Location Anonymity in Social Networks

M. Sc. Thesis
Submitted in Partial Fulfilment of the Requirements
For the Degree of
Master of Science
In
Computer Science & Engineering
Umm Al-Qura University

By:
Muneera Alotaibi
Student No: 43880168

Supervisor Name:
Dr. Waleed Alasmary

April 13, 2020

Acknowledgments

First of all, all praise and thanks are to Allah for granting me the opportunity, the ability, and the perseverance to accomplish this work. Thereafter, acknowledgement is due to Umm Al-Qura University for supporting this research. I am sincerely grateful to my supervisor Dr. Waleed Alasmary, for their technical knowledge and clear ideas regarding the direction of my research work. I would like also to thank Dr. Dawood Al Abri for his consistent collaboration throughout my thesis work. Finally, and especially, I would like to express my sincere gratitude and thanks to my parents, my husband for their endless love, patience, and prayers.

Abstract

Due to the wide availability of location-based services (LBSs), it becomes possible to trace the location of an individual by an adversary especially when the LBSs server is untrusted. k-anonymity is a well-known approach that is used to protect personal location privacy. This thesis introduces a novel user-based location selection scheme (UBLS) to hide the user location based on the k-anonymity. The proposed scheme uses the concept of dummy locations, but on top of that, it selects the dummy locations based on users that exist in these locations. We compare the proposed scheme with the well-known dummy location selection (DLS), enhanced dummy location selection (EDLS) and moving in neighborhood (MN) schemes, and it shows comparable performance to those schemes in terms of entropy metric, cloaking region metric and the location privacy level (LPL) metric. However, the proposed UBLS scheme significantly outperforms the DLS scheme in terms of the entropy metric whenever the number of users is low. And, our proposed scheme shows significant improvement over the EDLS scheme in terms of the LPL metric.

نبذة مختصرة

نظرًا للتوافر الواسع للخدمات القائمة على الموقع(LBSs) ، يصبح من الممكن تتبع موقع الفرد من قبل الخصم خاصة عندما يكون خادم LBSs غير موثوق به. إن k-anonymity هو نهج معروف يستخدم لحماية خصوصية الموقع الشخصي. تقترح هذه الرسالة مخططًا جديدًا لاختيار الموقع المستند إلى المستخدم (UBLS)لإخفاء موقع المستخدم بناءً على عدم الكشف عن الهوية. يستخدم المخطط المقترح مفهوم المواقع الوهمية، ولكن علاوة على ذلك، فإنه يختار المواقع الوهمية بناءً على المستخدمين الموجودين في هذه المواقع. قمنا بمقارنة المخطط المقترح مع مخطط اختيار موقع الدمية المعروف(DLS) ، ومخطط اختيار الموقع الوهمي المحسن (EDLS) ومخطط الانتقال بين الأماكن المتجاورة (MN)، ويظهر مخطط (UBLS) أداء مماثل لتلك المخططات من حيث مقياس الإنتروبي، ومقياس منطقة الخفاء الهوية ومقياس مستوى خصوصية الموقع (LPL) .ومع ذلك، فإن مخطط UBLS المقترح يتفوق بشكل كبير على نظام DLSمن حيث مقياس الانتروبي كلما كان عدد المستخدمين منخفضًا. ويظهر مخططنا المقترح تحسنًا كبيرًا على مخطط EDLS من حيث مقياس LPL .

Table of Content

List of Tables

\

List of Figures

List of Acronyms and Abbreviations

LBSs	Location Based Services
DLS	Dummy Location Selection
EDLS	Enhanced Dummy Location Selection
UBLS	User-Based Location Selection
MN	Moving in a Neighborhood
ALE	Attacker Location Exclusion
LPL	Location Privacy Level
EMD	Earth Mover's distance
GPUs	Graphics Processing Units
CPU	Central Processing Unit
FGcloak	Fine-Grained Spatial Cloaking
PkA	Probabilistic Framework of k-Anonymity
KDE	Kernel Density Estimation
AKDE	Approximate Kernel Density Estimation

1. Introduction

1.1. Overview of the Anonymity

Recently, the dependency of the smartphone devices applications on location has been dominating the Google Play and Apple stores [1]. Therefore, Location-Based Services (LBSs) are becoming essential in everyone's lifestyle. Furthermore, for example, in Uber/Uber-like applications [2], the user must reveal his/her location to request private drivers. Moreover, the need of the location information is not limited to the location-based applications, but the location information is also used in some social network applications such as Facebook [3] and Twitter [4]. Facebook uses the location information to let the user know about nearby friends [5], while Twitter uses the location to find tweets posted by people nearby [6].

Although disclosing the personal location enables many applications to provide user-tailored services, however, on the other hand, this practice might threat the user's privacy. For example, when an adversary can acquire the location of a certain user and he/she can use this information for tracking the user or identify the regular locations of the user visits. Therefore, location privacy is a crucial issue in mobile applications and social networks.

One of the existing approaches that are proposed to handle the location privacy issue is the k-anonymity scheme [7]. The k-anonymity scheme hides the individual's location by using a set of $k-1$ other locations. There are many schemes that use the k-anonymity to preserve the location privacy such as Dummy Location Selection (DLS) scheme and Enhanced Dummy Location Selection (EDLS) scheme [7]. Both schemes hide the real location of the user by using a set of dummy locations. The process of selecting the dummy locations in both schemes are based on k-anonymity.

1.2. Research Objectives

This thesis deals with $\boldsymbol{k}$-anonymity to protect the location privacy. The major goal and objectives of this thesis are summarized as follows.

- Introducing a novel scheme, namely, User-Based Location Selection (UBLS) using $\boldsymbol{k}$-anonymity. The UBLS scheme takes into consideration the user's query probability, which is different from the existing schemes such as DLS [7], Moving in a Neighborhood (MN) [8]. It chooses $\boldsymbol{k-1}$ dummy users whose query probabilities are close to the query

probability of the requester (i.e., the user who requests a service from the LBSs server), then it uses the $\boldsymbol{k-1}$ dummy users' locations to hide the location of the requester.

- Proposing an Attacker Location Exclusion (ALE) algorithm that can be used to attack many existing privacy-preserving schemes that do not take into consideration users' query probabilities. The ALE attempts to find the location of the requester among other $\boldsymbol{k-1}$ locations by excluding the locations that have low probabilities to be the requester's location. We use the ALE algorithm against the UBLS scheme and other existing schemes such as DLS [7], MN [8] to show which scheme is better in preserving location privacy when the LBSs server is malicious.
- Proposing a new metric denominated as a Location Privacy Level (LPL), and it qualifies the ability of the malicious LBSs server to reduce the privacy level of the requester.
- Evaluating the proposed UBLS scheme and compare it with different benchmarks schemes [7], [8].

1.3. Research Methodology

The methodology that is used in the novel algorithm in this thesis (UBLS scheme) is based on *k*-anonymity concept. The general idea of the proposed algorithm is an improvement of the DLS algorithm [7]. Instead of using dummy locations to anonymize the user's location as in the DLS algorithm, the proposed algorithm restricts the dummies to the users' locations whose query probabilities are close to the query probability of the target user. The proposed algorithm consists of two phases. In the first phase, the target user enters her/his query probability and her/his location. In the second phase, the proposed algorithm performs certain computations to select the dummy locations based on the query probabilities of other users. Also, the methodology takes into consideration the performance metrics that measure the anonymity level of the proposed algorithm and other related schemes. This thesis focuses on the well-known performance metrics which are entropy and cloaking region metrics in addition to the proposed metric (LPL metric).

1.4. Summary

This chapter gave an overview for the basic topic of the thesis which is preserving the location privacy of the user. Then the main objectives of the thesis were given. Moreover, the methodology was presented in this chapter.

2. Background and Literature Review

2.1. Introduction

Location privacy can be defined in different ways as described in [9]. For example, [10] defines the location privacy as a special type of information privacy focuses on how willing individuals are to share their locations with others. Whereas [11] defines the location privacy as the ability to prevent intruders from benefiting the location of an individual. Sometimes, the user's location must be revealed to the third party especially to acquire the LBSs. For example, some location measurement technologies use third-party infrastructure to find the user's location such as cell phone providers. Also, some network architecture of LBSs requires that the location must be transmitted to a vulnerable server such as requesting a bus schedule. However, location privacy may be violated when the LBSs server is a malicious server.

In [9], a comprehensive study of location privacy is presented. The study discussed the issues of the reasons for revealing the location, the peoples' awareness about their location privacy and the computational threats of location privacy. Furthermore, it shows some studies that focus on the awareness of people about the importance of location privacy such as [12] and [13]. The investigation study [12] applied 55 interviews in Finland and it found that most of the interviewees didn't worry about the privacy issue with LBSs. Moreover in [13], 15 volunteers were observed for their using of LBSs for five days. They found that the volunteers didn't take care of their privacy when using LBSs. The most important threats that exploit location privacy are the analysis of movement patterns and context inference. The location is usually traced from the GPS. In addition, the GPS may be used to infer other things about a person such as tracing the GPS to know the person's mode transportation (i.e. bus, foot, car).

There are many schemes that are used to limit the location privacy threats such as obfuscation and anonymity. The idea of the obfuscation scheme is based on degrading the quality of location measurements so that the location data is not accurate to reduce the location privacy threats. Furthermore, the anonymity can be classified into three types which are pseudonym, mix zone and k-anonymity. The pseudonyms are used to replace the associated name with an untraceable ID [9]. These pseudonyms frequently change to reduce the chance of inferring the identity of a person. However, the attacker could link the pseudonyms associated with requests of a single user's data. Therefore, instead of using the pseudonym scheme, it is

preferable to use a mix zone or k-anonymity schemes. In the mix zone scheme, people will only define their location in certain regions called "*application zones*" [9]. This is may help a user to mix him/her with other users in the same zone. On the other hand, the k-anonymity scheme defines a set of k people such that a person in this set cannot be distinguished from the other $k-1$ people [9]. Also, the k-anonymity has been used in different fields such as mobile computing and data publishing. Therefore, this thesis focuses on the related works that are based on k-anonymity as organized in table 1.

Table 1: The related works

Paper Title	**Publication Year**	**The κ-anonymity Field**
T-closeness: Privacy Beyond k-Anonymity and l-Diversity	2007	In data publishing
T-closeness: A New Privacy Measure for Data Publishing	2010	
Anonymity: Formalization of Privacy – k-anonymity	2013	
A GPU Algorithms for K-anonymity in Microdata	2019	
Truthful incentive mechanisms for k-anonymity location privacy	2013	In mobile computing
A Fine-Grained Spatial Cloaking Scheme for Privacy-Aware Users in Location-Based Services	2014	
Mobile r-gather: Distributed and Geographic Clustering for Location Anonymity	2017	
Achieving Effective k-Anonymity for Query Privacy in Location-Based Services	2017	
EPLA: efficient personal location anonymity	2017	
k-Anonymity Location Privacy Algorithm Based on Clustering",	2018	
A k-Anonymity Based Schema for Location Privacy Preservation	2019	

2.2. K-anonymity in Data Publishing

Microdata e.g. medical data or census data is a special type of data and usually used by government institutions for research and statistical study purposes. It is often categorized into 3 categories. The first category is the data that clearly identifies individuals such as the social security number. The second category is the quasi-identifiers that can be taken together to potentially identify individuals such as zip code, birthdate. Finally, the last category contains sensitive data of individuals such as salary. Therefore, the revealing of microdata may expose the individuals' privacy to attack by the adversary. The attacker can disclose the identity or the

attribute of the published microdata. Identity disclosure occurs when the adversary links a certain individual to a record in the published data while attribute disclosure occurs when new information of some individuals is disclosed. There are many proposed techniques such as k-anonymity and l-diversity.

