and more users' queries are expired. As a result, the success rate is deteriorated. So when all users issue larger k, both algorithms will be more effective. Furthermore, it is obvious that s_r in IClique is less than that in RobLoP, because in IClique more queries are expired, and many requests supposed to be successfully cloaked fail to be cloaked. Overall, compared with RobLoP*, 2% and 15% average success rates are sacrificed in RobLoP and IClique, respectively, for preserving location privacy against LDA.

In Figs. 9(b) and 9(c), the overall trend in all cloaking algorithms is that t_c and t_p increase with the increasing k, due to a more constrained privacy requirement when k increases. In addition, t_c and t_p in IClique are longer than those in RobLoP, since IClique cannot exclude unqualified users, thus cannot scale up to more constrained privacy requirements. Moreover, both t_c and t_p in RobLoP are larger than those in RobLoP*, as RobLoP* does not consider the LDA.

Fig. 9(d) shows the CDF of CRs, and we can see the size of CRs increases with users' privacy requirements k, since algorithms have to generate larger CRs so that more users are contained in CRs.

Parameter $\boldsymbol{A_{\min}}$*:* In Fig. 10, we vary the privacy parameter $A_{\min}$ and investigate the impact of $A_{\min}$. First, in Fig. 10(a), the success rate decreases with the increasing $A_{\min}$. The reason is that, on one hand, our algorithm has to cloak farther users to generate a larger CR to meet the larger $A_{\min}$, which will incur more computation cost. As a result, more users' queries will be expired, and thus cannot be successfully cloaked. On the other hand, a larger $A_{\min}$ will breach the MMB/MAB constraints, results in that more users' queries cannot be successfully cloaked. As expected, in Fig. 10(c), t_p first changes slowly, and then rapidly increases, as more constrained privacy requirement $A_{\min}$ increases the overhead. In contrast, t_c is not affected much by $A_{\min}$. That is because

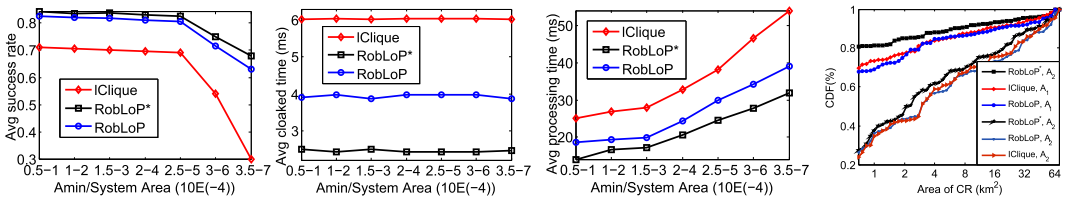

Fig. 10. Impact of privacy parameter $A_{\min}$. A_1 and A_2 indicate $A_{\min}$ is set to be $[0.005\%, 0.01\%]$ and $[0.02\%, 0.04\%]$. (a) Average success rate. (b) Average cloaking time. (c) Average processing time. (d) CDF of the area of CRs.

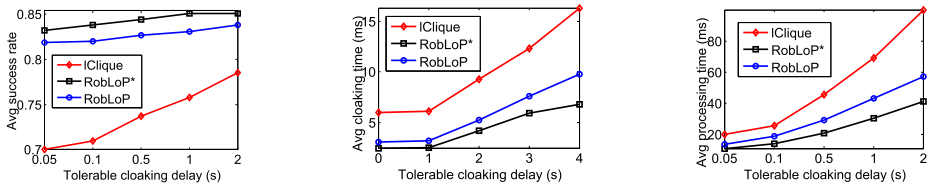

Fig. 11. Impact of tolerable cloaking delay d_t. (a) Average success rate. (b) Average cloaking time. (c) Average processing time.

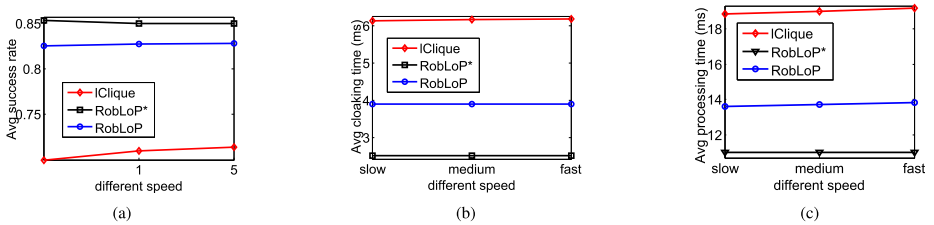

Fig. 12. Impact of moving speed. (a) Average success rate. (b) Average cloaking time. (c) Average processing time.

in RobLoP and in RobLoP*, the last step is to select a qualified safe CR as the final output according to $A_{\min}$, and the computation overhead in this paper is dominated by the first and second steps, searching candidate user sets and strict point sets. Similarly, the most expensive operations in IClique do not involve $A_{\min}$. At last, from Fig. 10 we can observe that RobLoP can maintain desirable cloaking rate and overhead while the size of CRs is sufficient to preserve users' location privacy.

