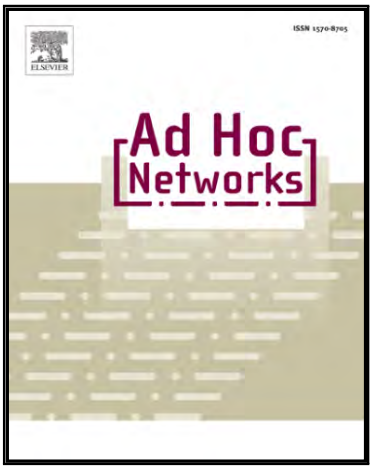


Privacy-Preserving Mobile Crowd Sensing in Ad Hoc Networks

Zhijie Wang, Dijiang Huang

PII: S1570-8705(18)30033-7
DOI: 10.1016/j.adhoc.2018.02.003
Reference: ADHOC 1637

To appear in: *Ad Hoc Networks*

Received date: 25 October 2016
Revised date: 1 July 2017
Accepted date: 7 February 2018

Privacy-Preserving Mobile Crowd Sensing in Ad Hoc Networks

Zhijie Wang, Dijiang Huang

Department of Computer Science and Engineering, Arizona State University, 699 S Mill Avenue, Tempe, AZ 85281, United States

Abstract

The presence of a rich set of embedded sensors on mobile devices has been propelling various sensing applications regarding individual activities and their surrounding environment, and these persuasive sensing-capable mobile devices are pushing the new paradigm of Mobile Crowd Sensing (MCS) from sketch to reality. MCS aims to outsource sensing data collection to Mobile Device Owner (MDO) and it could revolutionize the conventional ways of sensing data collection and processing. Nonetheless, the widespread deployment of MCS gives rise to the privacy concerns from both the MDOs and the Sensing Service Consumers (SSC), especially in the case where MCS relies on untrustworthy third-party infrastructures. This paper proposes three protocols to address the privacy issues of MCS in ad hoc network without depending on any third-parties. It first presents Privacy-Preserving Summation (PPS) protocol to protect the privacy of the SSCs. Next, it puts forward Privacy-Preserving Difference Rank Computation (PPDRC) protocol to ensure the privacy of the MDOs. Finally, it proposes Approximate K-Nearest Neighbor with Privacy Preservation(AKN2P2) to approximately identify the k-nearest neighbors without privacy leaks of both the MDOs and the SSCs. The performance evaluations demonstrate the computation overhead in different settings.

Keywords: Mobile Crowd Sensing, Ad Hoc Network, Privacy Preservation

Email addresses: `zwang134@asu.edu` (Zhijie Wang), `dijiang@asu.edu` (Dijiang Huang)

1. Introduction

Recent advances in micro-electro-mechanical systems (MEMS) technology have enabled the development of low-cost, low-power, multi-functional and small-size sensors to be embedded in mobile devices, such as smartphones, wearable devices and in-vehicle sensors. Many mobile devices come with Internet connectivity and embedded sensors (e.g., accelerometers, gyroscope, microphone, video camera, GPS, and speed sensors), thereby turning themselves into well-functioned sensor boxes to probe personal activities and environmental phenomena in the vicinity. Consequently, a new sensing paradigm named Mobile Crowd Sensing (MCS) comes into being to harness the potential of the widespread mobile sensors and describe the dynamic patterns of the physical world across a wide variety of application domains.

These sensing capable mobile devices, which consist of sensing, data processing, and communicating components, represent a significant improvement over traditional sensors networks. Most existing sensor networks still require overwhelming expenditure and professional personnel for both deployment and long-term maintenance, which are unaffordable for individuals and small companies. As a result, many areas of interest are out of sensing coverage and the collected data are insufficient for various application requirements. Worse still, the underutilization of current sensor networks and sensor-equipped mobile devices are wasteful of existing investment, as the sensor motes are static and they stay in hibernation mode most of time. Comparatively, MCS offers several advantages over the traditional sensor network infrastructures. First, MCS is built on the already-existing mobile devices with broad network access (e.g., cellular base stations and wifi access points), which are globally widespread and ready to be used. The persuasiveness of smartphones on an unprecedented scale reduces the deployment cost to almost zero, while the traditional static sensor networks involve overwhelming deployment costs that cannot be afforded by individuals and small enterprises. Furthermore, the movement of mobile device carriers implicates high spatio-temporal coverage and increases the possibility of capturing unexpected events as compared to static sensor networks. Last but not the least, the human involvement could provide additional intelligence such as persuasive user feedback on the sensor data, thus promoting the quality of service and improving the user experience of sensing applications.

Nevertheless, the potential leakage of personal information regarding the involved parties could adversely influence the growth of MCS market. On one hand, the privacy leakage comes from the queries of the SSCs. The sensing queries can be sensitive as they reflect the SSCs' interest and behavior pattern. On the other hand, the privacy leakage could occur in the process of the data collections over the MDOs. In MCS, an enormous amount of potentially sensitive information could be generated by tracking the users automatically on an ongoing basis, plentiful of sensitive information about MDOs can be collected, thereby resulting in the violation of the privacy of the MDOs' traces, interests, life styles and so on. As a result, the MDOs can be profiled thanks to the continuous personal data collection over long time periods.

Numerous techniques have been studied to secure the privacy of multiple-party data computation and data sharing in sensing scenarios. Substantial research work [17, 37, 31] have been done for privacy protection by data perturbation or dummy data generation, and these approaches incur significant overhead in mobile devices and decrease data utility. Some other approaches adopt k-anonymity [9, 21, 2], which heavily depends on the distribution and density of the mobile users, thereby rendering it impossible in many real settings. Many other approaches [3, 22, 38, 33] rely on a trusted third party to host the individual data of mobile users for sensing query requests, and compromising the third party can result in the breach of the private data.

To overcome these disadvantages, we investigate how to enforce privacy protection for the SSCs and the MDOs in MCS in ad hoc network without depending on third-party infrastructure, such that the privacy leak from third-party providers is eliminated. Homomorphic encryption plays a significant role in our proposed solutions to preserve the privacy of SSC's sensing queries and MDOs' individual data records, because it allows computations to be performed over ciphertext ending up with a ciphertext which equals the results of operations performed on the plaintext when it is decrypted. Accordingly, our paper makes novel contributions as follows:

- We first present Privacy-Preserving Summation (PPS) protocol to protect the privacy of the SSCs. It allows the SSCs to compute the summation of sensing readings of target MDOs from a larger semi-honest MDO group while no MDOs can distinguish the real target MDOs from others. Specifically, PPS protects the sensing query privacy by secretly excluding untargeted MDOs' sensing readings from the final aggregation result without the awareness of any involved MDOs.