The k-anonymity technique requires that each published record is indistinguishable with at least other $k-1$ records with respect to the quasi-identifiers as shown in fig. 1. The k-anonymity is secure against identity disclosure, but it stills insecure against attribute disclosure [14]. In addition to that, the k-anonymity technique is vulnerable to another four types of attacks which are an unsorted matching attack, complementary release attack, homogeneity attack and background knowledge attack. In the unsorted matching attack, the attacker uses two tables released from the same original table to link the dataset especially when the positions of tuples are identical in both tables. On the other hand, the complimentary release attack occurs when the second released table contains a subset of a previously released table. In this case, the attacker uses the attributes that are not part of quasi-identifiers to link the tables. Since the datasets are continuously changed over time, therefore, subsequent data sets are often released. Then the attacker can use these releases to link the preceding tables. Moreover, the homogeneity attack occurs when all the values of sensitive attributes are identical in the set of k records. While the background knowledge attack occurs when the adversary has some knowledge about a certain individual and he links this knowledge with the set of k records to leak the sensitive attributes of the individual. However, the k-anonymity technique has some advantages such as lower cost compared with the cryptographic techniques [15]. Whereas the concept of l-diversity is ensuring that there are at least l distinct values of sensitive attributes in each equivalence class (i.e. the records that contains the same values in the quasi-identifiers). The l-diversity is secure against a sketched attack which occurs when the sensitive attribute in the equivalence class has the same value. On the other hand, the l-diversity technique exposed by skewness and similarity attacks [16]. The skewness attack occurs when the equivalence class was skewed (i.e. the equivalence class satisfying the l-diversity but doesn't prevent the attribute disclosure). While the similarity attack occurs when the sensitive attributes in the equivalence class are distinct but semantically similar.

Due to the limitations of k-anonymity and l-diversity, Ninghui Li et al. [17] proposed a new technique for protecting individuals' privacy called t-closeness. Th t-closeness concept ensures that the distribution of a sensitive attribute in any equivalence class is similar to the distribution of the attribute in the overall table (i.e. the difference value between the two distributions should not exceed a threshold t). Furthermore, the t-closeness technique may produce a table with an essential information loss. Therefore, Ninghui Li et al. [17] improves the t-closeness technique to (n, t)-closeness. Suppose there is an equivalence class E_1 then the (n, t)-closeness concept occurs when E_1 has a set of records called E_2 and it contains at least n records. Then the distance between the two distributions of the sensitive attribute in E_1 and E_2 is no more than a threshold t. Their research tested the (n, t)-closeness technique based on three factors. The first factor is the privacy protection while the second and the third factors are utility preservation and efficiency respectively. Also, the dataset used in the testing is the ADULT dataset from the UC Irvine machine learning repository. The results showed that the t-closeness and (n, t)-closeness are better than the k-anonymity and l-diversity. Moreover, the (n, t) closeness is slower than the other techniques. However, the results proofed that the table produced by (n, t)-closeness has better utility than both l-diversity and t-closeness tables. Furthermore, N. Li et al. [9] defined the t-closeness as the variation between the distribution of a sensitive attribute in the equivalence class and the distribution of the sensitive attribute in the whole table should not be more than a threshold t. This definition considered the t-closeness as a refinement of the l-diversity. The distance or threshold t can be measured by Earth Mover's Distance (EMD) [19]. The t-closeness protects against skewness and similarity attacks. However, t-closeness has an important drawback which is limiting the amount of useful information that is published [20].

Also, a research in 2013 [21] analyzed the anonymity of a dataset collected from Android devices. Then the dataset of android devices transferred into the SQLite database. The SQL-queries results showed that the android devices dataset is not anonymous. To improve the anonymity of the android devices dataset, this research used UTN Anonymization Toolbox to transfer the dataset into the k-anonymous state.

In 2019 Di Pietro et al. [22] used three Graphics Processing Units (GPUs)-based parallel approaches for a micro aggregation technique. The micro aggregation technique is a special technique used to achieve k-anonymity for numerical microdata. The micro aggregation

technique guarantees individuals' privacy when releasing the microdata to the third parties. The purpose of using the three GPUs is to speed-up the execution of privacy-preserving algorithms such as micro aggregation technique. According to the results, the GPU is not efficient when the database is small. It consumed more time than the normal Central Processing Unit (CPU). However, the performance of the GPU is better with larger databases.

Original table

	ZIP Code	Age	Disease
1	47677	29	Heart Disease
2	47602	22	Heart Disease
3	47678	27	Heart Disease
4	47905	43	Flu
5	47909	52	Heart Disease
6	47906	47	Cancer
7	47605	30	Heart Disease
8	47673	36	Cancer
9	47607	32	Cancer

3-anonymous table

	ZIP Code	Age	Disease
1	476**	2*	Heart Disease
2	476**	2*	Heart Disease
3	476**	2*	Heart Disease
4	4790*	≥ 40	Flu
5	4790*	≥ 40	Heart Disease
6	4790*	≥ 40	Cancer
7	476**	3*	Heart Disease
8	476**	3*	Cancer
9	476**	3*	Cancer

Figure 1: K-anonymity in data publishing field [17]

2.3. K-anonymity in Mobile Computing

Due to the importance of location privacy in mobile computing, many works proposed to protect location privacy in mobile computing based on the k-anonymity such as [23], [24], [25], [26], [27], [28] and [29]. The proposed mechanism in 2013 [23] simulates the k-anonymity auction as a single-round sealed-bid double auction. It classifies mobile users into 3 groups. The first group is buyers while the second group is the sellers. The last group is auctioneer which represents central authority. It considers both buyers and sellers as agents when it does not distinguish them. The buyers offer prices for the desired k-anonymity privacy, while the sellers offer prices for participating in the anonymity set. Also, the price offered by each agent is private to the agent itself, and no agent is aware of the prices offered by others. The price that offered by the buyer called W_b while the price that offered by a seller called W_s. The first step

of the auction mechanism starts when the buyers and sellers submit their prices to the auctioneer. Then the auctioneer decides the winning buyer set W_b and the winning seller set W_s, such that $|W_b| + |W_s| \geq k$. Also, the auctioneer determines the payment charged to each buyer and the payment paid to each seller. The auction mechanism was implemented on a Linux machine with 3.2 GHz CPU and 16 GB memory. The testing of the mechanism measured the performance. The performance metric is measured by running time and the number of winning buyers. The testing showed that the mechanism taken more time in the case of the users' privacy degree requirements are different. Also, the results showed that the number of winning buyers increases when there are more sellers participating in the auction.

LBSs are one of the most important services that are provided to mobile users. However, these services require mobile users to send a query to the untrusted LBSs server to get the service. The submitted query contains the user's identifier, exact location, the query interest as well as the query range, …etc. which may cause a location privacy issue. To address this problem, B. Niu et al. [24] proposed a Fine-Grained Spatial Cloaking (FGcloak) scheme based on k-anonymity technique. The idea of FGcloak scheme depends on the idea of Hilbert curve with some modifications to effectively achieve the k-anonymity privacy protection. The steps of the FGcloak scheme as follows: first the Hilbert curve applies on a map (contains $n\ x\ n$ cells) then the Hilbert curve is modified according to the query probability of each cell and the modifications represented as points as illustrated in fig. 2. In the second step, the modified Hilbert curve separated into k segments such that the user's real location is in one segment. Then the $k-1$ dummy locations are chosen from other $k-1$ segments. This step guarantees the k-anonymity with a bigger cloaking region. FGcloak algorithm was tested according to three factors which are cloaking region, entropy and the exchange ratio σ. The cloaking region measured based on the k value and the simulation time. The results showed that the cloaking region of the FGcloak algorithm is much larger than other existing solutions such as EDLS [7] and SMILE [30]. However, the entropy value of the EDLS [7] is better than its value of the FGcloak algorithm. Finally, the exchange ratio σ factor represents the fraction of time during which a user exchanges information with other encountered users. Moreover, a bigger value of σ leads to higher communication cost. Therefore, the evaluation of the FGcloak algorithm demonstrated that the FGcloak algorithm guaranteed an efficient value of σ factor.

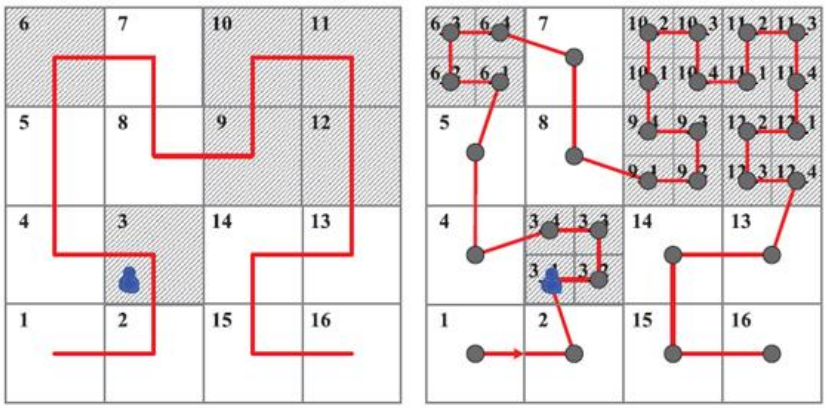


Figure 2: FGcloak scheme [24]

Nowadays, mobile devices had become an essential part of our life. Many mobile services used the GPS to collect and record the locations of mobile devices that may breach the person's location privacy. Therefore, a research proposed by [25] studied the mobile r-gather clustering problem under the condition of enabling the anonymity in LBSs to protect the privacy. In mobile r-gather clustering problem, there are n nodes and the problem is how to cluster the nodes into groups of at least r nodes such that the largest diameter of the clusters is minimized as illustrated in fig. 3. Regarding the mentioned problem, the authors proposed a distributed algorithm that generates compact clusters, within an approximation factor of the minimum cluster diameter possible. When they applied the proposed algorithm on mobile nodes, they considered two cases: maintaining dynamic clustering of the locations of nodes and offline clustering of paths. In the first case, the nodes continuously change their locations and the cluster must be updated according to these changes. While in the second case, according to the recorded paths in each node, the nodes with similar paths are clustered into the same group. The distributed algorithm tested under the Euclidean metric for dynamic/mobile nodes. The test measures the robustness and the stability of the algorithm. Their results show that the algorithm is robust against the noise/outliers because it didn't affect by outliers since the size of the cluster containing a node is determined only by the local node density. While it is more stable because the number of changes in the clustering membership is approximately $O(n^2)$.