Fig. 10(d) shows that the size of CRs increases with users' requirements A_{min}. That is not surprising, as algorithms have to generate larger CRs to meet larger A_{min}.

D. Performance Varies With Parameter d_t

We finally study the effect of parameter d_t on the performance. In Fig. 11, we vary the tolerable cloaking delay from 0.05 seconds to 2 seconds. It can be observed in Fig. 11(a) that s_r in IClique is heavily affected, increasing from 0.70 to 0.785. In contrast, s_r of RobLoP and RobLoP* are not significantly affected. That is because, t_c in IClique is longer than that in RobLoP and RobLoP*. Thus, when prolonging the d_t, more expired queries in IClique can be processed. As shown in Fig. 11(b), RobLoP and RobLoP* perform much better than IClique, as t_c in RobLoP and RobLoP* slowly increases with d_t while t_c in IClique rapidly increases. That is not surprising, as more expired queries in IClique can be cloaked when we prolong d_t. As a result, t_c in IClique rapidly increases. Lastly, t_p in each algorithm increases with d_t, since queries have more time to wait for being cloaked.

E. Performance Varies With User Movements

This part examines the impact of moving speed on location privacy protection. A faster speed means a bigger size of MMB, incurring more users in the MMB of one user, and thus greater possibilities to be successfully cloaked. As such, s_r of RobLoP and IClique slowly increase with the increasing speed. Since RobLoP* does not consider the MMB constraint, s_r of RobLoP* is not affected and fixes at around 0.99. Again, in Figs. 12(b) and 12(c), t_c and t_p in RobLoP* are also not affected, while those in RobLoP and IClique they are slightly increasing, due to more users included in the MMB of the users to be proceeded.

F. Real Dataset Results

1) Trucks Dataset Results: In Fig. 13, t_c and t_p in all algorithms increase with privacy parameter k, which is similar to that in Fig. 9, with RobLoP performing better than IClique and RobLoP*.

2) Measured Dataset Results: In Fig. 14, we can clearly see that s_r increases when increasing both the lower and upper bound of k, which is similar to that in Fig. 13(a). In addition, t_c and t_p increase with increasing the lower and upper bound at the same time, which shares the same trend to that in Figs. 13(b) and 13(c).

3) Loc-Gowalla Dataset Results: As shown in Fig. 15, s_r, t_c, and t_p in all algorithms increase with k. In addition, the size of CRs increases with users' privacy requirements k. That is because algorithms have to generate larger CRs so that more users are contained in CRs.

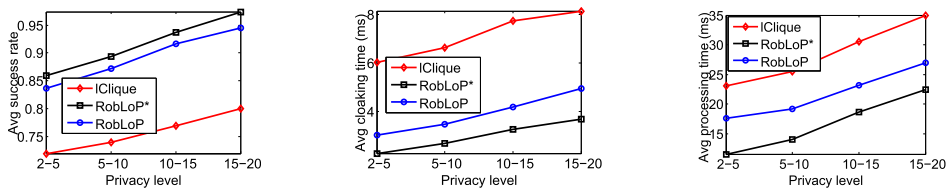

Fig. 13. Trucks dataset results. (a) Average success rate. (b) Average cloaking time. (c) Average processing time.

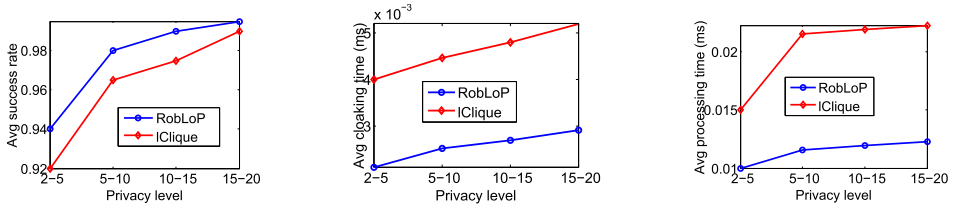

Fig. 14. Measured dataset results. (a) Average success rate. (b) Average cloaking time. (c) Average processing time.