- We put forward Privacy-Preserving Difference Rank Computation (PP-

DRC) protocol to check if the difference between each MDO's sensing reading and a baseline value d is smaller than a specific proportion of the summation of the differences corresponding to all MDOs. As the signs of polynomials can be derived without disclosing the numeric values of the data records of the involved MDOs, it secures the data privacy of the honest MDOs by comparing the difference between the sensing reading of any MDO and d against a designated proportion of the summation of all sensing reading differences without revealing the values of d and any sensing readings. It is $(n-2)$ collusion-resistant against data privacy attacks over honest MDOs where n is the total number of all MDOs.

- We construct Approximate K-Nearest Neighbor with Privacy Preservation(AKN2P2) protocol to identify the k-nearest neighboring MDOs around a Point Of Origin(POO) given by the SSC. In each iteration, polynomial inequalities in encrypted form would be constructed for each MDO's distance and a designated pivot value, and the sign of each polynomial inequality indicates if a MDO's distance is below the designated pivot value or not. Accordingly, the search range narrows down step by step in a divide-and-conquer manner until k-nearest MDOs are identified or the size of the search range has shrunk to the smallest acceptable size of the privacy window. It protects the sensing query privacy and the data privacy of MDOs at the same time.

- We give security analysis of the three protocols and offer performance evaluation to demonstrate their practicality.

The remainder of this paper is organized as follows. Section 2 gives system overview and preliminaries. Section 3, section 4 and section 5 present the construction details and security analysis of PPS, PPDRC and AKN2P2, respectively. Section 6 evaluates their performance. Section 7 discusses the related work, and Section 8 concludes this paper.

2. System Overview and Preliminaries

The MCS system is built on a generic two-tier service model as illustrated in Figure 1. A typical MCS system consists of the following parties:

- *Mobile Device Owners (MDOs)*: The geographically distributed physical sensing networks comprise of conventional sensor nodes and programmable smartphones equipped with embedded sensors. The owners of these sensing devices are termed as Mobile Device Owners (MDOs). The MDOs broadcast the information of their heterogeneous sensor nodes, either static or mobile, to maximize the utilization of the physical sensing resources. As such, the MDOs collaboratively construct the physical substrate sensing networks.

- *Sensing Service Consumers (SSCs)*: The SSCs issue queries for sensing data (e.g., "the average temperature in Phoenix in the past month") and obtain reports. Note that the SSC could be MDO at the same time.

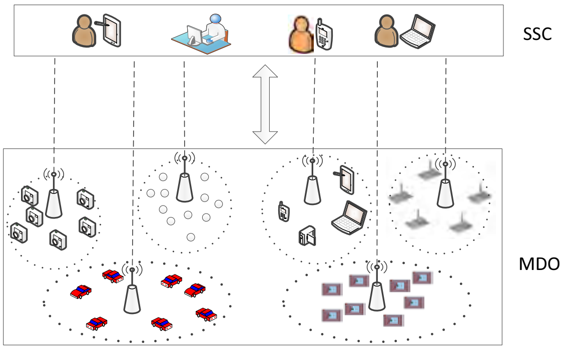


Figure 1: Architectural Overview of MCS

In the MCS infrastructure, the MDOs construct a decentralized network without pre-existing insfrastructure. They need to create sensor abstracts for sensing devices including the communication protocols, the operating systems, the deployment positions and the mobility state as well as hardware details, such as the micro-controller, on-board RAM, flash memory, battery power and storage. In addition, MDOs need to append their individual network topologies as well as adjacent reachability information to establish the links across different sensing domains such that the SSCs can communicate with them.

2.1. Homomorphic Encrypton

To address the issues stated above, PP-MCS is proposed to protect the privacy of MDOs and SSCs without relying on any trusted online third party.

It will exploit homomorphic encryptions to aggregate the private data of MDOs without revealing MDOs' individual data records. An encryption scheme is defined as an additive homomorphic one if and only if

$$E(m_1) \oplus E(m_2) = E(m_1 + m_2)$$

where $\oplus$ is an operator, and m_1 and m_2 are the numeric values to be encrypted. Similarly, an encryption scheme is defined as a multiplicative homomorphic one if and only if

$$E(m_1) \otimes E(m_2) = E(m_1 * m_2)$$

where $\otimes$ is an operator, and m_1 and m_2 are the numeric values to be encrypted.

Specifically, we rely on Paillier cryptography [26] of which the most expensive operations are encryption and decryption, while the operations with ciphertexts are relatively inexpensive [26, 13]. Given the ciphertexts $E(m_1)$ and $E(m_2)$, the public key $pk = \{g, N\}$ and the secret key $sk = \{\lambda, \mu\}$, the sum of $m_1 + m_2$ can be derived by computing

$$D(E(m_1, pk) \oplus E(m_2, pk), sk) = (m_1 + m_2)\, mod\, N$$

Additionally, we can derive the product $r * m$ from the ciphertext operations based on the multiplicative homomorphic property as follows:

$$D(E(m, pk)^r, sk) = r * m\, mod\, N$$

where r is a random number.

3. Privacy-Preserving Summation (PPS)

The PPS protocol only concerns about the SSCs' sensing query privacy. Specifically, it is designed in sophisticated ways such that some MDOs can be secretly excluded from the final aggregation result while other MDOs are privately included without the awareness of any involved MDOs. As a result, no MDO knows if it is selected as the real data source and the content privacy of the SSC's queries are preserved.

3.1. System Model and Attack Model

The PPS assumes the SSCs request the summation value of the sensing readings of some of the MDOs distributed across a geographical area. The system model is illustrated in Figure 2 and it consists of three parties:

- *Mobile Device Owner* (*MDO*): The MDOs encrypt the sensing readings upon receiving the encrypted sensing requests and send them back to a master node for aggregation.

- *Sensing Service Consumers* (*SSCs*): The SSCs issue sensing service requests for the summation/average values of the sensing readings of some MDOs in a target region. To preserve the privacy of the sensing requests, each SSC specifies a secret policy to define a cloaked ID set $\mathbb{I}^{(*)}$ consisting of the IDs of the target MDOs $\mathbb{I}^{(t)}$ and the untargeted ones, and sends out its requests based on Paillier cryptosystem. Note that the cloaked ID set has included all the MDOs to the maximum extent possible that can provide useful sensing readings for the final results to achieve the highest sensing quality. Meanwhile, it may also includes untargeted MDOs to protect the privacy of the SSC's sensing service request such that no MDOs know which MDOs are the actual target ones. It is also noteworthy that the SSCs know both $\mathbb{I}^{(*)}$ and $\mathbb{I}^{(t)}$ specified by the secret policies in their sensing service requests.