The LBSs provide comfort to mobile users in several aspects such as communication, information exchange, social activities, and so on. However, with the LBSs, the location privacy

problem arises. Therefore, there are many proposed approaches to protect query privacy such as cloaking based on k-anonymity. The cloaking depends on the trusted third-party server. The steps of the cloaking technique as follows: first, the user u sends an LBSs query to the trusted server. The trusted server generates a cloaking region that contains at least k users including u and sends the cloaking region to the LBSs server as a response. However, the cloaking technique has an important weak point which is the single point of failure. To recover this drawback, some solutions are proposed to be client-based such as DLS [7] algorithm that based on entropy metric. A proposed paper in 2017 [26] focuses on query privacy for preventing the exploiting of users' query contents. It proposed effective k-anonymity based solutions for query privacy in LBSs. Then this paper analyzed a recent proposed algorithm DLS [7] based on the Probabilistic Framework of k-Anonymity (PkA) framework. Also, it proposed two algorithms called MEE and MER as an enhancing to the privacy metrics. Both the MEE and MER requires that the prior probability distribution of query interests is given. The MEE and MER divide all the query interests into groups in which members have adjacent prior probability, and each k reported query interests are selected from the same group. In general, MEE maximizes the entropy-based metric, while MER minimizes the differential mannered privacy metric. Also, this measures the proposed algorithm based on two properties which are no more leakage and k-effectiveness. Then the MEE and MER algorithms evaluated on real-life data sets and synthetic query interest distributions. The results showed that the proposed algorithms provide the property of k Effectiveness which is absent from DLS [7]. Also, the proposed algorithms satisfy the property of no more leakage.

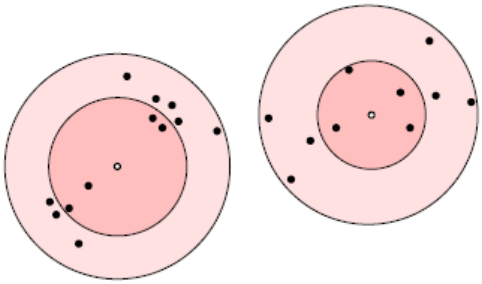


Figure 3: Mobile r-gather clustering problem [25]

Personal privacy is one of the important topics since it can be violated in different ways. One of these ways uses the location to expose the personal privacy. Therefore, a lot of

researchers proposed techniques for protecting users' location privacy in LBSs such as EPLA. EPLA [27] is an efficient personal location anonymity technique used to protect users' location privacy. The basic idea of the EPLA method is anonymizing the user's location by selecting alternative dummy locations based on the probability of visiting locations by the user as in fig. 4. Practically, EPLA divided into two phases. In the first phase, space is portioned into cells and determined the dummy locations candidate set P. Then computing the personal visit probability of each location p_i in the candidate dummy locations set P by two methods which are Kernel Density Estimation (KDE) and Approximate Kernel Density Estimation (AKDE). AKDE is an enhancing of KDE. While in the second phase, conducting location anonymity set as the user's location in LBSs by using k-anonymity. The datasets that used to test the EPLA are Gowalla and Foursquare datasets. The testing process of EPLA measured the performance and security level of EPLA. The experiment results showed that the EPLA had better performance since its computation cost was reduced to $O(|P|n)$ (where $|P|$ is the number of elements in the set P and n is the number of sampling user's visited locations) when using AKDE method. Moreover, in measuring the security level of EPLA, a special type of the adversaries was considered who can get the user's current queries and a lot of his visited locations. Also, the results presented that the EPLA is secure against this kind of adversary.

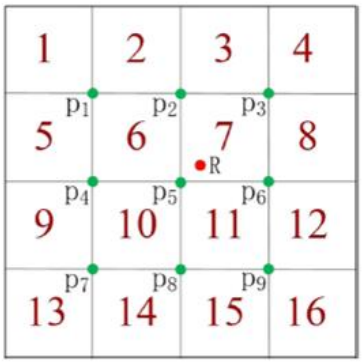


Figure 4: The EPLA algorithm [27]

One of the controversial issues is how to balance the conflict between the privacy-preserving security and the quality of the service caused by the accuracy of the location information, especially in LBSs. Therefore, a research paper in 2018 [28] proposed a k-anonymity location privacy algorithm based on clustering to serve this issue. This algorithm combines between k-anonymity and clustering. The general steps of the proposed algorithm as follow: first,

arranging the anonymous group based on clustering. Then choosing a node that has the largest density distribution as an anchor. The role of the anchor is taking account of the needs of most users' query requirements to improve the query accuracy after the anonymity. Finally, eliminating the outliers from the anonymous group if they are existing. The purpose of the outliers eliminating process is to make the anonymous group more convenient to meet user's privacy requirements and reduces the impact of anonymity on the quality of the services. The proposed algorithm [28] was formulated as a central server structure. The structure includes client, anonymous server and LBSs server as illustrated in fig. 5. First, the client sends an encrypted query request M and the privacy requirements to the anonymous server. The privacy requirements consist of the privacy degree k (the minimum number of users in the anonymous group) and the minimum size of an anonymous region Amin. Second, the anonymous server applies the k-anonymity algorithm based on clustering to produce the anonymous results set C and sends C to LSBs server. Third, the LSBs server processes C according to the location of the anchor and the information of the query issued by the client to produce the candidate results R and sends it to the anonymous server. Finally, the anonymous server filters R according to the actual location of the client and replies to the client. The complexity of the k-anonymity algorithm [28] based on clustering is $O(n^2)$. It implemented in Java and tested based on the anonymous success rate, anonymous processing time and query accuracy. The anonymous success rate represents the rate of the users who have successfully received anonymous queries. The results showed that the increase of the privacy degree k will decrease the anonymous success rate. According to the results, the anonymous processing time reduced when the number of users increases. Also, query accuracy decreased when the number of users increases.

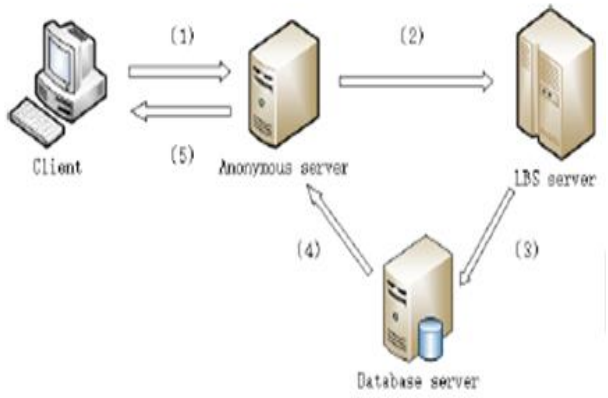


Figure 5: The ***k***-anonymity algorithm based on clustering [28]

In 2019 F. Fei et al. [29] proposed a two-tier schema for preserving privacy based on k-anonymity with guaranteeing a minimum cost. The idea of the two-tier schema based on distributing the users into groups. The purpose of the distribution is to increase the privacy level for each group. In each group, there is a proxy which is responsible for generating $k-1$ dummy locations and sends them to the LBSs provider. Moreover, the proxy shares the results that are returned from the LBSs provider with the other users in the same group. Also, Fei et al. [29] proposed an auction mechanism to specify the payment of each user to the proxy as the compensation. The two-tier schema evaluated according to the entropy metric and the privacy level cost ratio. According to the entropy metric, the two-tier schema achieved a high level of privacy. The privacy level cost ratio represents the rate between the total values of entropy for all users and the cost for calculating the entropy value for each user. The two-tier schema took a minimum cost to achieve a high level of privacy.

2.4. Location Privacy Metrics

Due to the importance of the location privacy problem, many schemes have been proposed to deal with this problem as we mentioned in the previous subsections. To quantify the effectiveness of these schemes, several location privacy metrics have been proposed. Most of them are uncertainty-based or entropy-based metrics. The uncertainty-based metric [31] measures the ability of the adversary to differentiate the real location of the user from other locations in the anonymity set. On the other hand, the entropy-based metric measures the quantity of querying the locations in the anonymity set [7]. It can be calculated as follows.

$$H = -\sum_{i=1}^{k} p_i \log(p_i) \qquad (1)$$

where p_i represents a probability that a possible location has been queried in the past. The maximum value of entropy gives a high level of privacy (i.e. the highest uncertainty to distinguish the real location of an individual among the other the locations in anonymity set). The maximum entropy is achieved when all the k possible locations have the same $p_i$which equals to $\frac{1}{k}$ and the maximum value of entropy will be $H_{max} = \log k$. Another metric that is used in this regard is the privacy area or cloaking region. The privacy area defines the size of the area that covers all the locations in the anonymity set. Bigger privacy region leads to more

anonymity. Our work uses the entropy and cloaking region metrics to measure the privacy level as well as a new novel privacy metric as we shall explain in chapter four.

2.5. Summary

Background of the problem was presented in this chapter —namely, preserving location privacy during the benefiting of LBSs. One of the suggested solutions to handle this problem is the k-anonymity technique. The use of the k-anonymity technique was reviewed in this chapter in different fields such as data publishing and mobile computing. Also, this chapter displayed the existing schemes that were using the k-anonymity in both fields. Finally, a brief review of location privacy metrics that quantifies the effectiveness of the existing proposed schemes in terms of the location privacy level was shown.

3. System Model and Anonymity Scheme

3.1. Introduction

One of the approaches that are proposed to protect personal location privacy is k-anonymity. The k-anonymity approach uses an anonymous set that consists of k people with the aim of making any person who belongs to this set indistinguishable from all other $k-1$ people. However, the k-anonymity approach has some limitation such as the single point of failure since all the burden of the operations is on the location anonymizer. Therefore, there are some approaches that allow the user (i.e. the service requester) to select the k locations, which are called dummy locations, instead of the location anonymizer. The MN scheme [8], the DLS scheme [7] and the EDLS scheme [7] are examples of these approaches. We use a unified framework to express the three schemes. Hence, it becomes easier for other researchers to reimplement and test the results. The notation used in this thesis is shown in table 2.

Table 2: Notation used in this thesis

m	Number of users in the map
k	Number of locations in the anonymity set
l_{real}	Real location of the target user (i.e the user who sends the anonymity set to LBSs server)
u_i	Query probability of user i
q_{ij}	Query probability of cell ij in the map
p_i	Normalized query probability
n	Number of locations
r	Number of rounds chosen by the user
C	The anonymity set
c_i	A certain location i in C
s	Number of queries of a particular user
a	Parameter was chosen by the user
x	X-coordinate in the map
y	Y-coordinate in the map

3.2. Unified Framework for the Benchmark Schemes

The MN scheme was proposed by H. Kido et al. [8]. The basic idea of the MN scheme is based on selecting the dummy locations randomly. The first dummy location is selected randomly. Then, the second dummy location is chosen randomly from the neighbors of the first dummy location that are within a certain range and so on as shown in fig. 6. Algorithm 1 describes the general steps of the MN scheme. We can see that the two main lines in the

algorithm are 6 and 7, which select the x and y coordinates of the dummy location and store it in the anonymity set C. The selection process is done randomly within a specific area. The area is related to the location of the previous dummy ± a as shown in lines 6 and 7 in algorithm 1 where a is a parameter chosen by the user.