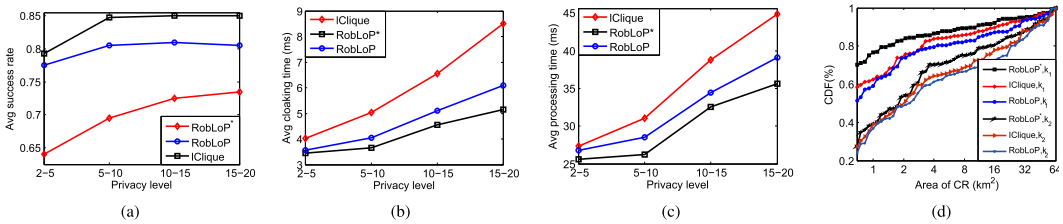

Fig. 15. Loc-Gowalla dataset results. (a) Average success rate. (b) Average cloaking time. (c) Average processing time. (d) CDF of the area of CRs.

G. Impact of MMB and MAB Estimation

In this part, we show some preliminary results on the impact of MMB and MAB estimation when taking the constraint of road networks into consideration. We use the real-word dataset, loc-Gowalla [37]. We compare RobLoP with RobLoP+, which estimates user's MMB and MAB as the existing work [23], where user's moving speed varies with time and is determined by the traffic condition.

In addition, we define another kind of attacks, LDA+ that accurately estimates MMB and MAB as RobLoP+ does. Note that LDA (resp. RobLoP) and LDA+ (resp. RobLoP+) only differ in the estimation of MMB and MAB. That is, LDA (resp. RobLoP) does a coarse-grained estimation without considering constraints of road network while LDA+ (resp. RobLoP+) does a fine-grained estimation taking constraints of road network into consideration.

Accordingly, we define attack success rate P_{AR} (resp. $P_{A^+R^+}$), the average percentage of users cloaked by RobLoP (resp. RobLoP+) and suffering from LDA (resp. LDA+). Likewise, we define attack success rate P_{A^+R} (resp. P_{AR^+}), the average percentage of users cloaked by RobLoP (resp. RobLoP+) and suffering from LDA+ (resp. LDA).

Table I shows the attack success rates varying with k. It can be observed that P_{AR}, P_{AR^+}, and $P_{A^+R^+}$ equal to 0, and are not affected by k. That is because, the attacker launching LDA (resp. LDA+) estimates user's MMB and MAB as RobLoP (resp. RobLoP+) does, and thus they cannot filter out other querying users cloaked with the user. As a result, the attacker cannot disclosure the user's location privacy, i.e., $P_{AR} = 0$ (resp. $P_{A^+R^+} = 0$). Likewise, RobLoP+ does a fine-grained estimation of MMB and MAB while the attacker performs a coarse-grained estimation (i.e., launch LDA), so the attacker cannot distinguish the user from other querying users and thereby $P_{AR^+} = 0$. In addition, we can observe that P_{A^+R} increases with k. The reason is that the attacker launching LDA+ obtains more accurate estimation of MMB and MAB than RobLoP, so they can filter out the users they are not interested in. Furthermore, larger k results in larger CRs, and thus the attacker can filter out more users they are not interested in. So P_{A^+R} increases with k. For the same reason, P_{AR}, P_{AR^+}, and $P_{A^+R^+}$ are not affected by $A_{\min}$, and P_{A^+R} increases with $A_{\min}$, as shown in Table II.

TABLE I
ATTACK SUCCESS RATES VARYING WITH k

k	$[2, 10]$	$[2, 5]$	$[5, 10]$	$[10, 15]$	$[15, 20]$
P_{AR}	0	0	0	0	0
P_{A^+R}	19%	10.4%	25%	46%	71.3%
P_{AR^+}	0	0	0	0	0
$P_{A^+R^+}$	0	0	0	0	0

In summary, the information asymmetry between the attacker and the defender (RobLoP) in terms of MMB and MAB estimation can affect the performance. Existing work studying users' mobility in road network can be directly applied to and combined with our work, to obtain a more accurate estimation of the MMB and MAB.

TABLE II
ATTACK SUCCESS RATES VARYING WITH $A_{\min}$

$A_{\min(\times 10^{-4})}$	$[0.5, 1]$	$[1, 2]$	$[1.5, 3]$	$[2, 4]$	$[2.5, 5]$	$[3, 6]$
P_{AR}	0	0	0	0	0	0
P_{A+R}	10.51%	15.94%	22.7%	30%	46.3%	75.2%
P_{AR+}	0	0	0	0	0	0
P_{A+R+}	0	0	0	0	0	0

VI. CONCLUSION

In this paper we have presented RobLoP, a robust algorithm aiming at preserving location privacy against LDA in continuous LBS queries. The key insight of RobLoP is to consider both constraints of MMB and MAB simultaneously in a uniform way. To the best of our knowledge, RobLoP is the first work that can preserve privacy against LDA thoroughly and closely in continuous LBS queries. Extensive experiments via both synthetic data and real data have demonstrated the effectiveness and efficiency of RobLoP.