- *Master Node*: The MDOs select a MDO out of the cloaked MDO set as the master node to aggregate the sensing readings from all the MDOs. The master node could be either the target MDO or the untargeted one. It has direct communication channel with the SSC as well as the tree-like communication routes with all the other MDOs in the ad-hoc network as shown in Figure 2.

We make the following security assumptions for the attack model: 1) There exist more than one MDO and all the MDOs including the master node are semi-honest attackers. In other words, they honestly follow the procedures while they are interested in de-anomynizing the identies of the target MDOs and the final aggregation result; 2) the data privacy of the MDOs are not concerned in this scenario; that is to say, the identity of the MDO corresponding to a specific data record is not sensitive and it does not cause any issues if the SSC knows which MDO a specific sensing reading is generated by.

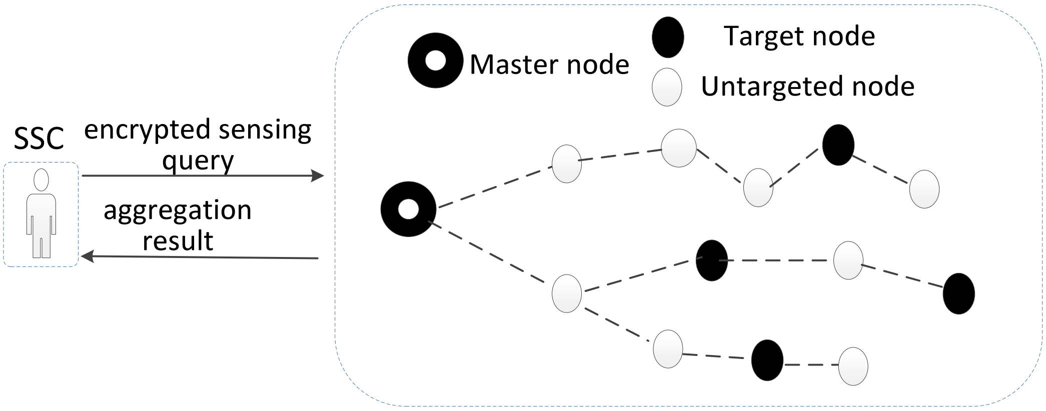


Figure 2: Privacy-Preserving Sensing Query in MCS

3.2. Construction

In this subsection, we describe how to construct our privacy-preserving sensing query scheme. The SSC selects the target MDOs $\mathbb{I}^{(t)}$ with appropriate sensing categories and geo-locations as well as a cloaking identification set $\mathbb{I}^{(*)}$ of MDOs to hide the real target MDOs $\mathbb{I}^{(t)}$, where there exist $|\mathbb{I}^{(t)}| = n^{(t)} \leq |\mathbb{I}^{*}| = n^{*} \leq n$. Each MDO_i holds the sensing reading d_i where there exist $d_i < d_{max}$ for $\forall i \in \mathbb{I}$. This protocol consists of three algorithms including Privacy-Preserving Query Generation, Response Generation and Response Retrieval as shown in **Algorithm 1** — **Algorithm 3** as follows:

Algorithm 1: PPSQ Query Generation (SSC)

```
randomly selects two large primes $p, q$ such that $N = pq > nd_{max}$ and
derives $\lambda = lcm(p-1, q-1)$ ;
generate $pk = (N, g)$ and $sk = (\lambda, \mu)$ by choosing a random $g \in \mathbb{Z}^{*}_{N^2}$
and $\mu = (L(g^{\lambda} \; mod \; N^2)^{-1} \; mod \; n$, such that
$gcd(L(g^{\lambda} \; mod \; N^2), N) = 1$, where $L(x) = (x-1)/N$ ;
For each $i \in \mathbb{I}^{(*)}$, pick a random integer $r_l \in \mathbb{Z}^{*}_{N^2}$ and computes
    $$c_i = \begin{cases} E(1, pk) = g^1 r_i^N (mod \; N^2), \text{ if } i \in \mathbb{I}^{(t)} \\ E(0, pk) = g^0 r_i^N (mod \; N^2), \text{ if } i \notin \mathbb{I}^{(t)} \end{cases}$$
returns $Q = \{\mathbb{I}^{(*)}, c_1, c_2, \cdots, c_{n^*}, pk\}$ .
```

Afterwards, the SSC sends $\{c_i, pk\}$ to each MDO_i along with the sensing frequency rate F within certain time period T through the Master Node.

Algorithm 2: PPSQ Response Generation (MDOs + Master Node)

```
1 On receiving $(c_i, pk)$, each $MDO_i$ encrypts its sensing reading $d_i$ by
  computing $C_i = c_i^{d_i}$, and sends it back to the master node as the
  response ;
2 After receiving the response $\{C_i\}_{i \in \mathbb{I}^{(*)}}$ from the all the MDOs, the
  master node computes the aggregation result $C = \prod_{i \in \mathbb{I}^{(*)}} C_i$;
3 The master node forwards $C$ to the SSC.
```

Algorithm 3: PPSQ Response Retrieval (SSC)

```
1 After receiving the aggregation result $C$ from the master node, the
  SSC performs the decryption to get the summation of the sensing
  reading values of the target MDOs by computing
  $sum = Decrypt(C, sk) = \frac{L(C^\lambda \ mod \ N^2)}{L(g^\lambda \ mod \ N^2)} \ mod \ N.$
```

Please note that for each sensing task, Algorithm 1 would run once and the sensing query is broadcasted once. The MDOs including the Master Node forward $\{c_i, pk\}_{i \in \mathbb{I}^{(*)}}$, the sensing frequency rate F and the time period T to their downstreaming child MDOs along the tree-like routes only once for the subsequent sensing reading generations and submissions. Nonetheless, Algorithm 2 and Algorithm 3 might run multiple times, because each MDO generates and transmits sensing readings back to their upstreaming parent MDOs with aggregation operations multiple times based on F and T. It is self-evident to see that the aggregation operations help save the upstreaming communication overhead significantly in the ad-hoc sensing network.