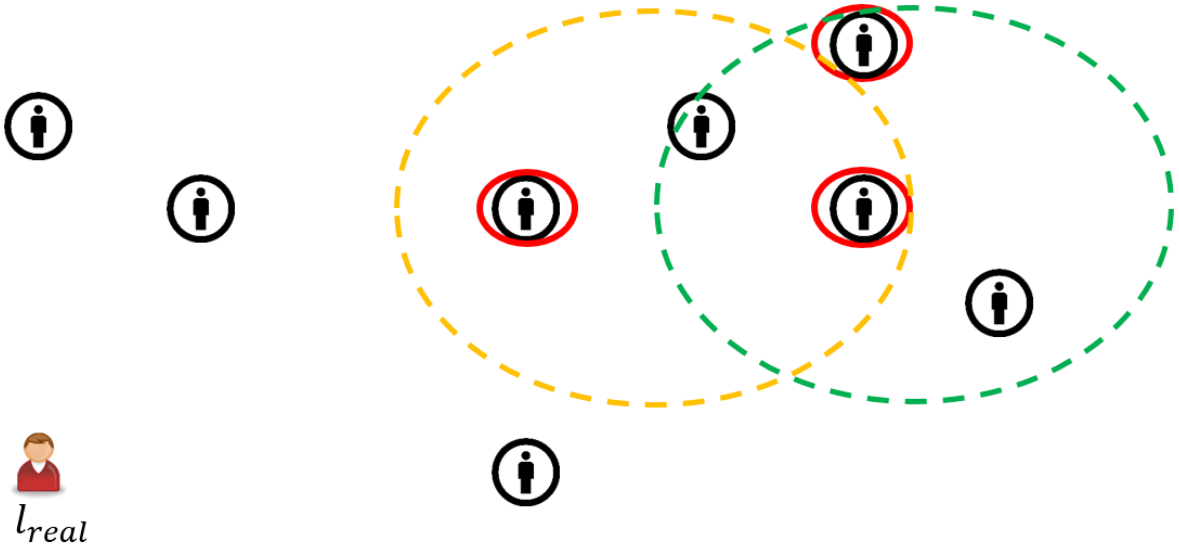


Figure 6: MN scheme

Algorithm 1: Moving in a Neighborhood Scheme

Input: k, a, l_{real}
Output: The anonymity set C

1 Store l_{real} in $C[1]$
2 Set x to a random number
3 Set y to a random number
4 Store the location with x and y coordinates in $C[2]$
5 **for** (i=3; i <= $k-2$; i++) **do**
6 Set x to a random number that belongs to the interval ($C[i-1].x - a, C[i-1].x + a$)
7 Set y to a random number that belongs to the interval ($C[i-1].y - a, C[i-1].y + a$)
8 Store the location with x and y coordinates in $C[i]$
9 **end**
10 **Return** C

The DLS scheme was proposed by B. Niu et al. [7] and it is designed to achieve k-anonymity for users in LBSs. The DLS scheme assumes that the adversary can exploit the side information. The DLS selects the dummy locations based on the maximum value of the entropy metric as illustrated in algorithm 2. We can see that in the line 1 the DLS orders the locations in the map in ascending order based on their query probabilities. Then in the line 2, it chooses $2k$ locations from the sorted list which consists of k locations before the real location l_{real} and k locations after the real location l_{real}. After that, it selects $k-1$ dummy locations from the $2k$ locations as in the line 6 and repeats this step a times to get an anonymity set C with maximum entropy.

Algorithm 2: Dummy Location Selection Scheme

Input: k, a, q of all cells in the map, l_{real}, q_{ij} of the cell of the target user

Output: The anonymity set C with H_{max}

1 Sort the users based on their q in ascending order
2 Select $2k$ dummy candidates among which k candidates are to the right and k candidates are to the left of the target user in the sorted list
3 $H_{max} = 0, C_{max} = \emptyset$
4 **for** (i=1; i <= a; i++) **do**
5 Store l_{real} in the anonymity set C_i
6 Choose $k-1$ dummy locations randomly from $2k$dummy candidates set
7 Calculate the p_i for each element in the C_i
8 $H_i = -\sum_{i=1}^{k} p_i \log(p_i)$
9 **If** ($H_{max} < H_i$)
10 $H_{max} = H_i$
11 $C_{max} = C_i$
12 **end**
13 **end**
14 **Return** C_{max} with H_{max}

Furthermore, the EDLS scheme [7] is an extension of the DLS scheme. The EDLS is based on the maximum value of both the entropy and the cloaking region metrics as shown in algorithm 3. The steps from 1 to 13 are executed as the DLS scheme but it produces an anonymity set C of $2k$ dummy locations instead of $k-1$ dummy locations as in the DLS

scheme. Then, the EDLS scheme chooses $k-1$ dummy locations from the $2k$ dummy locations that cover a big cloaking region as illustrated in the steps 14 to 19.

The difference between the original DLS and the EDLS is in the selection process of dummy locations. The EDLS selects the dummy locations that are spread far away as much as possible. However, the EDLS may reduce the quality of the service that is submitted by the LBSs server, since it chooses the dummy locations that are far away from the real location l_{real}.

Algorithm 3: Enhanced Dummy Location Selection Scheme

Input: k, a, q of all cells in the map, l_{real}, q_{ij} of the cell of the target user

Output: The anonymity set C

1 Sort the users based on their q in ascending order
2 Select $4k$ dummy candidates among which $2k$ candidates are to the right and $2k$ candidates are to the left of the target user in the sorted list
3 $H_{max} = 0, C = \emptyset$
4 **for** (i=1; i <= a; i++) **do**
5 Choose $2k$ dummy locations randomly from $4k$dummy candidates set C_i
6 Calculate the p_i for each element in the C_i
7 $H_i = -\sum_{i=1}^{k} p_i \log(p_i)$
8 **If** ($H_{max} < H_i$)
9 $H_{max} = H_i$
10 $\hat{C} = C_i$
11 **end**
12 **end**
13 Store l_{real} in the anonymity set C
14 **for** (i=1; i <= $k-1$; i++) **do**
15 Choose a dummy location $\acute{c}_i$ from $\hat{C}$ with the maximum cloaking region based on $\frac{\prod_{c_j \epsilon C} d(c_i, c_j)}{\sum_{c_i \epsilon \hat{C}} \prod_{c_j \epsilon C} d(c_i, c_j)}$
16 Add $\acute{c}_i$ to C
17 Remove $\acute{c}_i$ from $\hat{C}$
18 **end**
19 **Return** C

3.3. UBLS Scheme

The User-Based Location Selection scheme, or shortly UBLS, aims at hiding the real location of the user by carefully selecting the set of dummy locations based on the query probabilities of the users. Our approach is similar to DLS but in the case of DLS, it uses the query probability of the location q_{ij} whereas our scheme uses the query probability of the user u_i as illustrated in fig. 7. Our approach results in better resistance to the attacks launched by a malicious server that exploits its knowledge of the users' query probabilities to narrow down the real location of the user as we shall see in chapter four.

To explain the UBLS scheme, let us consider a map of size $n\,x\,n$ cells. Each cell (i,j) represents a location in the map and the query probability of that cell is q_{ij} . Also, there are m users distributed over the cells of the map and we consider the user who sends the anonymity set C to the LBSs server as a target user. Each user has query probability u_i. The probabilities q_{ij} and u_i can be calculated as follows.

$$u_i = \frac{\#\ of\ queries\ of\ user\ i}{\#\ of\ queries\ of\ all\ users\ on\ the\ map} \qquad (2)$$

where

$$\sum_{i=1}^{m} u_i = 1\ \ and\ \sum_{i=1}^{n}\sum_{j=1}^{n} q_{ij} = 1$$

$$q_{ij} = \frac{\#\ of\ queries\ in\ cell\ ij}{\#\ of\ queries\ in\ whole\ map}, 1 \le i,j \le n \qquad (3)$$

The following steps are an explanation of the UBLS scheme steps in more details:

.

1) In the first step, the target user chooses the size of the anonymity set k, which is closely related to the user's location privacy and the system overhead. Specifically, a bigger k leads to a higher degree of anonymity level but with more overhead. Also, the user needs to decide the number of rounds r that the UBLS scheme uses to compute the anonymity set C. This aims to choose C that maximizes the value of the entropy metric.
2) The UBLS scheme reads all the obtained users' query probabilities and then sorts all users based on the users' query probabilities in ascending order.

3) Then, it chooses k users right before and k users right after the target user from the sorted list to form a $2k$ candidates set.
4) After that, the UBLS scheme manipulates the C set. The C set consists of the target user's location l_{real} and $k-1$ locations of the dummy users that are chosen randomly from the $2k$ candidates set.
5) Then, the UBLS scheme checks that all the k users' locations in the C set are of different cells in the map. If there are repeated locations in the C, the UBLS scheme will choose another $k-1$ dummy users randomly from the $2k$ candidates set to form C.
6) Then normalized query probabilities of each user in the C set will be calculated by equation 4.

$$p_i = \frac{u_i}{\sum_{i=1}^{k} u_i}, i = 1,2, \ldots . k \qquad (4)$$

7) The entropy value H of the C set will be computed using equation 1.
8) The UBLS scheme repeats the steps 4 to 17 r times to get the C_{max} set with H_{max} as maximum as possible.

Algorithm 4: User-Based Selection Scheme

Input: k, r, u of all users in the map, l_{real}, u_i of the target user
Output: The anonymity set C that maximize H

1 Sort the users based on their u in ascending order
2 Select $2k$ dummy candidates among which k candidates are to the right and k candidates are to the left of the target user in the sorted list
3 $H_{max} = 0, C_{max} = \emptyset$
4 **for** (i=1; i <= r; i++) **do**
5 Store l_{real} in the anonymity set C
6 Choose $k-1$ dummy users randomly from $2k$ dummy candidates set and store their locations in the C
7 **while** (there are identical locations in the C)
8 Repeat step 6
9 **end**
10 Calculate the p_i for each element in the C
11 $H = -\sum_{i=1}^{k} p_i \log (p_i)$
12 **If** ($H_{max} < H$)
13 $H_{max} = H$

14 $C_{max} = C$

15 **end**

17 **end**

18 **Return** C_{max}

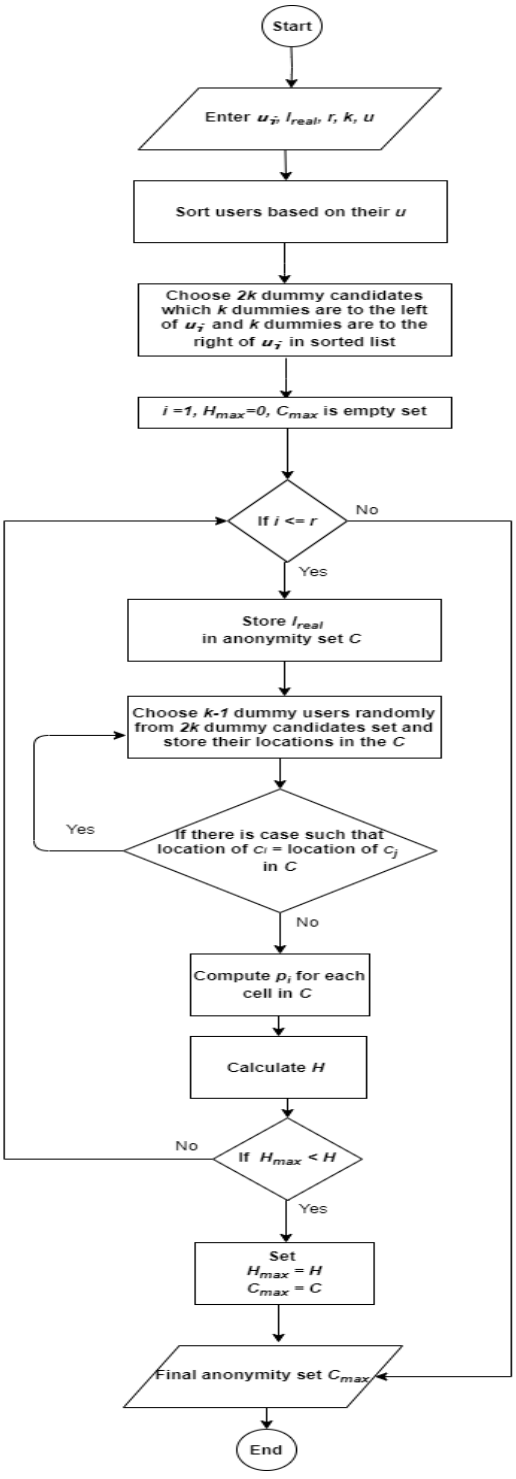


Figure 7: The flowchart of UBLS scheme

As we mentioned in algorithms 2 and 4, there are several differences between the DLS scheme and the UBLS scheme. The query probability $\boldsymbol{u_i}$ of the target user is taken as input in the UBLS scheme whereas the DLS scheme takes the query probability $\boldsymbol{q_{ij}}$ of the target user's location. Furthermore, there is a difference in the sorting procedure, the DLS scheme sorts the locations of the map based on their query probabilities $\boldsymbol{q}$ but the UBLS scheme sorts the users based on their query probabilities $\boldsymbol{u}$. Besides, the UBLS scheme checks the repeated locations in C while this step doesn't exist in the DLS scheme. finally, the normalized query probabilities $\boldsymbol{p}$ computed according to equation 4 in the UBLS scheme, on the other hand, the DLS calculated the normalized query probabilities $\boldsymbol{p}$ based on equation 5.