Since k-anonymity based privacy preserving techniques inherently rely on a larger number of users querying LBS, how to make our algorithm more robust to the number of users will be quite challenging and is left as our future work. We are also interested in the privacy preservation against spatio-temporal correlation attacks in continuous LBS, combining other privacy preserving approaches (e.g., mix zone [39] or dummy [28]) with k-anonymity.

APPENDIX
PROOF OF THEOREM 2

Proof: For the sake of explanation, we first present the following definitions.

Definition 5: Given two users u_1 and u_2, the threshold of T_{u_1,u_2} (denoted by $T_{u_1,u_2,0}$) is the maximum value of T_{u_1,u_2} which enables the CR to meet Theorem 1, where T_{u_1,u_2} is the distance between the centers of R_{u_1,t_i} and R_{u_2,t_i}

Definition 6: The diameter of the intersection region, denoted by DIR_{u_1,u_2}, is the distance between the two points where MMB_{u_1}, MMB_{u_2} and the line acrossing the centers of R_{u_1,t_i} and R_{u_2,t_i} intersect.

As three situations exist for a specific user (cf. § II-D), there are six cases for two users when pairwise coupling the privacy preserving requirements of a bunch of users. Next, we explain the proofs in six cases as follows.

1) Case 1: In Case 1, $2r_{u_1,t_i} \le v_{u_1}\Delta t_{u_1}$ and $2r_{u_2,t_i} \le v_{u_2}\Delta t_{u_2}$ (cf. Fig. 16(a)). Without loss of generality we assume $r_{u_2,t_i} \le r_{u_1,t_i}$. Obviously, $|o_{u_1,t_i}p| = v_{u_1}\Delta t_{u_1} + r_{u_1,t_i}$, $|ap| = |jc| = 2r_{u_1,t_i}$, and $|ao_{u_2,t_i}| = r_{u_2,t_i} + v_{u_2}\Delta t_{u_2}$. So, we can get:

$$\begin{aligned} T_{u_1,u_2,0} &= |o_{u_1,t_i}o_{u_2,t_i}| = |o_{u_1,t_i}p| + (|ao_{u_2,t_i}| - |ap|) \\ &= v_{u_1}\Delta t_{u_1} - r_{u_1,t_i} + r_{u_2,t_i} + v_{u_2}\Delta t_{u_2} \end{aligned} \tag{13}$$

Since $v_{u_2}\Delta t_{u_2} \le v_{u_1}\Delta t_{u_1}$, when MMB_{u_2} and the center circle O'_{u_1} intersect at point a, $MMB_{u_1} \cap O'_{u_2} \neq$, which enables the CR to meet Formula (1) in Theorem 1. Thus Formula (5) holds.

According to Definition 6,

$$DIR_{u_1,u_2} = |ap| = 2r_{u_1,t_i} \tag{14}$$

where DIR_{u_1,u_2} is the maximum diameter of the safe CR. So if $|u_{1,t_{i+1}}u_{2,t_{i+1}}|$ is longer than DIR_{u_1,u_2}, the CR determined by locations of u_1 and u_2 must be exceeding the intersection region, $MMB_{u_1} \cap MMB_{u_2}$. Therefore Formula (6) holds.

In summary, in Case 1, Theorem 2 holds.

2) Case 2: In Case 2, $v_{u_1}\Delta t_{u_1} \le r_{u_1,t_i}$, $v_{u_2}\Delta t_{u_2} \le r_{u_2,t_i}$, as shown in Fig. 16(b). Assume $r_{u_2,t_i} \le r_{u_1,t_i}$. According to Definition 5, the threshold of intersection degree $T_{u_1,u_2,0}$ in this case is,

$$\begin{aligned} T_{u_1,u_2,0} &= |o_{u_1,t_i}o_{u_2,t_i}| = |o_{u_1,t_i}d| - |o_{u_2,t_i}d| \\ &= |o_{u_1,t_i}f| - |o_{u_2,t_i}d| \\ &= (|o_{u_1,t_i}l| + |fl|) - (|ad| - |ao_{u_2,t_i}|) \\ &= (|o_{u_1,t_i}l| + |fl|) - (|bf| - |ao_{u_2,t_i}|) \end{aligned} \tag{15}$$

where $|o_{u_1,t_i}l| = r_{u_1,t_i}$, $|fl| = v_{u_1}\Delta t_{u_1}$, $|bf| = 2r_{u_1,t_i}$, and $|ao_{u_2,t_i}| = r_{u_2,t_i} + v_{u_2}\Delta t_{u_2}$. So,

$$T_{u_1,u_2,0} = r_{u_2,t_i} + v_{u_2}\Delta t_{u_2} + v_{u_1}\Delta t_{u_1} - r_{u_1,t_i} \tag{16}$$

According to the privacy requirement in Fig. 3(b), the two center circles O'_{u_1} and O'_{u_2}, should be included in $R_{u_1,u_2,t_{i+1}}$. Thus, T_{u_1,u_2} should be smaller than $T_{u_1,u_2,0}$. Therefore, Formula (5) holds.