Correctness: For the target $MDO_i (i \in \mathbb{I}^{(t)})$, its sensing reading would be embedded into $C_i = c_i^{d_i} = (g^1 r_i^N (mod \ N^2))^{d_i} = g^{d_i} (r_i^{d_i})^N (mod \ N^2)$. On the contrary, for the untargeted $MDO_i (i \notin \mathbb{I}^{(t)})$, its sensing reading would be canceled out by computing $C_i = c_i^{d_i} = (g^0 r_i^N (mod \ N^2))^{d_i} = g^0 (r_i^{d_i})^N (mod \ N^2)$. Hence, the aggregation of the sensing reading values of the target MDOs can be computed by

$$C = \prod_{i \in \mathbb{I}^{(*)}} C_i = \prod_{i \in \mathbb{I}^{(t)}} C_i \cdot \prod_{i \in \mathbb{I}^{(*)}, i \notin \mathbb{I}^{(t)}} C_i = g^{\sum_{i \in \mathbb{I}^{(t)}} d_i} (\prod_{i \in \mathbb{I}^{(*)}} r_i^{d_i})^N (mod\ N^2)$$

As a result, the summation of the sensing reading values of the target MDOs can be derived by $sum = Decrypt(C, sk)$ with only the sensing reading values of the MDOs in $\mathbb{I}^{(t)}$.

3.3. Security Analysis

The Paillier cryptosystem offers semantic security[26, 13], which is secure against chosen plaintext attacks. Therefore, given the public key pk of the Paillier cryptosystem, the CP and the MDOs can hardly differentiate between the ciphertexts $c_1, c_2, \cdots, c_{n^*}$ with encrypted "0" or "1". Consequently, the MDOs including the master node cannot distinguish the target MDOs $\mathbb{I}^{(t)}$ from the untargeted MDOs in the cloaked identification set $\mathbb{I}^{(*)}$. As a result, the privacy of the sensing query of the SSC are protected. The security of PPSQ can be further proved by the theorem 2 below:

THEOREM 1. *The PPSQ preserves the privacy of sensing query of the SSC if the Paillier cyptosystem is semantically secure.*

Proof of Theorem 1. *Suppose any MDO $\mathcal{A}$ has non-negligible advantage to break the privacy of the PPSQ protocol. $\mathcal{A}$ is hereby used to break the semantic security of the Paillier cryptosystem. Given the public key of the Paillier cryptosystem, the challenger randomly chooses $b \in \{0, 1\}$and sends $Q = c_1, c_2, \cdots, c_{n^*}, (g, N)$ to $\mathcal{A}$ where $c_i = E(b, pk)$. If $\mathcal{A}$ gives the guess $i' = i$,then we out put the guess $b' = 1$ or otherwise.*

The adversary $\mathcal{A}$ wins te game with the probability of $1/2 + \epsilon_1$ when $b = 0$, or with the probability of $1/2 + \epsilon_2$ when $b = 1$ where ϵ_1 and ϵ_2 are non-negligible. Accordingly, the advantage to break the Paillier cryptosystem is $1/2(1/2 + \epsilon_1) + 1/2(1/2 + \epsilon_2) = 1/2 + (\epsilon_1 + \epsilon_2)/2$. Hence, it breaks the semantic security of the Paillier cryptosystem, which contradicts the theorem. Therefore, the privacy of sensing query of the SSC is proved.

4. Privacy-Preserving Difference Rank Computation (PPDRC)

In this section, we only concern about the data privacy of the MDOs regarding the Difference Rank Query (DRQ) along with the privacy of SSC's baseline value d. We first describe its system model and attack model. Next, two techniques are presented as the building blocks to preserve the data privacy among multiple parties, respectively. Subsequently, we construct a privacy-preserving scheme to protect MDOs' data privacy in the procedure of multi-party sensing computation.

4.1. System Model and Attack Model

The DRQ is defined as follows: assume a SSC has a baseline value d, and some MDOs $\mathbb{I} = \{0, 1, \cdots, n\}$ have the sensing reading $\{d_i\}_{i\in\mathbb{I}}$ and their absolute differences from d are $\{\Delta_i\}_{i\in\mathbb{I}} = \{|d_i - d|\}_{i\in\mathbb{I}}$ where $|\mathbb{I}| = n$ and $n \geq 3$. The SSC is concerned about which MDO_i have the absolute difference $|\Delta_i|$ above a certain proportion p_1/p_2 of the summation of the absolute differences $\sum_{i\in\mathbb{I}} \Delta_i$ where $p_1, p_2 \in \mathbb{Z}$ and $0 < p_1 < p_2$, such that $\Delta_i = |d_i - d| \geq \dfrac{p_1 \sum_{i\in\mathbb{I}} \Delta_i}{p_2}$ as shown in Figure 3 without revealing d to any MDOs. The value of d is determined by the SSC based on the application requirement. For example, if the SSC is interested in learning which MDO_i has the sensing reading d_i with the absolute difference above the average sensing readings, it can be reduced to the special case to check the inequality $d_i \geq \dfrac{\sum_{i\in\mathbb{I}} d_i}{n}$ where $d = 0$, $p_1 = 1$ and $p_2 = n - 1$. In some other use cases, d can be set as the max/min/mean value over the historical data based on the application needs.

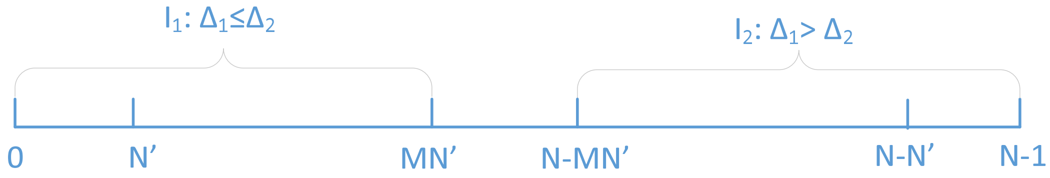


Figure 3: Privacy-Preserving Integer Comparison in MCS

We make the following security assumptions for the attack model: 1)At most $n - 2$ out of n MDOs are semi-honest attackers. In other words, they honestly follow the procedures, while the malicious MDOs are interested in other MDOs' sensing readings and the SSC's baseline value d, and the SSC is interested in all MDOs' sensing readings; ii) the privacy of the sensing query of the SSC is out of concern; iii)the MDOs are $n - 2$ collusion-resistant; in other words, at most $n - 2$ out of n MDOs collude with the SSC to breach the data privacy of the honest MDOs. We believe this assumption is feasible, as a few MDOs could be still honest in most application settings.