3.4. Adversary Model

In this thesis, we mainly consider an untrusted LBSs server as the adversary, which tries to infer the actual location of the target user from the anonymity set C. We assume that the LBSs server has the maximum advantage (i.e. it is able to obtain all global information such as number of queries for the target user, number of total queries for a certain location, number of users inside a certain location in the map…etc.). Therefore, we propose the ALE algorithm that can be used by a malicious server to narrow down the real location of the user. The ALE algorithm operates as a checker by passing a certain location c_i in the anonymity set C through some conditions. If c_i satisfies one condition from these conditions, then c_i will be excluded from the consideration as illustrated in fig. 8. Hence, the ALE algorithm improves the attacker's ability to identify the real location in the anonymity set C by eliminating some dummy locations through the incorporation of the query probabilities of the users.

Suppose that the LBSs server knows the number of queries of the target user $\boldsymbol{s}$ and each user on the map queries the server (i.e. there is no user who doesn't send a query to the LBSs server). There are four conditions used by the ALE (algorithm 5) to exclude a location $\boldsymbol{c_i}$ from the set $\boldsymbol{C}$. These conditions are explained in more detail next.

1) **First condition:** the ALE will check the number of users inside the location $\boldsymbol{c_i}$ and if it is equal to zero then the ALE will exclude the location $\boldsymbol{c_i}$. The reason for the excluding should be obvious in this case as the target user cannot be in a location that is empty of users.

2) **Second condition:** when the number of users inside the location $\boldsymbol{c_i}$ is equal to 1 then the ALE will compare the total queries of the location $\boldsymbol{c_i}$ and the number of queries of the target user $\boldsymbol{s}$. If they are not equal, the location $\boldsymbol{c_i}$ will be excluded. If the target user were inside $\boldsymbol{c_i}$, the total number of queries of the location would necessarily be equal to the number of queries of the target user $\boldsymbol{s}$ since there is only one user inside that location.

3) **Third condition:** in this condition, the total queries of the location $\boldsymbol{c_i}$ is compared with the number of queries of the target user $\boldsymbol{s}$, and if $\boldsymbol{s}$ is bigger than the total queries of the location $\boldsymbol{c_i}$, then $\boldsymbol{c_i}$ will be excluded. If the target user were inside $\boldsymbol{c_i}$, the total queries of $\boldsymbol{c_i}$ would be at least equal to $\boldsymbol{s}$.

4) **Fourth condition:** to understand this condition, we first note that a location contains $\boldsymbol{h}$ users, then the total number of queries of that location will be at least $\boldsymbol{h}$ (since each user queries the server at least one time). Hence, if the target user exists in the location $\boldsymbol{c_i}$, then the following condition should hold:

$$\boldsymbol{Q_{c_i} \geq s + h - 1}$$

where $\boldsymbol{Q_{c_i}}$ is the number of queries of $\boldsymbol{c_i}$.

Inversely, if the total number of queries of the location $\boldsymbol{c_i}$ is less than $\boldsymbol{s + h - 1}$, then the target user cannot be in the location $\boldsymbol{c_i}$. The fourth condition makes use of the last observation to exclude locations.

Algorithm 5: Attacker Location Exclusion

Input: The anonymity set C, the number of queries of the target user s
Output: The anonymity set C^r reduced

```
1  Exclusion set C' = ∅
2  for ( i=1; i <= size(C); i++) do
3      if (c_i.number of users == 0)
4        Add c_i to C'
5      end
6      if (c_i.number of users == 1)
7        if (c_i.total of queries ! = s)
8          Add c_i to C'
9        end
```

10 **end**
11 **if** ($s > c_i.total\ of\ queries$)
12 Add c_i to C'
13 **end**
14 **if** (($c_i.total\ of\ queries - \boldsymbol{s}$) < ($c_i.number\ of\ users - 1$))
15 Add c_i to C'
16 **end**
17 **end**
18 $C^r = C - C'$
19 **Return** C^r

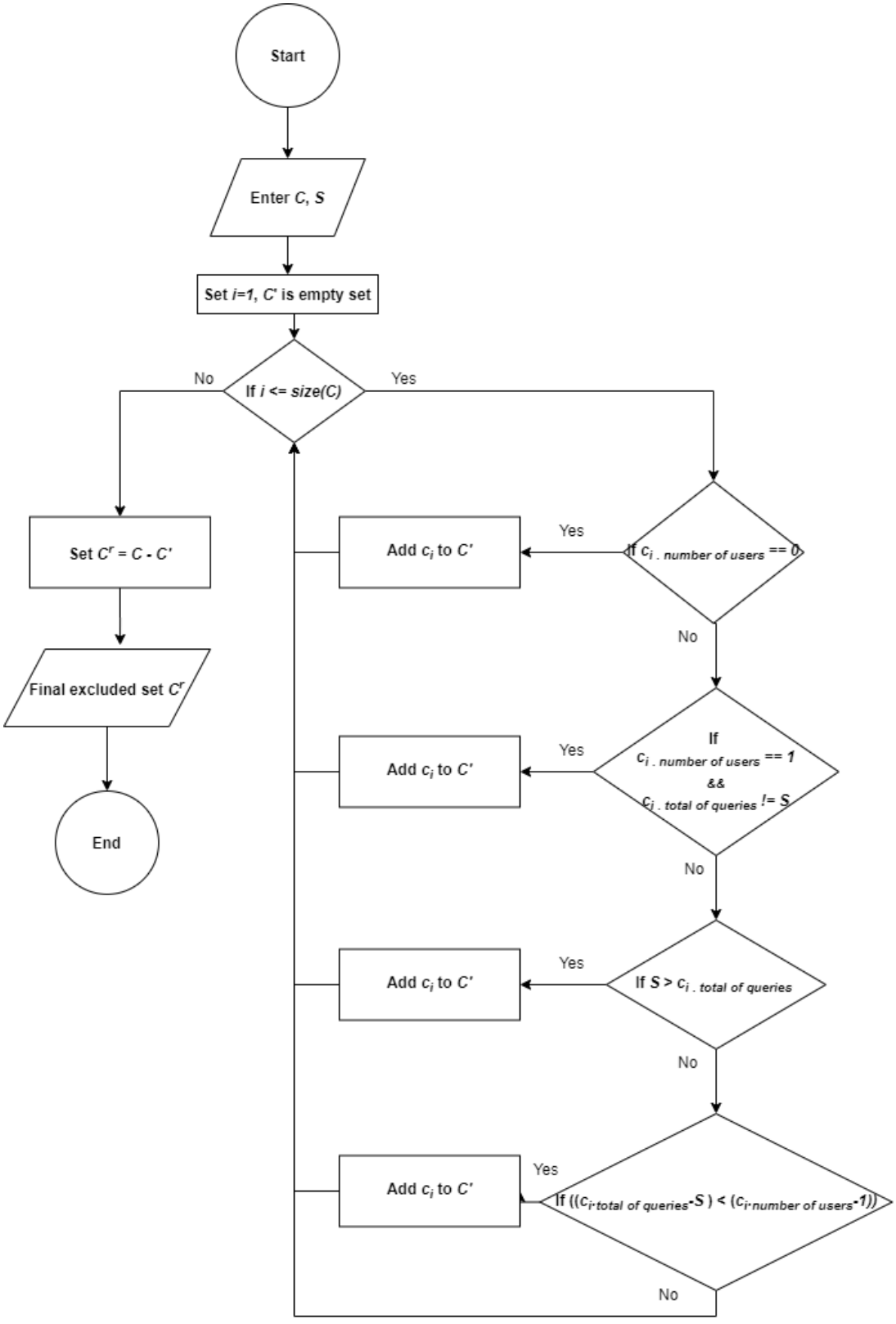


Figure 8: The flowchart of ALE algorithm

3.5. Summary

In this chapter, the main unified framework of the thesis was presented. Some benchmark schemes that selected the dummy locations based on k-anonymity technique at the client-side were discussed in detail. Also, the proposed scheme namely, the UBLS scheme was debated and it was used to protect location privacy. Finally, the adversary model used by the malicious server to expose the real location of the user was shown in this chapter.

4. Simulation Results

4.1. Performance Metrics

This chapter presents the performance metrics that are used to evaluate the performance of the UBLS scheme and the other related schemes.

The entropy-based metric represents the quantity of the querying the locations in the anonymity set. The entropy value is calculated using equation 1. For the UBLS scheme, we computed p_i as in equation 4. For the benchmark schemes (i.e. the MN, the DLS, and the EDLS schemes), we computed p_i using the following equation.

$$p_i = \frac{q_i}{\sum_{i=1}^{k} q_i}, i = 1,2, \dots . k \qquad (5)$$

Note that, these schemes choose the locations in the anonymity set based on their query probabilities q_i regardless of the number of users inside these locations.

The cloaking region represents the area that covers all the locations in the anonymity set. We quantify the cloaking region using two metrics:

(i) The product of distances.

(ii) The total area.

The product of distances represents the product of distances between every pair of locations in the anonymity set and it can be calculated as follows.

$$\prod_{i=1}^{k-1} \prod_{j=i+1}^{k} d(c_i, c_j) \qquad (6)$$

where c_i and c_j represent locations in the anonymity set.

On the other hand, we considered the total area as the largest rectangle that contains all locations in the anonymity set. Therefore, the total area calculated as follows.

$$(i_{max} - i_{min}) * (j_{max} - j_{min}) \qquad (7)$$

where i_{max} and j_{max} represents the x-coordinate and the y-coordinate of rightmost location in the anonymity set respectively. Furthermore, the x-coordinate and the y-coordinate of the leftmost location in the anonymity set are defined by i_{min} and j_{min} respectively.

The Location Privacy Level (i.e. LPL) metric: we introduce this new metric to qualify the ability of the adversary to reduce the level of privacy experienced by the user. We assume that the LBSs server uses the ALE algorithm to exclude some dummy locations. We define the LPL as follows.

$$LPL = ln(size(C^r)) \qquad (8)$$

Note that the LPL value increases with the increase of the reduced set C^rsize. Recall that C^r represents the set of locations that remains out of the original anonymity set C after the attacker excludes locations that do not contain the target user. It worth noting that the LPL metric can be used to quantify the impact on the privacy level of any attack that manages to reduce the anonymity set. It is not restricted to the ALE algorithm. In an ideal case, the probability of identifying the real location is $\frac{1}{(size(C))}$. However, if the attacker manages to exclude some locations as dummy and reduces the anonymity set to C^r, then the probability of identifying the real location improves to $\frac{1}{(size(C^r))}$. This means, that as the probability of identifying the real location increases, the LPL metric decreases. Hence, it can be produced by applying any other kind of attackers on the original anonymity set C. Therefore, LPL represents the probability of identifying the real location. In other words, the LPL metric value increases as the attacker's ability to exclude locations from the original anonymity set decreases. Note that the maximum value for the LPL metric occurs when $C^r = C$. This is the ideal case when the attacker completely fails to exclude any location from the anonymity set C.

