Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/MBE

MBE, 17(3): 2636–2649.
DOI: 10.3934/mbe.2020144
Received: 20 April 2019
Accepted: 03 September 2019
Published: 06 March 2020

Research Article

A stochastic location privacy protection scheme for edge computing

Yuan Tian[1], Biao Song[2,]*, Mznah Al Rodhaan[3], Chen Rong Huang[1], Mohammed A. Al-Dhelaan[3], Abdullah Al-Dhelaan[3] and Najla Al-Nabhan[3]

[1] School of Computer Engineering, Nanjing Institute of Technology, Nanjing 211167, China
[2] School of Computer Software, Nanjing University of Information Science and Technology, Nanjing 210044, China
[3] Department of Computer Science, King Saud University, 11362, Saudi Arabia

* **Correspondence:** Email: ytian@njit.edu.cn.

Abstract: Location-based Service has become the fastest growing activity related service that people use in their daily life due to the boom of location-aware mobile devices. In edge computing along with the benefits brought by LBS, privacy preservation becomes a more challenging issue because of the nature of the paradigm, in which peers may cooperate with each other to collect and analyze user's location data. To avoid potential information leakage and usage, user's exact location should not be exposed to the edge node. In this paper, we propose a stochastic location privacy protection scheme for edge computing, in which the geographical distribution of surrounding users is obtained by analyzing proposed long-term density map and short-term density map. The cloaking scheme transfers user's exact location to a cloaked location to satisfy predefined probability of having *k*-users in that area. Our scheme does not reveal any exact location information, thus it is practicable for the real scenario when edge computing is honest but curious. Extensive experimental results are conducted to verify the efficiency and effectiveness of our method. By varying the privacy protection requirements, the corresponding performance have been examined and discussed.

Keywords: location-based service; edge computing; location privacy

1. Introduction

Due to the development and rapid expansions of the use of big data, IoT and 5G networks, massive data is generated by the edge equipment of the network. New challenges arise with respect to highly responsive cloud services delivery for mobile computing, scalability and privacy-policy enforcement, and the ability to mask transient cloud outages [1].

Edge computing is a novel promising computing model that breaks the frontier of computing applications, data, and services away from the centralized nodes/Cloud to multitude of end-user or near-user edge terminal devices coordinates by the edge nodes [2]. It enables new applications and services such as VR head tracked systems requiring less than 16ms to achieve perceptual stability [1]. The data storing, analyzing, and processing all occur near the source of the data.

The natural features and benefits of edge computing like heterogeneous distributed interactive architecture and massive data processing make the Cloud-to-User security and privacy protection methodologies no longer suitable/ tailor for edge computing. By serving as the first point of contact, edge nodes should help the users to enforce the privacy policies prior to release of the data to the cloud [1]. Besides, as some of the end devices are resources constrained, complex security algorithms cannot be executed or a large amount of data cannot be stored there [3].

Among all of the pervasive mobile and cloud-based services, location-based service is one of the most suitable services for the decentralized deployment of edge computing scenario. The burgeoning edge computing provides great opportunity to enhance the traditional location-based service because of its distinct characteristics like low latency, proximity and location awareness [4].

Since location-based service is provided inside of the sub-area of edge computing community independently, data is collected and processed without uploading to the cloud. Although deploying location-based service on edge node could avoid the location privacy threat from the central cloud provider, the concern on the location privacy still exists in the location-based service for the edge computing scenario. Location privacy threat arises during the procedure of the fingerprint localization, and the previous studies on location privacy are ineffective because of different threat models and information semantic.

Based on the above observation, in this paper, we propose a location privacy preservation scheme for edge computing. We respect the fact that users may use one of the existing location cloaking techniques. Our solution is designed to be compatible with existing techniques by forming the users' location distribution retrieved from the heterogeneous location information of all users. Based on the location distribution, we provide location cloaking techniques for the unprotected user by satisfying the predefined probability of having k-users in the cloaked area. We focus on the honest-but-curious (HBC, also known as semi-honest) adversary. According to [5], the HBC adversary is a legitimate participant in a communication protocol who will not deviate from the defined protocol but will attempt to learn all possible information from legitimately received messages. In our scenario, HBC edge node will be honest in performing the location based services but will be interested in learning all possible locations. Our scheme could provide privacy preservation on edge node and does not require any participates from the trusted third party. Meanwhile, our solution also prevent the edge node from knowing exact user location.

The rest of this paper is organized as follows. Section 2 briefly reviews the related work. Section 3 describes the proposed SLPP architecture with its assumptions. We describe the design details of our proposed stochastic scheme in Section 4. Section 5 provides theoretical analyses and experimental evaluations. Finally, conclusions are drawn in section 6.