4.2. Building Block I: Privacy-Preserving Comparision of Two Integers

The millionaire's problem has bee proposed and addressed by Yao [35]. Its goal is to solve the inequality $\Delta_1 \geq \Delta_2$ without revealing the actual values of Δ_1 owned by Party A and Δ_2 owned by Party B, respectively. Ghinita *et al.* [13] proposed an easy solution to this problem based on Paillier cryptosystem with order of N. It assumes that $\Delta_1, \Delta_2 \in \mathbb{Z}_{N'}$ where $N' \ll (N-1)/2$. Party A generates the public/private key pair pk/sk, and sends $E(N-\Delta_1, pk)$ to Party B. Accordingly, B generates a random integer $r \in \mathbb{Z}_M^*$ as a blinding factor where $M \leq \lfloor \frac{N-1}{2N'} \rfloor$ and computes

$$(E(N-\Delta_1, pk) \oplus E(\Delta_2, pk))^r = E(N+\Delta_2-\Delta_1, pk)^r = E(r(N+\Delta_2-\Delta_1), pk)$$

and sends it back to A. Subsequently, A decrypts this message and derives $r(N+\Delta_2-\Delta_1)$. If $\Delta_2-\Delta_1 \geq 0$, then $r(N+\Delta_2-\Delta_1) \in I_1 = \{0, 1, \cdots, M\cdot N'\}$, otherwise $r(N+\Delta_2-\Delta_1) \in I_2 = \{N-M\cdot N', \cdots, N-1\}$ where $I_1 \bigcap I_2 = \emptyset$.

Ghinita *et al.* [13] pointed out this approach is feasible in real-world settings. It suggested the magnitude of the modulus N should be at least 768 bits large to guarantee security strength, and the values of Δ_1 and Δ_2 can be represented by 64 bits, which suffice in most real-world applications. At the same time, the random blinding factor domain will be bounded by $M = \frac{2^{768}}{2\cdot 2^{64}}$ with the order of 2^{700}, which is sufficiently large to provide a strong degree of security.

4.3. Building Block II: Secret Sharing Among Distributed Multiple Parties

Chase *et al.* [6] proposed a secret-sharing technique among n distributed parties with $(n-2)$ collusion-resistance, which implies at least 2 out of n parties are outside the collusion group. It assumes each pair of parties (i, j) share $s_{ij} = PRF_{k_{ij}}(i+j)$ where $i, j \in \mathbb{Z}_N^*$, $PRF_{k_{ij}}(\dot{})$ is a pseudo-random function as described in [24], and k_{ij} is the shared secret between MDO_i and MDO_j. As such, each party i generates a function $F(i) = \sum_{j<i, j\in\mathbb{Z}_N^*} s_{ij} - \sum_{j>i, j\in\mathbb{Z}_N^*} s_{ij}$ as the blinding value such that there exists $\sum_{i\in\mathbb{Z}_N^*} F(i) = 0$. In this manner, any adversary who knows the secrets of no more than $n-2$ parties cannot derive the secrets of the remaining parties. This technique can be used to protect the data privacy in the process of multi-party computation as detailed in the next section.

4.4. Construction

Without loss of generality, we assume there exists n MDOs in the system and each MDO_i has a sensing reading $d_i \in [0, N']$. For each $d_i \in \mathbb{Z}_{N'}$, it can be converted to a binary vector $\vec{d_i} = \{\hat{d}_{i,0}, \hat{d}_{i,1}, \cdots, \hat{d}_{i,N'-1}\}$ where $\hat{d}_{i,k} = 1 (k \in [0, d_i])$ and $\hat{d}_{i,k} = 0 (k \in (d_i, N'-1])$. The SSC has a baseline value d, and it generates Paillier public key $pk = (N, g)$ and private key sk such that $nN' \ll N$. By the same token, d can also be converted to a binary vector $\vec{d} = \{\hat{d}_0, \hat{d}_1, \cdots, \hat{d}_{N'-1}\}$ where $\hat{d}_k = 1 (k \in [0, d])$ and $\hat{d}_k = 0 (k \in (d, N'-1])$. Similar to [39], we can derive the absolute difference between d_i and d by computing

$$\begin{aligned} \Delta_i &= |d_i - d| = \sum_{k=0}^{N'-1} |\hat{d}_{i,k} - \hat{d}_i| = \sum_{k=0}^{N'-1} |\hat{d}_{i,k} - \hat{d}_i|^2 \\ &= \sum_{k=0}^{N'-1} \hat{d}_{i,k}^2 - 2 \sum_{k=0}^{N'-1} \hat{d}_{i,k} \hat{d}_i + \sum_{k=0}^{N'-1} \hat{d}_i^2 \\ &= \sum_{k=0}^{N'-1} \hat{d}_{i,k}^2 - 2 \vec{d_i} \vec{d} + \sum_{k=0}^{N'-1} \hat{d}_i^2 \end{aligned}$$

Accordingly, our protocol consists of three algorithms including PPDRC's query generation, response generation and response retrieval as follows. The SSC generates DRQ as shown in **Algorithm 4** :

Algorithm 4: PPDRC Query Generation (SSC)

```
The SSC constructs a vector $\vec{d} = \{\hat{d}_0, \hat{d}_1, \cdots, \hat{d}_{N'-1}\}$ where
  $\hat{d}_k = 1(k \in [0, d))$ and $\hat{d}_k = 0(k \in [d, N'-1])$ based on its baseline
  value $d$;
The SSC picks a distinct random integer $r_k, r'_k \in \mathbb{Z}_N$ and computes
  $\prod_{k=0}^{N'-1} E(\hat{d}_k^2, pk) = E(\sum_{k=0}^{N'-1} \hat{d}_k^2, pk) = E(d, pk) = g^d r_k^N (mod\ N^2)$ and
  $E(\hat{d}_k, pk) = g^{\hat{d}_k} r_k'^N (mod\ N^2)$ for each $k \in [0, N'-1]$ ;
SSC transmits $(pk, E(d, pk), \{E(\hat{d}_k, pk)\}_{k \in [0, N'-1]})$ to the MDOs.
```

The SSC sends $\{E(\hat{d}_i, pk), pk\}$ to each MDO_i, and the MDOs generate response as shown in **Algorithm 5** :

We herein explain the correctness of **Algorithm 5** described above. Each MDO_i first computes the encrypted sensing reading difference $E(\Delta_i)$ without knowing the real values of d and Δ_i. Subsequently, the MDOs compute

$$\begin{aligned} E_{sum} &= E(\sum_{i=1}^{n} (\Delta_i + P_i), pk) \\ &= E(\sum_{i=1}^{n} (\Delta_i + \sum_{j<i, j \in \mathbb{Z}_N^*} s_{ij} - \sum_{j>i, j \in \mathbb{Z}_N^*} s_{ij}, pk) \\ &= E(\sum_{i=1}^{n} \Delta_i, pk) \end{aligned}$$