Let consider an example to illustrate the calculation of the LPL metric. Assume we have the anonymity set $\boldsymbol{C}$ as in table 3, which consists of six locations and $\boldsymbol{s = 5}$. The first location $\boldsymbol{c_1}$ will be excluded based on the first condition of the ALE algorithm because there are no users inside $\boldsymbol{c_1}$. According to the fourth condition of the ALE algorithm, $\boldsymbol{c_3}$ will be excluded because of ($\boldsymbol{6 - 5 = 1}$) < ($\boldsymbol{3 - 1 = 2}$). That is by subtracting the number of queries of the target user from the total queries of $\boldsymbol{c_3}$, the remaining queries are below the minimum number of queries that should be generated by other users inside $\boldsymbol{c_3}$. Also, $\boldsymbol{c_5}$ will be excluded because of the total queries of $\boldsymbol{c_5}$, which equals to 3, is smaller than 5 (the third condition of the ALE algorithm). Based on the second condition of the ALE algorithm, $\boldsymbol{c_6}$ will be excluded because there is only one user exists in $\boldsymbol{c_6}$ and the total queries of $\boldsymbol{c_6}$ isn't equal to $\boldsymbol{s}$. Hence, the LPL metric for this example is $\boldsymbol{LPL = ln(6 - 4) = 0.693}$ instead of the maximum possible value of $\boldsymbol{LPL = ln(6 - 0) = 1.792}$ when there are no excluded locations. Therefore, the real location is either in $\boldsymbol{c_2}$ or $\boldsymbol{c_4}$. Furthermore, the location privacy level relies on the value of $\boldsymbol{LPL}$. The high value of $\boldsymbol{LPL}$ leads to a high level of location privacy.

Table 3: Example of LPL metric

Location Number	# of total queries	# of users
c_1	0	0
c_2	7	2
c_3	6	3
c_4	17	5
c_5	3	3
c_6	10	1

4.2. Experiment Results

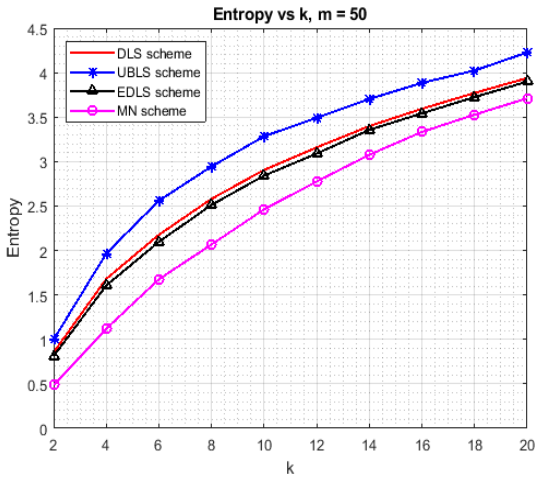
We implemented the three schemes, which are MN, DLS, and EDLS, in addition to the proposed UBLS scheme. The environment used in the implementation is MATLAB. In the simulation, we assume that there is a map of size 20 * 20 cells. Each cell represents a location. We tested the four schemes under six scenarios. In all scenarios, we measured the entropy metric, the product of distances, the total area and the LPL metric for the four schemes. We changed the number of users m in each scenario such that in the first scenario, m is equal to 50 while it is equal to 100 in the second scenario and so on.

The entropy vs k is shown in fig. 9. As can be seen from the figure, UBLS outperforms the other schemes when the number of users is small ($m = 50$ and $m = 100$). Moreover, its performance is comparable to DLS and EDLS when the number of users is large. Moreover, MN has the worst performance in all cases due to the approach of selecting dummy locations randomly.

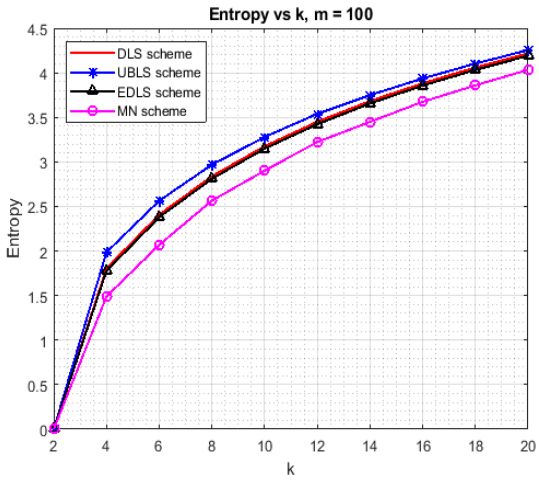
The product of distance vs k is illustrated in fig. 10. According to the results, the value of the product of distances in the EDLS scheme is better than the values of both the DLS and the UBLS schemes. This is because the EDLS selects the dummy locations that are spread far away as possible. On the other hand, the UBLS scheme is approximately close to the DLS scheme but it is better the DLS scheme when $\boldsymbol{m = 50}$, $\boldsymbol{m = 300}$ and $\boldsymbol{m = 1000}$.

The total area vs k is shown in fig. 11. The EDLS covers a bigger area than the DLS and the UBLS schemes as we mentioned in chapter three, the EDLS scheme takes into account the cloaking region. However, the UBLS scheme is better than the DLS in the first scenario to the fourth scenario ($\boldsymbol{m = 50}$ to $\boldsymbol{m = 500}$).

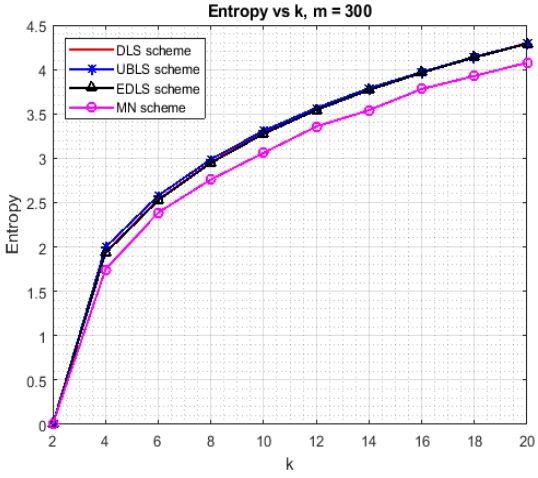
The LPL vs k is clarified in fig. 12. Based on the results in all scenarios, the UBLS scheme is superior to DLS and EDLS schemes and it is close to the optimal case in terms of location privacy. Moreover, the EDLS is better than the DLS scheme in most scenarios.



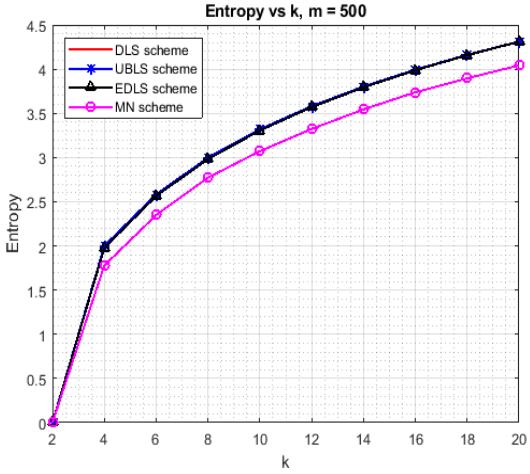
(a) $m = 50$



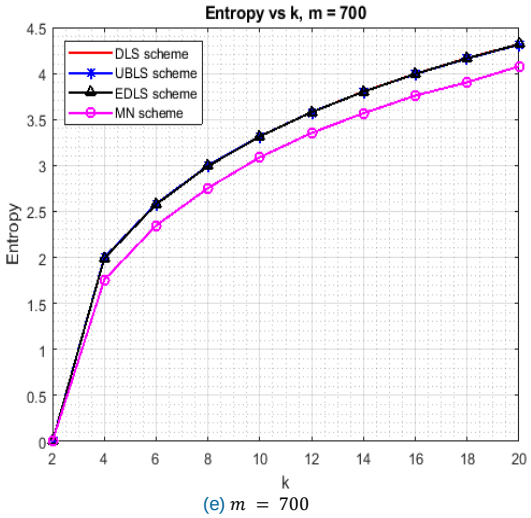
(b) $m = 100$



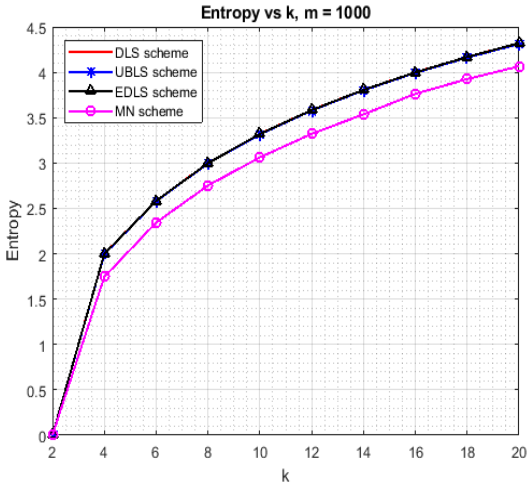
(c) $m = 300$



(d) $\boldsymbol{m = 500}$

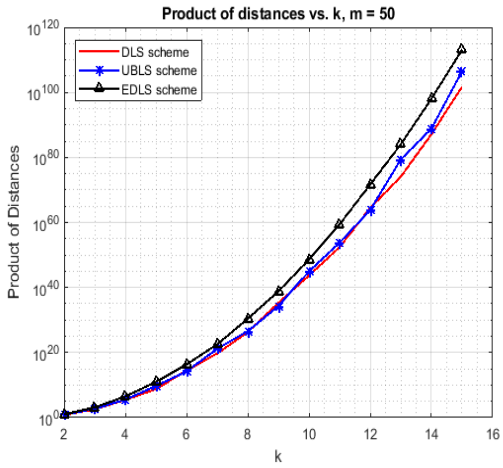


(e) $m = 700$

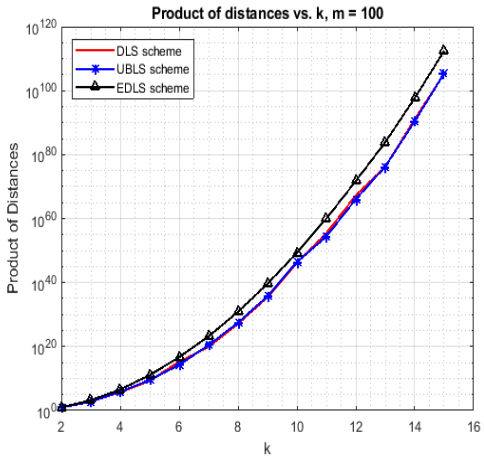


(f) $m = 1000$

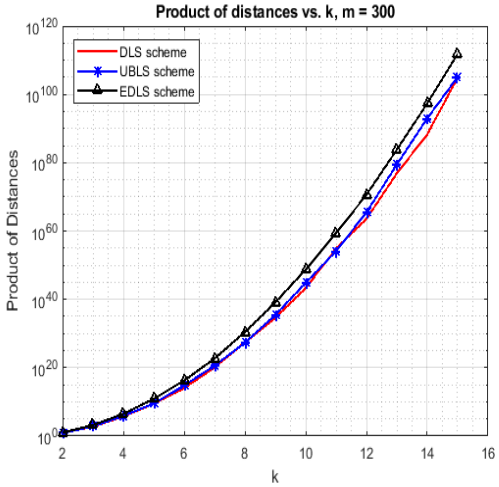
Figure 9: The entropy vs the size of the anonymity size $\boldsymbol{k}$ for different values of $\boldsymbol{m}$