2. Related work

Plenty of researches focus on how to design privacy protection models or algorithms for location-based service in fog or edge computing scenarios. The works are generally divided in three categories, which are location anonymization, obfuscation method and cryptography techniques.

The most well-known anonymization method is k-anonymity, which is to combine users exact location with the rest k-1 users' locations and send all k users' location back to the server. The authors in [6] introduced an anonymous method by adding fake dummy locations to protect location privacy. Their method requires user's exact locations, which may result in the location privacy disclosure. Ma et al. [7] achieved a k degree anonymity by modifying the vertex and edge in social networks. In [6–8], Grunwald et al. proposed an adaptive k-anonymity algorithm to meet the user's anonymity constraints, by adjusting the resolution of location information based on the time and space dimensions.

The popularity of applying k-anonymity scheme for location privacy is mainly because of simplicity [9]. A large number of works discuss about how to increase the k-anonymity efficiency and reduce the query obfuscation cost [10,11–13], by extending the obfuscation method to protect trajectory privacy [14], or adapting Gruteser et al.'s architecture to other domains such as VANET [15]. The idea of obfuscation is to replace user's actual location by an imprecise location. A cloaking algorithm was proposed by Ghinita et al., in [16], Hilbert space filling curve ordering was used to sort and group user's location. In [17], Xu et al. designed an algorithm based on user's proximity information without knowing their real locations. Whereas they did not consider the case when user sends the area continuously, the overlapping regions may result in privacy leakage. Ma et al. [18] solved the above issue by proposing an algorithm to avoid overlapping circle attack by changing the confidence level to a fixed value. Yuan T. et al. [19] also solved the overlapping problem by proposing a location privacy-protection method to preserver user's trajectory privacy. They apply mobility Markov chain to analyze user's moving behaviors. The proposed cloaking algorithm enlarges small area in order to meet user's privacy preferences.

Though cryptograph can be used for protecting location privacy by not sending location to LBS, the current index methods cannot be used. In [20], T. Wang et al. proposed a trajectory privacy preserving scheme on a Fog structure for cloud-based location service. They applied public key cryptography to encrypt the data to ensure data security. The encrypted data is divided into separate parts, in which partial data is stored on the fog server for the physical protection, while the other part will be uploaded to the LBS server. Although their method prevent the attacks from restoring the raw data, data utility and quality of service are reduced as incomplete information provided to LBS.

There are other privacy preserving methods which are used to protect user's privacy while publishing their data, i.e. differential privacy [21,22] and t-closeness [23]. The user's data is stored in a database, apply the above mentioned methods before sharing. After the data is published, it is then analyzed for different applications.

3. System architecture and assumption

In this section, we present and describe the system architecture of the proposed SLPP scheme for edge computing first. Then we discuss the assumption and compare it with the existing works.

3.1. System architecture

Figure 1 shows the system architecture of the proposed SLPP scheme. We define four roles in our system including mobile end users, location-based service providers (LSPs), third party location protectors and edge node.

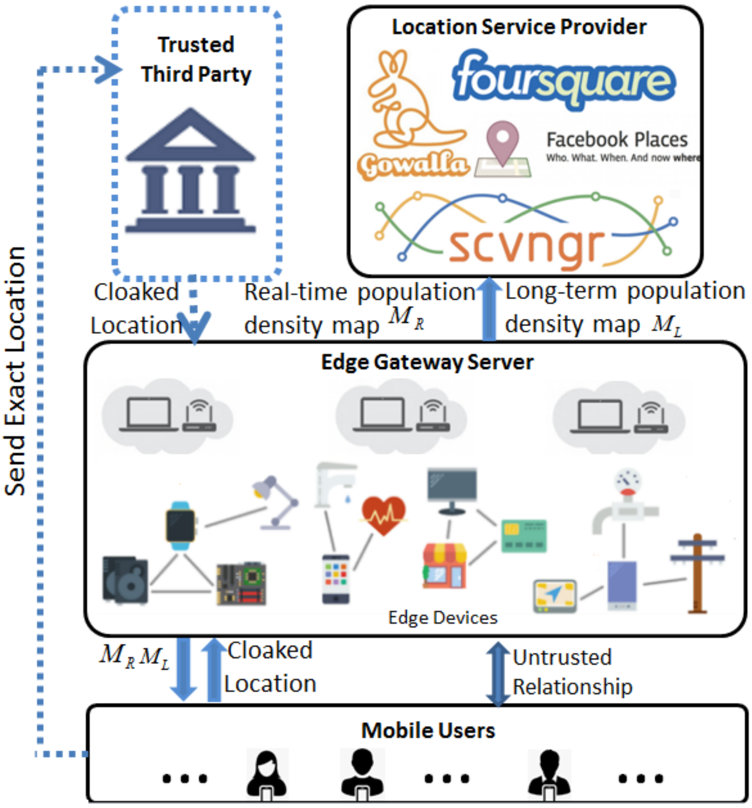

Figure 1. The proposed SLPP architecture.