Algorithm 5: PPDRC Response Generation (MDOs)

```
$Response = \{Resp_1, Resp_2, \ldots, Resp_n\} \leftarrow \{1, 1, \ldots, 1\}, E_{sum} \leftarrow 1$ ;
foreach $MDO_i \in \{MDO_{i'}\}_{i' \in [1,n]}$ do
    The $MDO_i(1 \le i \le n)$ constructs $\vec{d_i} = (\hat{d}_{i,0}, \hat{d}_{i,1}, \cdots, \hat{d}_{i,N'-1})$ where $\hat{d}_{i,k} = 1(k \in [0, d_i))$ and $\hat{d}_{i,k} = 0(k \in [d_i, N'-1])$, and computes $E(d_i, pk) = E(\sum_{k=0}^{N'-1} \hat{d}_{ik}^2, pk) = \prod_{k=0}^{N'-1} E(\hat{d}_{ik}^2, pk) = g^{d_i} r_k^N (mod\, N^2)$ ;
    The $MDO_i(1 \le i \le n)$ selects distinct random integers $\{r_{ik}\}_{k \in [0,d_i]}$ where $r_{ik} \in \mathbb{Z}_{N'}^*$ and computes $E(\vec{d} \cdot \vec{d_i}, pk) = E(\sum_{k=0}^{d_i} \hat{d}_k \hat{d}_{ik}, pk)$ $= \prod_{k=0}^{d_i} E(\hat{d}_k \hat{d}_{ik}, pk)\ mod\ N^2$;
    The $MDO_i(1 \le i \le n)$ further computes $E(-2\vec{d} \cdot \vec{d_i}, pk) = E((N-2)\vec{d} \cdot \vec{d_i}, pk) = E^{(N-2)}(\vec{d} \cdot \vec{d_i}, pk)$ ;
    Consequently, each $MDO_i(1 \le i \le n)$ computes $E(\Delta_i) = |d_i - d| = E(\sum_{k=0}^{N'-1} \hat{d}_{i,k}^2 - 2\vec{d_i}\vec{d} + \sum_{k=0}^{N'-1} \hat{d}_i^2) = E(d_i, pk) \cdot E^{(N-2)}(\vec{d} \cdot \vec{d_i}, pk) \cdot E(d, pk)$ ;
end
foreach $MDO_i \in \{MDO_{i'}\}_{i' \in [1,n]}$ do
    The $MDO_i$ shares a secret $s_{ij}$ with $MDO_j$ where $i, j \in [1, n], j \neq i$, $s_{ij} = s_{ji} \in \mathbb{Z}_N^*$, and derives $P_i = \sum_{j<i, j \in \mathbb{Z}_N^*} s_{ij} - \sum_{j>i, j \in \mathbb{Z}_N^*} s_{ij}$ ;
    The $MDO_i$ computes $E_{sum} = E_{sum} \cdot E(\Delta_i, pk) \cdot E(P_i, pk)$ ;
end
The $MDO_n$ broadcasts $E_{sum}$ to all the MDOs;
foreach $i \in \{1, 2, \ldots, n\}$ do
    The $MDO_i$ picks a random integer $r_{i1}, r_{i2} \in \mathbb{Z}_{N'}^*$ where $r_{i2} > r_{i1} > 0$, and computes $Resp_t = ((E_{sum})^{p_1} \cdot E(\Delta_i, pk)^{N-p_2})^{r_{i2}} \cdot E(r_{i1}, pk)$ ;
end
returns $Response$ .
```

```
Algorithm 6: PPDRC Response Retrieval (SSC)
$Result = \{isLower_1, isLower_2, \dots, isLower_n\} \leftarrow \{false, \dots, false\}$ ;
foreach $MDO_i \in \{MDO_{i'}\}_{i' \in [1,n]}$ do
    $result_i = Decrypt(Resp_i, sk) = r_{i2}(\sum_{j \neq i, 1 \leq j \leq n} \Delta_j - (n-1)\Delta_t) + r_{i1}\, mod\, N$ ;
    if $result_i \in [0, nN']$ then
        $isLower_i \leftarrow true$ ;
    else if $result_t \in [N - nN' - 1, N - 1]$ then
        $isLower_i \leftarrow false$;
end
returns $Result$;
```

Afterwards, **Algorithm 5** takes n iterations to yield $Response = \{Resp_1, Resp_2, \dots, Resp_n\}$. Specifically, for MDO_t, we have

$$\begin{aligned} Resp_t &= ((E_{sum})^{p_1} \cdot E(\Delta_t, pk)^{N-p_2})^{r_{t2}} \cdot E(r_{t1}, pk) \\ &= E(r_{t2}(p_1 \textstyle\sum_{i=1}^{n} \Delta_i + (N - p_2)\Delta_t) + r_{t1}), pk) \\ &= E(r_{t2}(p_1 \textstyle\sum_{i=1}^{n} \Delta_i - p_2\Delta_t) + r_{t1}), pk) \end{aligned}$$

Accordingly, the SSC can derive $r_{t2}(p_1 \sum_{i=1}^{n} \Delta_i - p_2\Delta_t) + r_{t1}$ by decrypting $Resp_t$ with the secret key sk and infers the sign of $r_{t2}(p_1 \sum_{i=1}^{n} \Delta_i - p_2\Delta_t) + r_{t1}$. For $y = r_{t2}x + r_{t1}$ where $x \in \mathbb{Z}$, it is self-evident that if $x \geq 0$ then $y > 0$ and $x < 0$ then $y < 0$ because of $r_{t2} > r_{t1}$. Therefore, the SSC gets the knowledge that if $\Delta_i = |d_i - d|$ is smaller than $\dfrac{p_1 \sum_{i=1}^{n} \Delta_i}{p_2}$ or not.