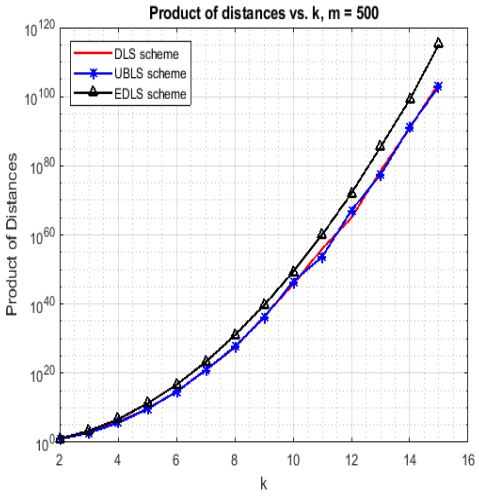
(a) $m = 50$



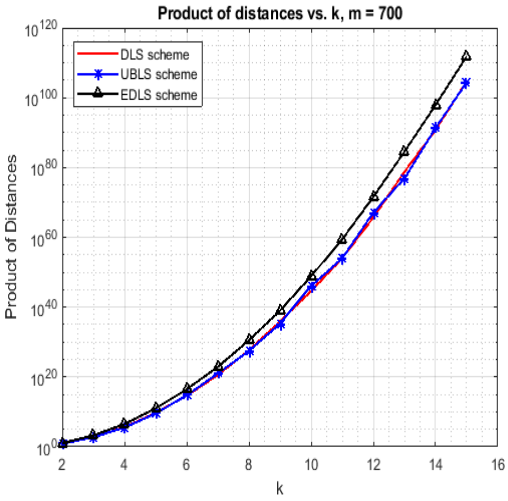
(b) $m = 100$



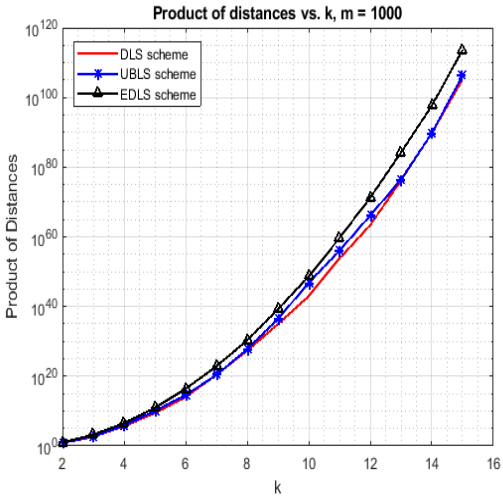
(c) $m = 300$



(d) $m = 500$

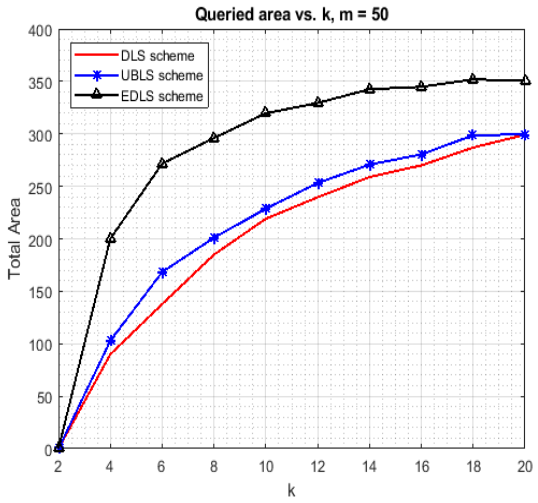


(e) $m = 700$

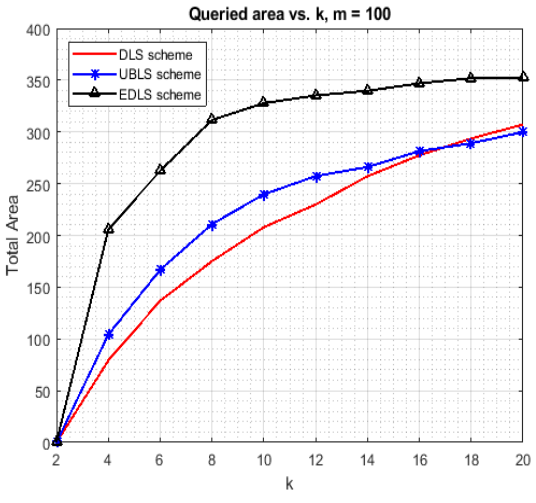


(f) $m = 1000$

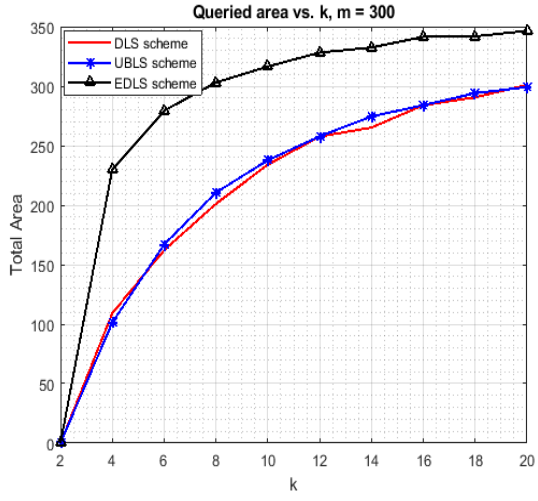
Figure 10: The product of distances vs the size of the anonymity set $\boldsymbol{k}$ for different values of $\boldsymbol{m}$



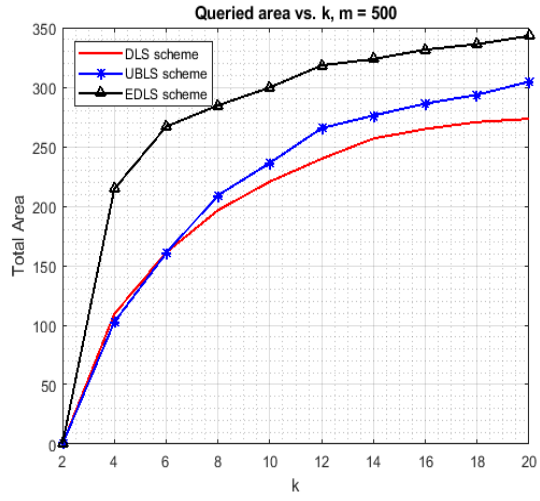
(a) $m = 50$



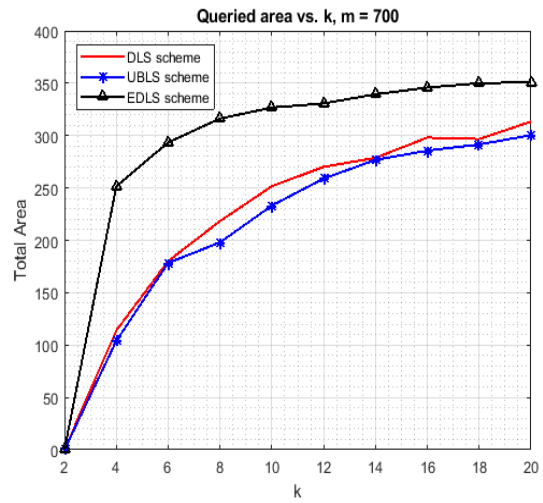
(b) $m = 100$



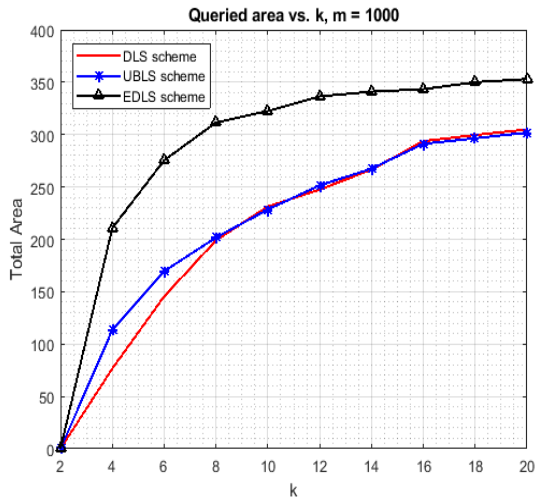
(c) $m = 300$



(d) $m = 500$

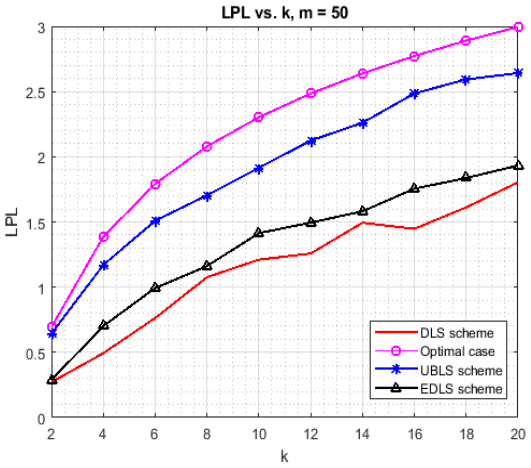


(e) $m = 700$

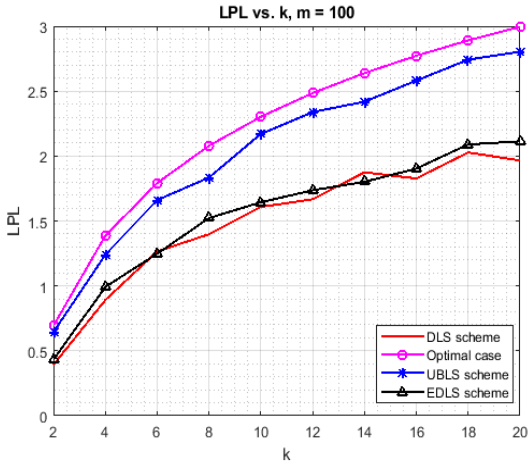


(f) $m = 1000$

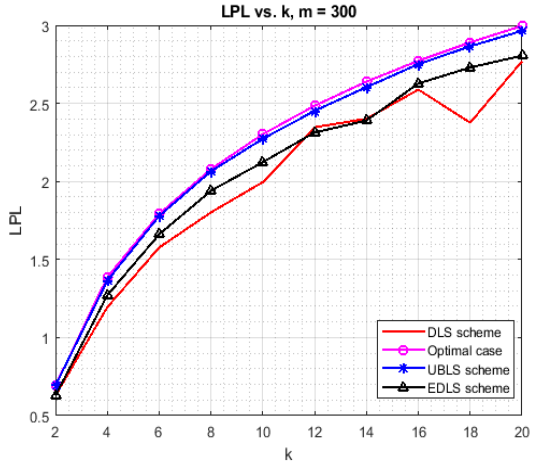
Figure 11: The total area vs the size of the anonymity set $\boldsymbol{k}$ for different values of $\boldsymbol{m}$