Thus in this case

$$DIR_{u_1,u_2} = |ae| = |bf| = 2r_{u_1,t_i} \tag{17}$$

Since DIR_{u_1,u_2} is the maximum diameter of the safe CR, Formula (6) holds.

In summary, in Case 2, Theorem 2 holds.

3) Case 3: As shown in Fig. 16(c), in Case 3 $r_{u_1,t_i} \le v_{u_1}\Delta t_{u_1} \le 2r_{u_1,t_i}$, $r_{u_2,t_i} \le v_{u_2}\Delta t_{u_2} \le 2r_{u_2,t_i}$. Assume $r_{u_2,t_i} \le r_{u_1,t_i}$. DIR_{u_1,u_2} in this case is,

$$T_{u_1,u_2,0} = |o_{u_1,t_i}e| - |o_{u_2,t_i}e| = |o_{u_1,t_i}e| - (|ae| - |o_{u_2,t_i}a|) \tag{18}$$

where, $|o_{u_1,t_i}e| = r_{u_1,t_i} + v_{u_1}\Delta t_{u_1}$, $|ae| = 2r_{u_1,t_i}$, and $|o_{u_2,t_i}a| = r_{u_2,t_i} + v_{u_2}\Delta t_{u_2}$. So,

$$T_{u_1,u_2,0} = v_{u_1}\Delta t_{u_1} + v_{u_2}\Delta t_{u_2} - r_{u_1,t_i} + r_{u_2,t_i} \tag{19}$$

According to the privacy requirement in Fig. 3(c), $R_{u_1,u_2,t_{i+1}}$ should not only be in both MMB_{u_1} and MMB_{u_2}, but also overlap the two center circles, O'_{u_1} and O'_{u_2}. Thus, T_{u_1,u_2} should be smaller than $T_{u_1,u_2,0}$. Therefore, Formula (5) holds.

Thus in this case

$$DIR_{u_1,u_2} = |ae| = |lm| = 2r_{u_1,t_i} \tag{20}$$

So, $|u_{1,t_{i+1}}u_{2,t_{i+1}}|$ should be less than $2r_{u_1,t_i}$, and Formula (6) holds.

In summary, Theorem 2 holds in Case 3.

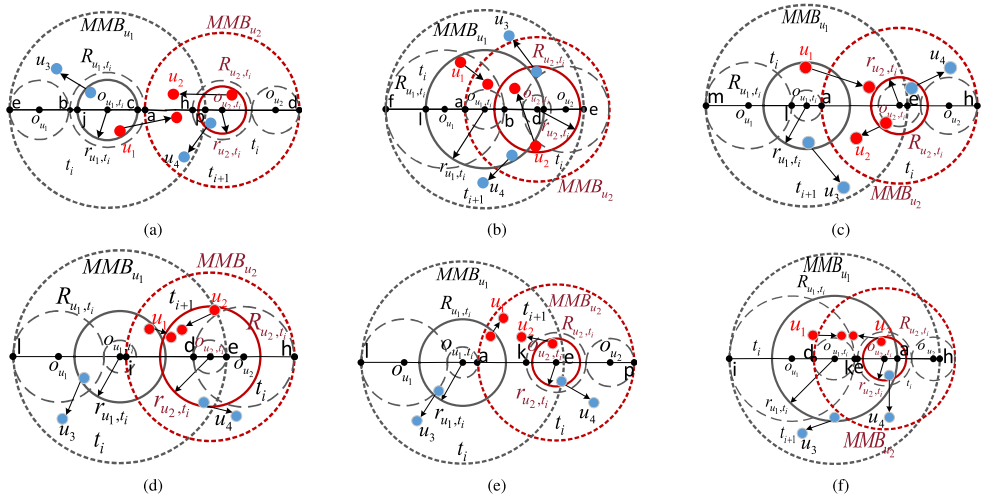

Fig. 16. Illustrations of the six cases when considering the requirements of two users simultaneously. (a) Case 1. (b) Case 2. (c) Case 3. (d) Case 4. (e) Case 5. (f) Case 6.