Mobile End Users: Mobile end users along with their mobile devices continuously produce exact location information through GPS. They can provide their exact location to a LSP if they believe the LSP is trusted, or apply any location protection approach (on their device or through third party location protector), and then send the cloaked location information to LSP. In the scenario of edge computing, the protected/unprotected personal location information is stored and processed on the edge node since LSPs deploy their services on those nodes.

LSPs: LSPs are online location-based service provides, e.g., Wechat or Twitter. LSPs stores map information and many different types of points and interests (POI) such as restaurants, hospital and gas station. Given user location information (exact or cloaked) and type of POI as a query, LSPs use their application logic to find answers from their database. In edge computing scenario, LSPs can also request real-time population density information from edge node to help them to improve their services.

Third Party Location Protector: Third party location protector is often trusted by the registered users. They receive users' exact location information and apply some particular types of algorithms to cloak user location information before sending to LSPs. Having trusted third party is posting an attractive target for attacker since all exact location information is stored on it. In our proposed approach, mobile users do not need to use third party location protector. However, we still consider them since we respect the fact that third party location protectors exist in market, and our proposed scheme will be compatible with them.

Edge Nodes: Edge nodes refer to the computing infrastructure/resource that exists close to the sources of data. LSPs deploy services on edge node to improve the quality of their services, e.g., service response time. In our proposed scheme, edge nodes are responsible for providing long-term and real-time population density information to LSPs and mobile end users.

3.2. Assumption

3.2.1. (K, P_K, D_{MAX})-User Privacy Profile:

(K, P_K, D_{MAX})**-User Privacy Profile**: Mobile end users can define their privacy requirement by specifying three parameters. Users are allowed to arbitrarily change the settings of their privacy profiles at any time.

i) *Anonymization degree,* K: This parameter indicates the anonymity level requirement. Protection algorithm should calculate a cloaked area containing at least K different users.

ii) *Probability threshold,* P_K: This parameter indicates the stochastic privacy requirement. The probability of having at least K different users in the cloaked area should not be less than P_K.

iii) *Maximum distance,* D_{MAX}: It specifies the maximum tolerable distance from the current location to the farthest boundary of cloaked area.

3.2.2. Information Sharing:

Information Sharing: Information is shared among different roles.

i) *Real-time population density map,* M_R: This information is produced by edge nodes and shared with users/LSPs. It only contains the cloaked user location information that is permitted to share by user and gets updated by edge node in real-time manner. User identification is not allowed to be shared along with M_R. Edge nodes are not trusted in our scheme, but we assume that they are honest when sharing M_R.

ii) *Long-term population density map,* M_L: This information is produced by edge nodes and shared with users/LSPs. It contains user appearance frequency information counted by edge nodes over a time period L. Edge nodes are not trusted in our scheme, but we assume that they are honest when sharing M_L.

iii) *Cloaked individual user location information,* A_u: This information is produced by individual user and shared with edge nodes/LSPs. Individual user u takes M_R, M_L, (K, P_K, D_{MAX}) as input and adopt proposed SLPP algorithm to generate A_u. Alternatively, some users may rely on trusted third party location protector to generate this information. We do not restrict the shape of A_u to circle, rectangle, or any other specific shape, which means our scheme is compatible with most of the existing techniques. All users are assumed to be legal users.

In our scheme, we do not require any protection over the communication channels since none of the sensitive user location information will be shared during the entire process. Such security requirement is often observed when using trusted third party for location protection as sensitive data must be transmitted from user to third party. It significantly improves the flexibility of privacy protection and makes it adoptable in untrusted network environment.

4. Stochastic location privacy protection scheme

In this section, we explain three major algorithms in our scheme. The edge node holds a cache in which to store the location information received from the users or the third party location protectors. The data are recorded in the format $\{A_u^T, T\}$, where A_u^T is the cloaked location information from any user u generated at time T. The edge node applies density map generating algorithm to produce long-term population density map M_L by using all records over a time interval L and to produce real-time population density map M_R by using the records with $T = latest\_time\_stamp$ only. Then M_R and M_L is transmitted to any mobile user u who needs to cloak his/her location. Given user current location at (h,v), M_R, M_L and user privacy profile (K, P_K, D_{MAX}), we propose a stochastic evaluation algorithm to judge whether an arbitrary cloaked location A' satisfies user privacy protection requirement or not. Finally, a location cloaking algorithm is designed to calculate the cloaked individual user location information A_u, which will be submitted to edge node as a part of LBS query.