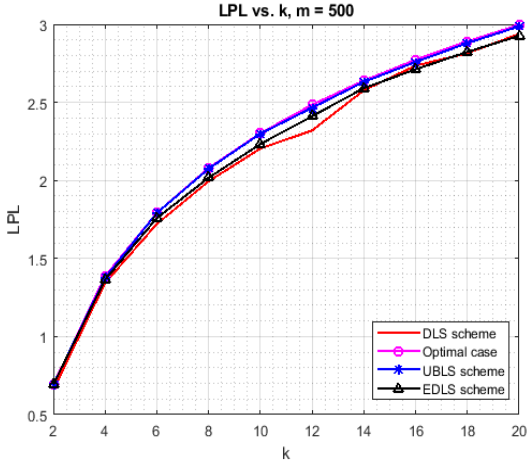
(a) $m = 50$



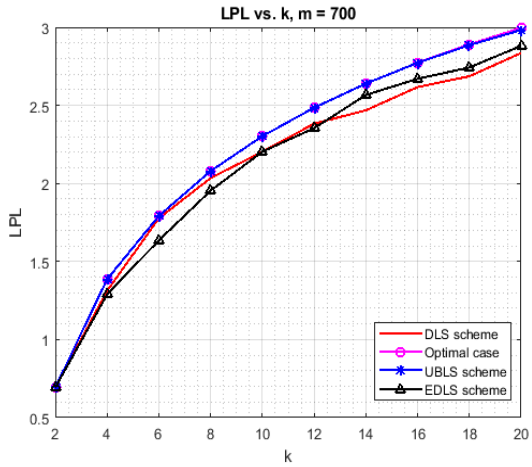
(b) $m = 100$



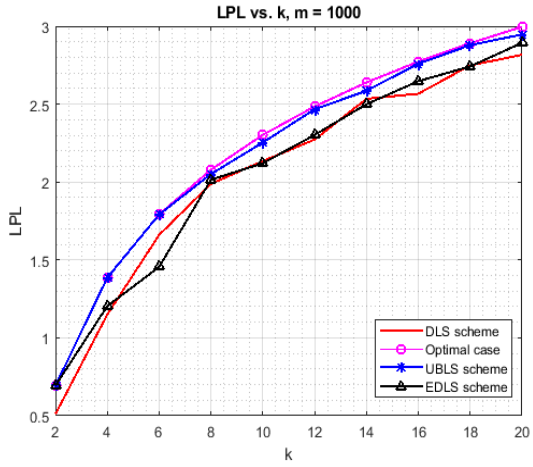
(c) $m = 300$



(d) $m = 500$



(e) $m = 700$



(f) $m = 1000$

Figure 12: The ***LPL*** metric vs the size of the anonymity set $\boldsymbol{k}$ for different values of $\boldsymbol{m}$

4.3. Summary

This chapter explained the performance metrics which are entropy, cloaking region, and LPL metrics. These metrics were used to evaluate UBLS, MN, DLS and EDLS schemes in terms of the anonymity level of the real location of the user. According to the results that were given in this chapter, the UBLS scheme outperformed the other schemes in terms of entropy metric when the number of users was small. Moreover, the UBLS scheme supreme other schemes in terms of the LPL metric.

5. Conclusions and Future work

5.1. Conclusion and Final Remarks

In this thesis, we proposed the UBLS scheme to protect user's location privacy against adversaries in the case that the LBSs server is untrusted. Based on the users' queries and the k-anonymity technique, the UBLS scheme carefully chooses a set of users who have the same users' queries or close to the user's query that exists in the real location. These locations are used by the UBLS scheme as dummy locations. We also proposed a new metric namely the LPL metric. The LPL measures the privacy level of the anonymity set incorporating into calculation the ability of the attacker to exclude some dummy locations from that set. We evaluated the UBLS scheme against other related schemes such as DLS, EDLS and MN schemes. The evaluation results show that the UBLS scheme can significantly improve the privacy level in terms of entropy and the LPL metric. Moreover, we summarize the main important points that are observed during the simulation results of this thesis:

- The higher degree of anonymity represented by symbol k leads to a high level of privacy with more cost in computation.
- The distribution of dummy locations affects both the privacy level and the quality of the service provided by the LBSs server (i.e. when the dummy locations are far away from each other leads to a high level of privacy but a weak in the service offered by the LBSs server).
- The type of dummy locations that are chosen may lead to expose the real location of the user by the LBSs server such as population-free locations (e.g. mountains, valleys, rivers…etc.).

Finally, we give some remarks for the researcher who is interested in the location privacy topic. First, he/she must have a well understanding of the location privacy issue. Second, he/she should search for the existing techniques that handle the location privacy issue. If he/she is interested in a certain technique (e.g. k-anonymity technique) then he/she should look for the recent researches that are using this technique. Subsequently, he/she tries to find a gab that is not covered by these researches and starts his/her research from this gab.

5.2. Future Work

In future work, we will try to develop the UBLS scheme by considering different kinds of attackers. Also, we will enhance the ALE algorithm with respect to the eavesdropper. Moreover, we will propose another location privacy metric after looking at the current researches in this field.

References

[1] W. Martin, F. Sarro, Y. Jia, Y. Zhang, and M. Harman, "A Survey of App Store Analysis for Software Engineering," *IEEE Trans. Softw. Eng.*, vol. 43, no. 9, pp. 817–847, 2017.

[2] T. Kalanick and G. Camp, "Uber - Earn Money by Driving or Get a Ride Now," *Uber.com*, 2018. https://www.uber.com/ (accessed Dec. 15, 2018).

[3] M. Zuckerberg, E. Saverin, A. McCollum, D. Moskovitz, and C. Hughes, "Facebook," *Facebook*, 2018. https://www.facebook.com/ (accessed Dec. 15, 2018).

[4] J. Dorsey, N. Glass, E. Williams, and B. Stone, "Twitter," *Twitter.com*, 2018. https://twitter.com/ (accessed Dec. 15, 2018).

[5] Y. Lin, C. Lai, J. William Chapman, S. Felix Wu, and G. Barnett, "Geo-Location Identification of Facebook Pages," in *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, Barcelona, Spain, 2018, pp. 441–446.

[6] G. Abalı, E. Karaarslan, A. Hürriyetoğlu, and F. Dalkılıç, "Detecting citizen problems and their locations using twitter data," in *6th International Istanbul Smart Grids and Cities Congress and Fair (ICSG)*, Istanbul, Turkey, 2018, pp. 30–33.

[7] B. Niu, Q. Li, X. Zhu, G. Cao, and H. Li, "Achieving k-anonymity in Privacy-Aware Location-Based Services," presented at the IEEE Conference on Computer Communications, 2014, pp. 754–762.

[8] H. Kido, Y. Yanagisawa, and T. Satoh, "An anonymous communication technique using dummies for location-based services," in *Proceedings of International Conference on Pervasive Services*, 2005.

[9] J. Krumm, "A survey of computational location privacy," *Pers. Ubiquitous Comput.*, vol. 13, no. 6, pp. 391–399, 2008.

[10] M. Duckham and L. Kulik, "Location privacy and location-aware computing," *Drummond J Ed Dyn. Mob. GIS Investig. Change Space Time*, pp. 34–51, 2006.

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

[12] E. Kaasinen, "User needs for location-aware mobile services," *Pers. Ubiquitous Comput.*, vol. 7, no. 1, pp. 70–79, 2003.

[13] L. Barkuus and A. Dey, "A Location-based services for mobile telephony: a study of users' privacy concerns," presented at the 9TH IFIP TC13 International Conference on Human-Computer Interaction, Switzerland, Jul. 2003.

[14] K. Rajendran, M. Jayabalan, and M. Rana, "A Study on k-anonymity, l-diversity, and t-closeness Techniques," *IJCSNS*, vol. 17, no. 12, pp. 172–177, 2017.

[15] P. Belsis and G. Pantziou, "A k-anonymity privacy-preserving approach in wireless medical monitoring environments"," *Pers. Ubiquitous Comput.*, vol. 18, no. 1, pp. 61–74, 2012.

[16] A. Machanavajjhala, J. Gehrke, D. Kiefer, and M. Venkitasubramanian, "L-diversity: privacy beyond k-anonymity," in *IEEE ICDE*, 2006.

[17] N. Li, T. Li, and S. Venkatasubramanian, "Closeness: A New Privacy Measure for Data Publishing," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 7, pp. 943–956, 2010.

[18] N. Li, T. Li, and S. Venkatasubramanian, “t-Closeness: Privacy Beyond k-Anonymity and l-Diversity,” presented at the in 23rd International Conference on Data Engineering, Istanbul, Turkey, 2007, pp. 106–115.
[19] V. Iyengar, “Transforming data to satisfy privacy constraints,” in *In Proc. 8th ACM KDD*, 2002, pp. 279–288.
[20] J. Domingo-Ferrer and V. Torra, “A Critique of k-Anonymity and Some of Its Enhancements,” presented at the in 3rd International Conference on Availability, Reliability and Security (ARES ’08), Barcelona, Spain, 2008, pp. 290–293.
[21] J. Maier and B. Ralph Holz, “Anonymity: Formalisation of Privacy – k-anonymity,” *Netw. Archit. Serv.*, pp. 41–48, 2013.
[22] R. Di Pietro, F. Lombardi, and L. Jero, “A GPU Algorithms for K-anonymity in Microdata,” presented at the In: IEEE Conference on Communications and Network Security (CNS). Washington DC: IEEE.
[23] D. Yang, X. Fang, and G. Xue, “Truthful incentive mechanisms for k-anonymity location privacy,” in *2013 Proceedings IEEE INFOCOM*, Italy, 2013, pp. 2994–3002.
[24] B. Niu, Q. Li, X. Zhu, and H. Li, “A Fine-Grained Spatial Cloaking Scheme for Privacy-Aware Users in Location-Based Services,” presented at the 23rd International Conference on Computer Communication and Networks (ICCCN), Shanghai, China, 2014.
[25] J. Zeng *et al.*, “Mobile r-gather: Distributed and Geographic Clustering for Location Anonymity,” in *Proceedings of the 18th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, India, 2017.
[26] J. Wang, Y. Li, D. Yang, H. Gao, G. Luo, and J. Li, “Achieving Effective k-Anonymity for Query Privacy in Location-Based Services,” *IEEE Access*, vol. 5, pp. 24580–24592, 2017.
[27] D. Zhao, Y. Jin, K. Zhang, X. Wang, P. Hung, and W. Ji, “EPLA: efficient personal location anonymity,” *GeoInformatica*, vol. 22, no. 1, pp. 29–47, 2017.
[28] L. ZHENG, H. YUE, Z. LI, X. PAN, M. WU, and F. YANG, “k-Anonymity Location Privacy Algorithm Based on Clustering,” *IEEE*, vol. 6, pp. 28328–28338, 2018.
[29] F. Fei, S. Li, H. Dai, C. Hu, W. Dou, and Q. Ni, “A K-Anonymity Based Schema for Location Privacy Preservation,” *IEEE Trans. Sustain. Comput.*, vol. 4, no. 2, pp. 156–167, 2019.
[30] J. Manweiler, R. Scudellari, and L. Cox, “Smile: Encounter-based trust for mobile social services,” in *ACM CCS*, 2009.
[31] M. Gruteser and D. Grunwald, “Anonymous usage of location-based services through spatial and temporal cloaking,” 2003.
[32] M. Duckham and L. Kulik, “Location privacy and location-aware computing,” in *Dynamic & mobile GIS: investigating change in space and time*, Boca Raton, 2006, pp. 34–51.