4) Case 4: Case 4 is shown in Fig. 16(d), where $r_{u_1,t_i} \leq v_{u_1}\Delta t_{u_1} \leq 2r_{u_1,t_i}$, $v_{u_2}\Delta t_{u_2} \leq r_{u_2,t_i}$. Assume $r_{u_1,t_i} \leq r_{u_2,t_i}$, DIR_{u_1,u_2} in this case is,

$$\begin{aligned} T_{u_1,u_2,0} &= |o_{u_1,t_i}o_{u_2,t_i}| = |o_{u_1,t_i}e| - |o_{u_2,t_i}e| \\ &= |o_{u_1,t_i}e| - |o_{u_2,t_i}d| \\ &= |o_{u_1,t_i}e| - (|dh| - |o_{u_2,t_i}h|) \end{aligned} \tag{21}$$

where $|o_{u_1,t_i}e| = r_{u_1,t_i} + v_{u_1}\Delta t_{u_1}$, $|dh| = 2r_{u_1,t_i}$, and $|o_{u_2,t_i}h| = r_{u_2,t_i} + v_{u_2}\Delta t_{u_2}$. So,

$$T_{u_1,u_2,0} = (r_{u_1,t_i} + v_{u_1}\Delta t_{u_1}) - (r_{u_2,t_i} - v_{u_2}\Delta t_{u_2}) \tag{22}$$

According to the privacy requirement in Fig. 3(b), the center circle O'_{u_2}, should be included in $R_{u_1,u_2,t_{i+1}}$. While according to the privacy requirement in Fig. 3(c), $R_{u_1,u_2,t_{i+1}}$ should overlap the center circle O'_{u_1}. Thus, T_{u_1,u_2} should be smaller than $T_{u_1,u_2,0}$. Therefore, Formula (5) holds.

Thus in this case

$$\begin{aligned} DIR_{u_1,u_2} &= |ie| = |io_{u_2,t_i}| + (|dh| - |o_{u_2,t_i}h|) \\ &= v_{u_2}\Delta t_{u_2} + 3r_{u_2,t_i} - r_{u_2,t_i} - v_{u_2}\Delta t_{u_2} \\ &= 2r_{u_2,t_i} \end{aligned} \tag{23}$$

As DIR_{u_1,u_2} is the maximum diameter of the safe CR, Formula (6) holds.

In summary, in Case 4, Theorem 2 holds.

5) Case 5: As shown in Fig. 16(e), in Case 5 $r_{u_1,t_i} \leq v_{u_1}\Delta t_{u_1} \leq 2r_{u_1,t_i}$, $2r_{u_2,t_i} \leq v_{u_2}\Delta t_{u_2}$. Assume $r_{u_2,t_i} \leq r_{u_1,t_i}$. According to the privacy requirement in Figs. 3(c) and 3(a), $R_{u_2,u_2,t_{i+1}}$ should overlap the center circles, O'_{u_1} and O'_{u_2}. Consequently, the threshold of intersection degree in this case is,

$$T_{u_1,u_2,0} = |o_{u_1,t_i}o_{u_2,t_i}| = |o_{u_1,t_i}e| - (|ae| - |ao_{u_2,t_i}|) \tag{24}$$

where $|o_{u_1,t_i}e| = r_{u_1,t_i} + v_{u_1}\Delta t_{u_1}$, $|ae| = 2r_{u_1,t_i}$, and $|ao_{u_2,t_i}| = r_{u_2,t_i} + v_{u_2}\Delta t_{u_2}$. So,

$$T_{u_1,u_2,0} = (r_{u_2,t_i} + v_{u_2}\Delta t_{u_2}) + (v_{u_1}\Delta t_{u_1} - r_{u_1,t_i}) \tag{25}$$

Thus, T_{u_1,u_2} should be smaller than $T_{u_1,u_2,0}$. Therefore, Formula (5) holds.

In this case, we can get

$$DIR_{u_1,u_2} = |ae| = |kl| = 2r_{u_1,t_i} \tag{26}$$

DIR_{u_1,u_2} is the maximum diameter of the safe CR, thus Formula (6) holds.

In summary, in Case 5, Theorem 2 holds.

6) Case 6: As shown in Fig. 16(f), $v_{u_1}\Delta t_{u_1} \leq r_{u_1,t_i}$, $2r_{u_2,t_i} \leq v_{u_2}\Delta t_{u_2}$. Assume $r_{u_2,t_i} \leq r_{u_1,t_i}$. According to the privacy requirement in Figs. 3(b) and 3(c), R_{u_2,u_2,t_i} should overlap the center circle O'_{u_2}, and include the center circle O'_{u_1}. Consequently, DIR_{u_1,u_2} in this case is,

$$T_{u_1,u_2,0} = |o_{u_1,t_i}o_{u_2,t_i}| = |o_{u_1,t_i}e| - (|ad| - |do_{u_2,t_i}|) \tag{27}$$

where $|o_{u_1,t_i}e| = r_{u_1,t_i}$, $|ad| = |kc| = 2r_{u_1,t_i} - v_{u_1}\Delta t_{u_1}$, and $|do_{u_2,t_i}| = r_{u_2,t_i} + v_{u_2}\Delta t_{u_2}$. So,

$$T_{u_1,u_2,0} \tag{28}$$

which enables the CR to meet Formula (1) in Theorem 1. Thus Formula (5) holds.