4.1. Density map generating algorithm

The long-term population density map M_L is defined as a $m \times n$ matrix presenting the population density information in a $m \times n$ area. The individual items in M_L is denoted by m_{ij} showing the density information at location (i, j), where max $i = m$ and max $j = n$. Given current time stamp C, the input data of long-term population density map contains all records of $\{A_u^T, T\}, \forall u, T \geq C - L$.

For each time slot S, the algorithm first uses all cloaked locations $\{A_u^S\}$ to generate a $m \times n$ matrix M_L^S. We assume that the exact user location in a cloaked area A_u^S is completely random, in another word, the probability of having an individual user u is same for all locations $(i, j) \mid (i, j) \in A_u^S$. Thus, the elements of M_L^S is defined as a summation of likelihood values shown in the following formula:

$$m_{ij}^S = \sum_u \frac{1}{SIZE(A_u^S)} \mid (i, j) \in A_u^S \tag{1}$$

where $SIZE(A_u^S)$ is the size of cloaked area A_u^S.

Then we combine the set of $\{M_L^S\}$, $\forall C-L \le S \le C$ in a weighted cumulative manner to calculate M_L as follows:

$$M_L = \sum_{C-L\le S\le C} w^S \times M_L^S \tag{2}$$

where the weight factor w^S is adopted as an exponential smoothing factor. It guarantees that the weight values of past observations are exponentially decreased over time. For example, if we use a degradation parameter α between 0 and 1, we can have $w^C = 1$, $w^{C-1} = \alpha, \ldots,$ $w^S = \alpha^{C-S}$, etc. The small the α is, the faster the weight value decreases.

It can be shown that in Figures 2 and 3, the cloaked location information and the long-term population density map. In Figure 2, a set of cloaked locations shaped round or square are randomly generated and presented. The lighter the color is, the older the time stamp is. In Figure 3, the final long-term population density map M_L for the same area.

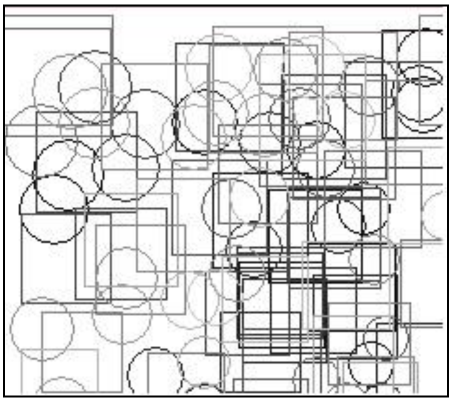

Figure 2. A set of cloaked location information generated over time.

The real-time population density map M_R is defined as a set of cloaked area with current stamp C, i.e., $M_R = \{A_u^C\}$. Edge node just collects the information, and no further calculation is needed.

4.2. Stochastic evaluation algorithm

Our proposed stochastic evaluation algorithm first takes a cloaked location A' and tries to find a set of cloaked locations from M_R that have non-empty intersection with A', i.e., finding $M_R' = \{A_u^C\}, \forall A_u^C \cap A' \ne \varnothing$. For every cloaked location in, M_R' the probability of having one user at $A_u^C \cap A'$ can be calculated by using the following formula:

$$P(u \text{ at } A_u^C \cap A') = \frac{\sum_{i,j} m_{ij}}{\sum_{h,v} m_{hv}} \,|\, (i,j) \in A_u^C \cap A', (h,v) \in A_u^C \tag{3}$$

If $A_u^C \cap A' = A_u^C$, then we can get $P(u \text{ at } A_u^C \cap A') = 100\%$. It can be demonstrated in Figure 4 that the probability calculation method using the values of the long-term population density map.

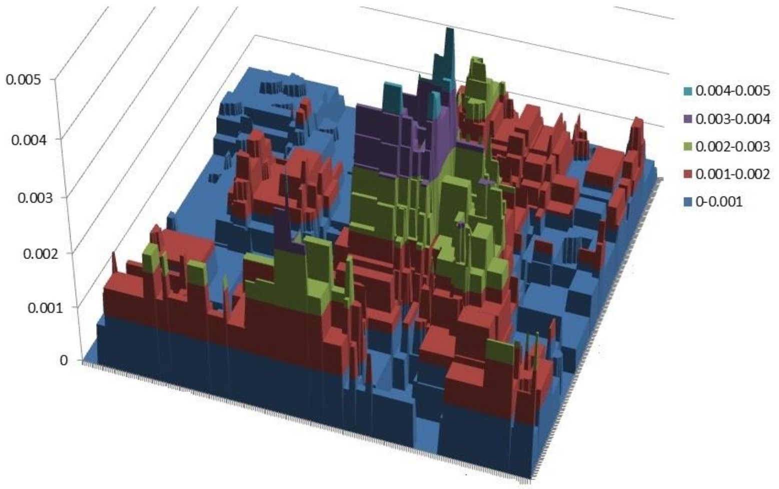

Figure 3. Long-term population density map of the same area as shown in Figure 2.