4.5. Security Analysis

The PPDRC offers semantic security against individual data privacy attacks. The security of PPDRC can be further proved by the theorem 2 below:

THEOREM 2. *The PPDRC is $(n-2)$ collusion-resistant against data privacy attacks over honest MDOs where n is the total number of all MDOs.*

Proof of Theorem 2. *Suppose the SSC colludes with $n-m$ dishonest MDOs $\mathcal{A}$ to break the data privacy of m honest MDOs with non-negligible advantage where $m \geq 2$. As the SSC posses the private key of the Paillier cryptosystem, the aggregation of $E(\Delta_i + P_i, pk)$ is reduced to the summation $\sum_{i=1}^{n}(\Delta_i + P_i)$.In* *the multi-party computation process, for each honest MDO_i, its Δ_i is protected by s_{ij} shared with any other honest MDO_j. In addition, for each honest pair MDO_i and MDO_j in the operation of $\Delta_i + P_i + \Delta_j + Pj$, Δ_i or Δ_j is protected by the summation $\Delta_i + \Delta_j$. Hence, breaking PPDRC security is equivalent to breaking the security of k-out-of-n secret sharing, which contradicts the theorems in [24]. Therefore, the PPDRC security is proved.*

5. Approximate K-Nearest Neighbors with Privacy Preservation(AKN2P2)

The query for the k-nearest neighbors has significant implications in location-based sensing scenarios, and the K-Nearest Neighbors algorithm can be used in numerous fields of applications including classification and regression. In this section, we discuss how to identify the k-nearest neighboring MDOs around a Point Of Origin(POO) given by the SSC with the guarantee of the privacy window with the smallest size δ for any MDOs in the worst case. Specifically, the locations of any MDOs remain hidden, and the neighbors have no knowledge of their distance to the POO, while the distances between the POO and any of its neighbors are masked by privacy windows with the smallest size δ.

5.1. System Model and Attack Model

The POO provided by the SSC is $L_0 = (x_0, y_0)$, and the location of each neighbor $MDO_i(i \in [1, n])$ is denoted by $L_i = (x_i, y_i)$. Note the locations are derived from the latitude x_i and the longitude y_i which are both integers (e.g.,$(33.423856, 111.939575) \rightarrow (33423856, 111939575)$). Assume d^* is the distance threshold to separate the POO's actual k-nearest neighbors from other neighbors, and we define a privacy window δ for privacy preservation, such that the neighbors falling within the distance range $[d^*, d^* + \delta]$ can be taken as alternative ones equivalent to some actual k-nearest neighbors. We believe this assumption holds as many location-based sensing service applications are tolerant to location deviations to some extent.

We make the following security assumptions for the attack model: 1)All the involving parties are semi-honest attackers. In other words, they honestly follow the procedures, but the SSC is interested in the MDOs' locations, and the MDOs attempt to pinpoint the PPO; ii) the SSC should not know the exact distances of the MDOs to the PPO due to the privacy window with the smallest size δ, and each MDO cannot learn its distance to the PPO.

Without loss of generality, we assume there exist $x_i, y_i \in [1, N^{(L)}]$. For the sake of security, the SSC generates $N^{(L)}, N^{(\gamma)}$ where $2(N^{(L)})^2 N^{(\gamma)} \leq \lfloor \frac{N-1}{2} \rfloor$. Accordingly, the SSC calculates the distance array $D = \{D[k]\}_{k \in [1,K]}$ as shown in Figure 4 (i.e., $D = \{0, 1, \sqrt{2}, 2, \sqrt{5}, \sqrt{8}, 3, \sqrt{10}, \sqrt{13}, \cdots\}$) based on all the possible distances within the range $[0, \sqrt{2}N^{(L)}]$. At the same time, all the MDOs agree on the minimum size w_{min} (e.g., $w_{min} = 2$) of the privacy window and make it public. In each iteration of the process, the SSC privately updates the identity set $\mathbb{I}_l$, which includes the indices of $|\mathbb{I}_l|$ nearest neighbors where $|\mathbb{I}_l| \leq k$. At the same time, it also privately updates $|\mathbb{I}_r|$, which includes the indices of $|\mathbb{I}_r|$ nearest neighbors where $|\mathbb{I}_r| > k$. The implication of $N^{(\gamma)}$ is described in the following part. **Algorithm 7** elaborates how the SSC approximately identifies k nearest neighbors by narrowing down the privacy window and shifting the pivot value *pivot* via a divide-and-conquer approach. This algorithm adopts a binary search approach with complexity of $O(\log N)$. As D is a pre-sorted array and the maximal distance would not exceed N based on the assumption, the complexity of this algorithm is $O(n \log N)$ where n is the number of MDOs involved.

5.2. Construction

Correctness of Algorithm 7: We hereby provides the proof of the correctness of **Algorithm 7**. Given $(E(x_0^2 + y_0^2, pk), E(x_0, pk), E(y_0, pk))$, MDO_i computes $E(-2x_0x_i, pk) = E((N-2)x_0x_i, pk) = E(x_0, pk)^{(N-2)x_i}$, $E(-2y_0y_i, pk) = E((N-2)y_0y_i, pk) = E(y_0, pk)^{(N-2)y_i}$, and derives $E(\Delta_i) = E((x_i - x_0)^2 + (y_i - y_0)^2, pk) = E(x_i^2 + y_i^2 + x_0^2 + y_0^2 - 2x_ix_0 - 2y_iy_0, pk) = E(x_i^2 + y_i^2, pk) \cdot E(x_0^2 + y_0^2, pk) \cdot E(-2x_ix_0, pk) \cdot E(-2y_iy_0, pk)$.

In t-th iteration, MDO_i perturbs $\Delta_i - D^2[pivot^{(t)}]$ by $\gamma_{i1}^{(t)}$ and $\gamma_{i2}^{(t)}$ where $\gamma_{i1}^{(t)} > \gamma_{i2}^{(t)}$. As a result, if $\Delta_i - D^2[pivot^{(t)}] > 0$, there exists $\gamma_{i1}^{(t)}(\Delta_i - D^2[pivot^{(t)}]) - \gamma_{i2}^{(t)} > 0$; if $\Delta_i - D^2[pivot^{(t)}] \leq 0$, there exists $\gamma_{i1}^{(t)}(\Delta_i - D^2[pivot^{(t)}]) - \gamma_{i2}^{(t)} < 0$.