Then we get the maximum diameter of the safe CR

$$DIR_{u_1,u_2} = |dh| = |ki| = 2r_{u_1,t_i} \tag{29}$$

Therefore, Formula (6) holds.

In summary, in Case 6, Theorem 2 holds.

All in all, Theorem 2 holds. ■

REFERENCES

[1] G. Heinemann and C. Gaiser, *Social-Local-Mobile: The Future of Location-Based Services*. Berlin, Germany: Springer, 2014.

[2] G. Gartner and H. Huang, Eds., *Progress in Location-Based Services 2014*. Berlin, Germany: Springer, 2015.

[3] T. Shu, Y. Chen, and J. Yang, "Protecting multi-lateral localization privacy in pervasive environments," *IEEE/ACM Trans. Netw.*, vol. 23, no. 5, pp. 1688–1701, Oct. 2015.

[4] C. Y. T. Ma, D. K. Y. Yau, N. K. Yip, and N. S. V. Rao, "Privacy vulnerability of published anonymous mobility traces," *IEEE/ACM Trans. Netw.*, vol. 21, no. 3, pp. 720–733, Jun. 2013.

[5] K. Fawaz and K. G. Shin, "Location privacy protection for smartphone users," in *Proc. ACM SIGSAC CCS*, 2014, pp. 239–250.

[6] E. Naghizade, J. Bailey, L. Kulik, and E. Tanin, "How private can I be among public users?" in *Proc. ACM UbiComp*, 2015, pp. 1137–1141.

[7] Y. Zhang, Q. Chen, and S. Zhong, "Privacy-preserving data aggregation in mobile phone sensing," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 5, pp. 980–992, May 2016.

[8] X. Pan, J. Xu, and X. Meng, "Protecting location privacy against location-dependent attacks in mobile services," *IEEE Trans. Knowl. Data Eng.*, vol. 24, no. 8, pp. 1506–1519, Aug. 2012.

[9] L. Sweeney, "k-anonymity: A model for protecting privacy," *Int. J. Uncertainty, Fuzziness, Knowl. Syst.*, vol. 10, no. 5, pp. 557–570, 2002.

[10] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking," in *Proc. ACM MobiSys*, 2003, pp. 31–42.

[11] B. Gedik and L. Liu, "Protecting location privacy with personalized k-anonymity: Architecture and algorithms," *IEEE Trans. Mobile Comput.*, vol. 7, no. 1, pp. 1–18, Jan. 2008.

[12] R. Paulet, M. G. Kaosar, X. Yi, and E. Bertino, "Privacy-preserving and content-protecting location based queries," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 5, pp. 1200–1210, May 2014.

[13] K. Vu, R. Zheng, and L. Gao, "Efficient algorithms for k-anonymous location privacy in participatory sensing," in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 2399–2407.

[14] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, "Achieving k-anonymity in privacy-aware location-based services," in *Proc. IEEE INFOCOM*, Apr. 2014, pp. 754–762.

[15] R. Cheng, Y. Zhang, E. Bertino, and S. Prabhakar, "Preserving user location privacy in mobile data management infrastructures," in *Proc. Int. Workshop Privacy Enhancing Technol.*, 2006, pp. 393–412.

[16] G. Ghinita, M. L. Damiani, C. Silvestri, and E. Bertino, "Preventing velocity-based linkage attacks in location-aware applications," in *Proc. ACM SIGSPATIAL GIS*, 2009, pp. 246–255.

[17] T. Hashem, L. Kulik, and R. Zhang, "Countering overlapping rectangle privacy attack for moving KNN queries," *Inf. Syst.*, vol. 38, no. 3, pp. 430–453, 2013.

[18] N. Nguyen, S. Han, and M. Shin, "URALP: Unreachable region aware location privacy against maximum movement boundary attack," *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 8, p. 246216, 2015.

[19] T. Xu and Y. Cai, "Location anonymity in continuous location-based services," in *Proc. ACM SIGSPATIAL GIS*, 2007, p. 39.

[20] X. Liu, K. Liu, L. Guo, X. Li, and Y. Fang, "A game-theoretic approach for achieving k-anonymity in location based services," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 2985–2993.

[21] J. Xu, X. Tang, H. Hu, and J. Du, "Privacy-conscious location-based queries in mobile environments," *IEEE Trans. Parallel Distrib. Syst.*, vol. 21, no. 3, pp. 313–326, Mar. 2010.