Suppose that the size of M'_R is n, we use $\{P_1, P_2, \ldots, P_n\}$ to denote the probability values derived from Eq (3). A set of Bernoulli random variables $\{b_1, b_2, \ldots, b_n\}$ is defined where $b_i = 1$ means user i is at $A_u^C \cap A'$ and $b_i = 0$ means user i is at $A_u^C - (A_u^C \cap A')$. Thus, we can get $P(b_i = 1) = P_i$ and $P(b_i = 0) = 1 - P_i$. It is reasonable to assume that $\{b_1, b_2, \ldots, b_n\}$ are independent and identically distributed since they refer to different individual users.

Given $\{b_1, b_2, \ldots, b_n\}$, the number of users in A' can be calculated as $\sum_{i=1}^{n} b_i$, and the probability can be calculated as $\prod_{i=1}^{n} (P_i)^{b_i} (1 - P_i)^{1-b_i}$. Finally, we are able to evaluate the probability of having K users at A' with the following formula:

$$P(K \text{ users at } A') = \text{ sum of } \prod_{i=1}^{n} (P_i)^{b_i} (1 - P_i)^{1-b_i}, \forall \sum_{i=1}^{n} b_i = K \tag{4}$$

Equation (4) is a special case of the multinomial distribution, and is also a special case of the categorical distribution as each trial only two possible outcomes, but the success probabilities of trials are different.

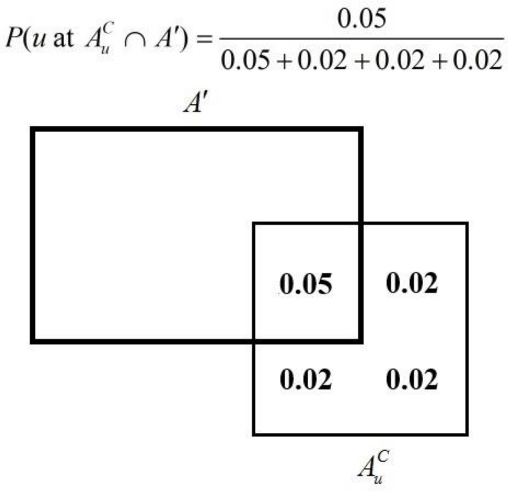
$P(u \text{ at } A_u^C \cap A') = \frac{0.05}{0.05+0.02+0.02+0.02}$
A'
0.05
0.02
0.02
0.02
A_u^C

Figure 4. Calculating the probability of having one user in $A_u^C \cap A'$.

At the end, the stochastic evaluation algorithm will return a Boolean value denoting whether the probability of having more than K users in A' satisfies privacy protection requirement P_K or not. The rule is presented in the following formula:

$$\begin{cases} 1-\sum_{i=1}^{K} P(i \text{ users at } A') \geq P_K & \Rightarrow \quad true \\ \quad otherwise & \Rightarrow \quad false \end{cases} \tag{5}$$

5. Simulations

In this section, the simulations are conducted to verify the efficiency and effectiveness of our proposed SLPP algorithm. We first present the simulation settings, followed by the results and discussions.

5.1. Simulation settings

We conducted all of our simulations with the Java SE Development Kit and Eclipse programming environment, running on a PC with Intel Core-i7 2.4 GHz, 16 GB RAM and Microsoft Windows 7, 64-bit operating system. We used multivariate normal distributions to generate 5000 mobile users' locations with random μ and σ values in an area of 1000 × 1000. The time stamps of users' locations were evenly distributed over 100 continuous time slots. Various parameter have been used in our simulations including $5 \leq K \leq 10$, $50\% \leq P_K \leq 95\%$ and $\alpha = 95\%$.

5.2. Size of cloaked location

We computed the average size of cloaked location with K and P_K value varied. In LBS scenario, the size of cloaked location is often considered as the major indicator of performance as it greatly affects the quality of LBS. A large cloaked location can protect user privacy better, but also makes the contents of LBS less accurate. Thus, most of the privacy protection scheme including our one tried to minimize the size of cloaked location while satisfying user privacy requirement.