[22] D. Yang, X. Fang, and G. Xue, "Truthful incentive mechanisms for k-anonymity location privacy," in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 2994–3002.

[23] P. H. Li, M. L. Yiu, and K. Mouratidis, "Discovering historic traffic-tolerant paths in road networks," *GeoInformatica*, vol. 21, no. 1, pp. 1–32, 2017.

[24] C. Wang, H. Lin, and H. Jiang, "CANS: Towards congestion-adaptive and small stretch emergency navigation with wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 15, no. 5, pp. 1077–1089, May 2016.

[25] M. De Berg, O. Cheong, M. van Kreveld, and M. Overmars, *Computational Geometry: Algorithms and Applications*, 3rd ed. Berlin, Germany: Springer, 2008.

[26] F. P. Preparata and M. Shamos, *Computational Geometry: An Introduction*. Berlin, Germany: Springer, 2012.

[27] A. R. Beresford and F. Stajano, "Location privacy in pervasive computing," *IEEE Pervasive Comput.*, vol. 2, no. 1, pp. 46–55, Jan. 2003.

[28] H. Kido, Y. Yanagisawa, and T. Satoh, "An anonymous communication technique using dummies for location-based services," in *Proc. Int. Conf. Pervasive Services*, 2005, pp. 88–97.

[29] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan, "Private information retrieval," in *Proc. IEEE Symp. Found. Comput. Sci.*, Oct. 1995, pp. 41–50.

[30] C. Dwork, "Differential privacy," in *Proc. Int. Colloq. Autom., Lang. Programm.*, 2006, pp. 1–12.

[31] B. Palanisamy, L. Liu, K. Lee, A. Singh, and Y. Tang, "Location privacy with road network mix-zones," in *Proc. IEEE Int. Conf. Mobile Ad-Hoc Sensor Netw.*, Dec. 2012, pp. 124–131.

[32] B. Palanisamy and L. Liu, "Attack-resilient mix-zones over road networks: Architecture and algorithms," *IEEE Trans. Mobile Comput.*, vol. 14, no. 3, pp. 495–508, Mar. 2015.

[33] V. Bindschaedler and R. Shokri, "Synthesizing plausible privacy-preserving location traces," in *Proc. IEEE S&P*, May 2016, pp. 546–563.

[34] Y. Xiao and L. Xiong, "Protecting locations with differential privacy under temporal correlations," in *Proc. ACM CCS*, 2015, pp. 1298–1309.

[35] H. Sun and S. A. Jafar, "The capacity of private information retrieval," *IEEE Trans. Inf. Theory*, vol. 63, no. 7, pp. 4075–4088, Jul. 2017.

[36] *Athens Trucks Data*. Accessed: Feb. 27, 2017. [Online]. Available: http://chorochronos.
datastories.org/

[37] J. Leskovec and A. Krevl. (Jun. 2014). *SNAP Datasets: Stanford Large Network Dataset Collection*. [Online]. Available: http://snap.stanford.edu/data

[38] *Thomas Brinkhoff: Network-based Generator of Moving Objects*. Accessed: Feb. 27, 2017. [Online]. Available: http://iapg.jade-hs.de/personen/brinkhoff/generator/faq.php

[39] A. R. Beresford and F. Stajano, "Mix zones: User privacy in location-aware services," in *Proc. IEEE PerCom Workshops*, Mar. 2004, pp. 127–131.

Hongbo Jiang (M'08–SM'14) received the Ph.D. degree from Case Western Reserve University in 2008. He was a Professor with the Huazhong University of Science and Technology. He is currently a Full Professor with the College of Computer Science and Electronic Engineering, Hunan University. His research concerns computer networking, especially algorithms and protocols for wireless and mobile networks. He currently serves as an Editor for the IEEE/ACM TRANSACTIONS ON NETWORKING, an Associate Editor for the IEEE TRANSACTIONS ON MOBILE COMPUTING, and an Associate Technical Editor for the *IEEE Communications Magazine*.

Ping Zhao received the B.E. degree from the Tianjin University of Science and Technology, China, in 2013. She is currently pursuing the Ph.D. degree with the School of Electronic Information and Communications, Huazhong University of Science and Technology, China. Her research interests are in the area of wireless networking, especially privacy protection in mobile networks.

Chen Wang (S'10–M'13) received the B.S. and Ph.D. degrees from the Department of Automation, Wuhan University, China, in 2008 and 2013, respectively. From 2013 to 2017, he was a Post-Doctoral Research Fellow with the Networked and Communication Systems Research Lab, Huazhong University of Science and Technology, China, where he joined the faculty and is currently an Associate Professor. His research interests are in the broad areas of wireless networking, Internet of Things, and mobile computing, with a recent focus on privacy issues in wireless and mobile systems.