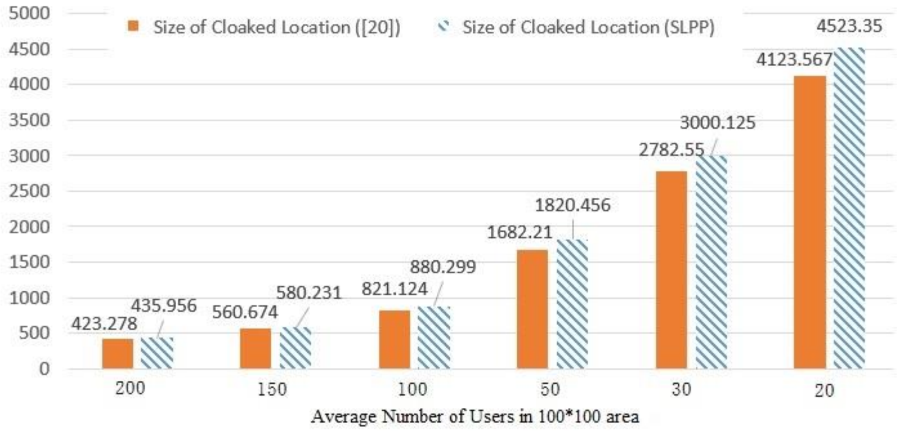

Figure 5. Average number of users vs. size of cloaked location.

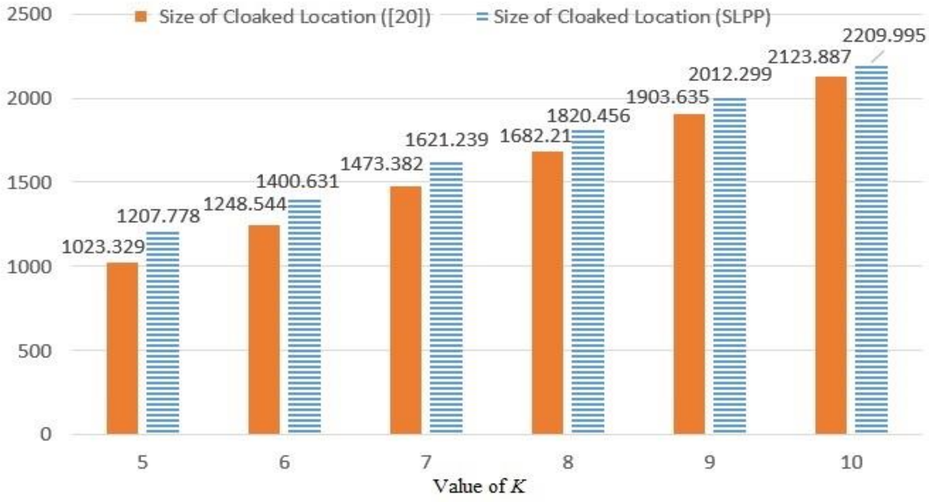

Figure 6. Value of K versus size of cloaked location.

5.2.1. Average number of users vs. size of cloaked location

We first examined performance of SLPP in the areas with different population density. A relatively strict privacy requirement has been adopted in this simulation, i.e., $K = 8$ and $P_K = 90\%$.

The results are shown in Figure 5. We compare our results with the method proposed in [24], which can always find the optimal cloaked area by using exact location information. When the cloaking has been done in an area with high population density, SLPP can provide near-optimal solution without using exact location. In many of the individual records, the size of cloaked location from SLPP is even smaller since only 90% of having at least K users is required. With the population density decreased, we can observe that the performance of SLPP drops gradually. However, since the size of cloaked area increases dramatically, the difference between SLPP and optimal solution is not significant.

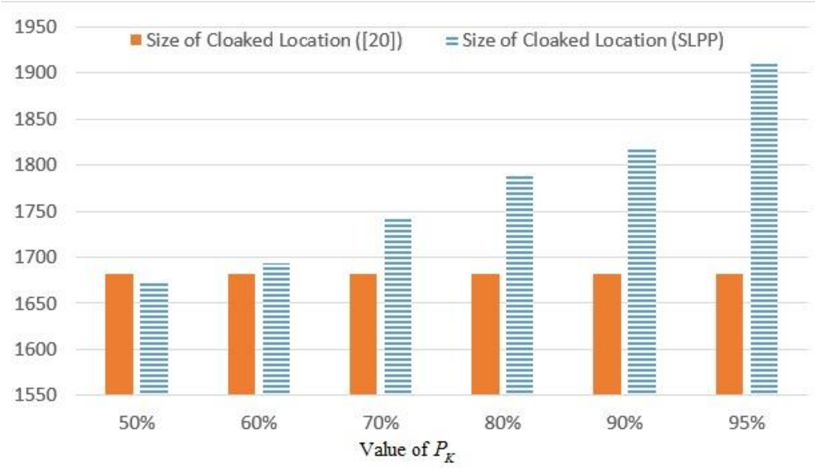

Figure 7. Value of P_K versus size of cloaked location.

5.2.2. Value of K versus size of cloaked location

In this simulation, we varied value of K with $P_K = 90\%$ in a 100 × 100 area and average 50 users. The results shown in Figure 6 tell that the results of SLPP approach to the optimal solution when the value K is increased. The reason is that the users are not evenly distributed in the whole area. With larger K value, the size of cloaked location definitely grows up, which results in a better opportunity to cover an area with higher population density. From previous results we know that SLPP can perform well in such area.

5.2.3. Value of P_K versus size of cloaked location

In this simulation, we varied value of P_K with $K = 8$ in a 100 × 100 area and average 50 users. As we can see from Figure 7, the results of SLPP can reach optimal level if P_K is 50%. The reason is that covering half of a cloaked location can provide approximately 50% of chance to include one user. With larger P_K value, the size of cloaked location in SLPP must grow up as the cost of using stochastic input instead of exact location information.

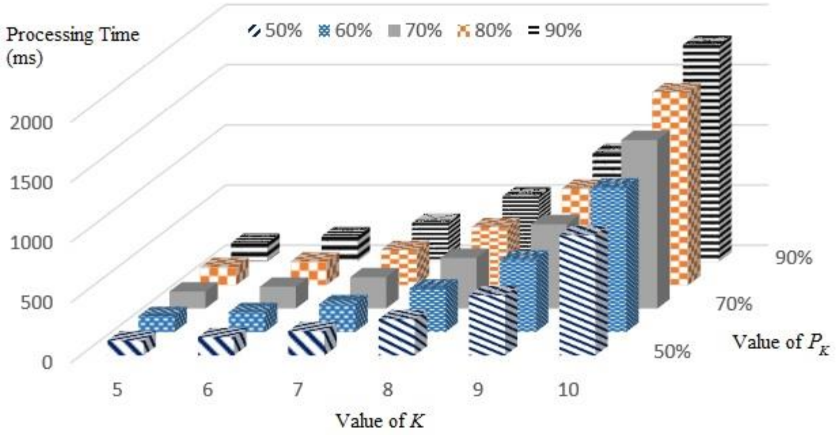

Figure 8. Value of K and value of P_K versus processing time.

5.3. Overhead of SLPP

We examined the overhead of our proposed SLPP from the aspect of processing time. Since SLPP requires mobile user to cloak his/her own location, the processing time can have significant impact on both service response time and energy efficiency of mobile device. It is not hard to identify the stochastic evaluation algorithm is the most computational intensive task in entire solution since it has to calculate a lot of probability values. Thus, we varied the values of K and P_K in this simulation to see the changes in processing time.

As we can observe from Figure 8, the impact of K 's value is significant. The processing time increases dramatically due to the fact that the complexity of Eqs (4) and (5) is related with value of K in an exponential way. On the other hand, the value of P_K is linearly related with the processing time. As we request a higher probability of having same number of user, the algorithm only need to constantly increase the size of area to satisfy the required P_K .

6. Conclusions and future work

In this paper, we proposed a stochastic scheme for edge computing to protect user's location privacy. The main idea of our scheme is to cloak the location information for unprotected user that has to satisfy predefined probability of having *k*-users in the cloaked area. Our scheme does not require any participates from the trusted third part, and it is practicable for the real scenario when edge computing is honest and curious. In the future, we will extend our proposed stochastic scheme to preserve tractor privacy for moving user's with continuous queries.

Acknowledgments

The authors extend their appreciation to the Deanship of Scientific Research at King Saud University for funding this work through research group no. RGP-264.

Conflict of Interest

The authors declared that they have no conflicts of interest to this work. We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted

References

1. M. Satyanarayanan, The emergence of edge computing, *Computer*, **50** (2017), 30–39.
2 L. Xiong, Y. Shi, On the Privacy-Preserving Outsourcing Scheme of Reversible Data Hiding over Encrypted Image Data in Cloud Computing, *CMC Comput. Mater. Con.*, **55** (2018), 523–539.
3. A. Alabdulkarim, M. Al-Rodhaan, Y. Tian A. Al-Dhelaan, A Privacy-Preserving Algorithm for Clinical Decision-Support Systems Using Random Forest, *CMC Comput. Mater. Con.*, **58** (2019), 585–601.
4. X. Du, H. H. Chen, Security in Wireless Sensor Networks, *IEEE Wireless Commun.*, **15** (2008), 60–66.
5. A. J. Paverd, A. Martin, I. Brown, Modelling and automatically analysing privacy properties for honest-but-curious adversaries, *Univ. Oxford Tech. Rep.*, **2014** (2014).
6. H. Kido, Y. Yanagisawa, T. Satoh, *An anonymous communication technique using dummies for location-based services*, ICPS '05, International Conference on Pervasive Services, 2005, 88–97. Available from: https://ieeexplore_ieee.xilesou.top/abstract/document/1506394.
7. T. Ma, Y. Zhang, J. Cao, J. Shen, M. Tang, Y. Tian, KDVEM: A k-degree anonymity with Vertex and Edge Modification algorithm, *Computing*, **97** (2015): 1165–1184.
8. M. Gruteser, D. Grunwald, *Anonymous usage of location-based services through spatial and temporal cloaking*, Proceedings of the 1st international conference on Mobile systems, applications and services, ACM, 2003, 31–42.
9. R. Shokri, C. Troncoso, C. Diaz, J. Freudiger, J. P. Hubaux, *Unraveling an Old Cloak: K-anonymity for Location Privacy*, Proceeding WPES'10 Proceedings of the 9th annual ACM workshop on Privacy in the electronic society, 2010, 115–118. Available from: https://dl_acm.xilesou.top/citation.cfm?id=1866936.
10. B. Gedik, L. Liu, Protecting location privacy with personalized k-anonymity: Architecture and algorithms, *IEEE Trans. Mobile Comput.*, **7** (2008), 1–18.
11. M. F. Mokbel, C. Y. Chow, W. G. Aref, *The new casper: Query processing for location services without compromising privacy*, VLDB '06 Proceedings of the 32nd international conference on Very large data bases, 763–774. Available from: https://dl_acm.xilesou.top/citation.cfm?id=1164193.
12. K. W. Tan, Y. Lin, K. Mouratidis, *Spatial cloaking revisited: Distinguishing information leakage from anonymity*, International Symposium on Spatial and Temporal Databases. Springer, Berlin, Heidelberg, 2009, 117–134. Available from: https://link_springer.xilesou.top/chapter/10.1007/978-3-642-02982-0_10.
13. T. Xu, Y. Cai, *Feeling-based location privacy protection for location-based services*, CCS '09 Proceedings of the 16th ACM conference on Computer and communications security, 2009. 348–357. Available from: https://dl_acm.xilesou.top/citation.cfm?id=1653704.

14. C. Bettini, X. S. Wang, S. Jajodia, *Protecting privacy against location-based personal identification*, Workshop on Secure Data Management, Springer, Berlin, Heidelberg, 2005, 185–199. Available from: https://link_springer.xilesou.top/chapter/10.1007/11552338_13.
15. K. Sampigethaya, L. Huang, M. Li, R. Poovendran, K. Matsuura, K. Sezaki, *Caravan: Providing location privacy for VANET*, Proceeding Embedded Security in Cars (ESCAR) Workshop, 2005, 13–15. Available from: https://labs.ece.uw.edu/nsl/papers/ESCAR-05.pdf.
16. G. Ghinita, P. Kalnis, S. Skiadopoulos, *Prive: Anonymous location-based queries in distributed mobile systems*, WWW '07 Proceedings of the 16th international conference on World Wide Web, 371–380, 2007. Available from: https://dl_acm.xilesou.top/citation.cfm?id=1242572.1242623.
17. J. Xu, X. Tang, H. Hu, J. Du, Privacy-consious location-based queries in mobile environments, *IEEE Trans. Parallel Distrib. Syst.*, **21** (2010), 313–326.
18. T. Ma, J. Jia, Y. Xue, Y. Tian, A. Al-Dhelaan, M. Al-Rodhaan, Protection of location privacy for moving kNN queries in social networks, *Appl. Soft Comput.*, **66** (2018), 525–532.
19. Y. Tian, M. M. Kaleemullah, M. A. Rodhaan, B. Song, A. Al-Dhelaan, T. Ma, A privacy preserving location service for cloud-of-things system, *J. Parallel Distrib. Comput.*, **123** (2019), 215–222.
20. T. Wang, J. Zeng, Z. A. Bhuiyan, H. Tian, Y. Cai; Y. Chen, et al., Trajectory Privacy Preservation Based on a Fog Structure for Cloud Location Services, *IEEE Access*, **5** (2017), 7692–7701.
21. C. Dwork, M. Naor, T. Pitassi, G. N. Rothblum, *Differential privacy under continual observation*, STOC '10 Proceedings of the forty-second ACM symposium on Theory of computing, 715–724, 2010. Available from: https://dl_acm.xilesou.top/citation.cfm?id=1806787.
22. C. Dwork, *The promise of different privacy: A tutorial on algorithmic techniques*, 2011 IEEE 52nd Annual Symposium on Foundations of Computer Science, 2011. Available from: https://www.microsoft.com/en-us/research/wp-content/uploads/2011/10/PID2016981.pdf.
23. N. Li, T. Li, S. Venkatasubramanian, *t-closeness: Pricavy beyond k-anonymity and l-diversity*, 2007 IEEE 23rd International Conference on Data Engineering, 2007, 106–115. Available from: https://ieeexplore_ieee.xilesou.top/abstract/document/4221659.
24. L. Zheng, H. Yue, Z. Li, X. Pan, M. Wu, F. Yang, k-Anonymity Location Privacy Algorithm Based on Clustering, *IEEE Access*, **6** (2017), 28328–28338.