Accordingly, the SSC can derive n linear inequalities in t-th iteration as follows:

$$r_{11}^{(t)}(\Delta_1 - D^2[pivot^{(t)}]) - r_{12}^{(t)} \lesseqgtr 0$$

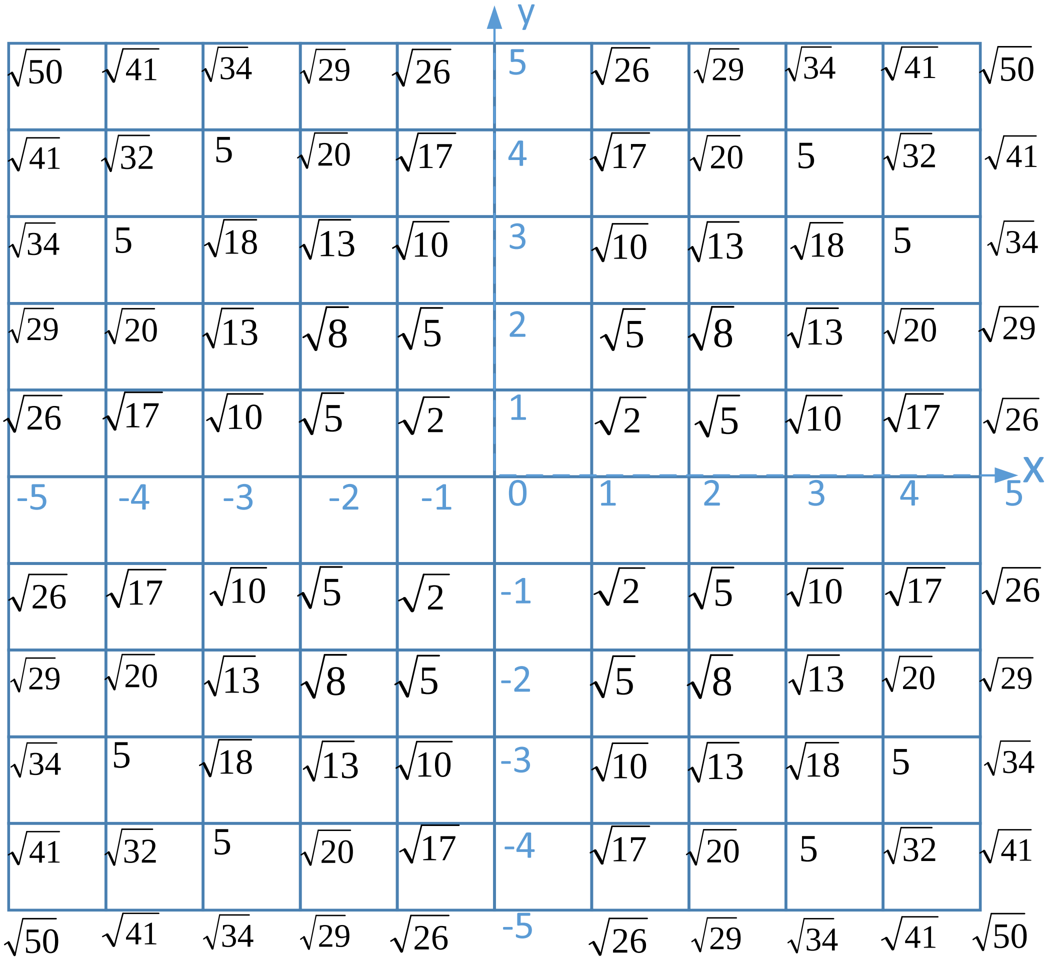
$$r_{21}^{(t)}(\Delta_2 - D^2[pivot^{(t)}]) - r_{22}^{(t)} \lesseqgtr 0$$

$$\cdots$$

$$r_{n1}^{(t)}(\Delta_n - D^2[pivot^{(t)}]) - r_{n2}^{(t)} \lesseqgtr 0$$

Algorithm 7: Approxiamte K-Nearest Neighbors with Privacy Window

```
The SSC generates Paillier public/private key $pk = \{g, n\}$,
$sk = \{\lambda, \mu\}$, $E = (E(x_0^2 + y_0^2, pk), E(x_0, pk), E(y_0, pk))$, initializes
$\overrightarrow{sign} = \{sign_i\}_{i \in [1,n]} \leftarrow \{1, 1, \ldots, 1\}$, the loop index $t \leftarrow 0$, $\mathbb{P}^{(0)} \leftarrow \emptyset$,
$\mathbb{I}_l \leftarrow \emptyset$ and $\mathbb{I}_r \leftarrow [1, n]$. It also pre-computes all the possible distances
within the range $[1, 16N^{(\gamma)}(N^{(L)})^2]$ and derives the array
$D = \{D[k]\}_{k \in [1,K]}$. Accordingly, it initializes the privacy window
$\vec{w} = \{w_{left}, w_{right}\} \leftarrow [0, K-1]$ and publishes $(pk, E, \overrightarrow{sign}, t, D, \vec{w})$;
while the number of elements in $\overrightarrow{sign}$ equivalent to $1$ is not $k$ do
    The SSC sets $t \leftarrow t + 1$, $pivot^{(t)} \leftarrow \lfloor \frac{w_{left} + w_{right}}{2} \rfloor$ ;
    if $|\mathbb{P}^{(t-1)}| \geq 1$ and $Diff_{min}(pivot^{(t)}, \mathbb{P}^{(t-1)}) \leq w_{min}$ then break end;
    $\mathbb{P}^{(t)} \leftarrow \mathbb{P}^{(t-1)} \cup \{pivot^{(t)}\}$ ;
    foreach $MDO_i \in \{MDO_{i'}\}_{i' \in [1,n]}$ do
        $MDO_i$ computes $E(-2x_0x_i, pk)$, $E(x_i^2 + y_i^2, pk)$, $E(-2y_0y_i, pk)$
        and derives $E(\Delta_i, pk) = E((x_i - x_0)^2 + (y_i - y_0)^2, pk)$ just once;
        it also picks random $\gamma_{i1}^{(t)} \in (1, N^{(\gamma)}]$, $\gamma_{i2}^{(t)} \in [1, \gamma_{i1})$, computes
        $E(N - D^2[pivot^{(t)}], pk)$, derives
        $E(\gamma_{i1}^{(t)}(\Delta_i - D^2[pivot^{(t)}]) - \gamma_{i2}^{(t)}, pk)$ and sends it to the SSC ;
        The SSC derives $\gamma_{i1}^{(t)}(\Delta_i - D^2[pivot^{(t)}]) - \gamma_{i2}^{(t)}$ by decryption,
        and updates $sign_i$ as 1 if it is positive or $-1$ if it is negative ;
    end
    if more than $k$ $sign_i \in \overrightarrow{sign}$ is $-1$ then
    $\mathbb{I}_r \leftarrow \{i | sign_i = -1, i \in [1, n]\}$, $w_{right} \leftarrow pivot^{(t)}$ else
    $\mathbb{I}_l \leftarrow \{i | sign_i = -1, i \in [1, n]\}$, $w_{left} \leftarrow pivot^{(t)}$ end;
end
The SSC randomly picks $k - |\mathbb{I}_l|$ $MDO_i$ where $i \in (\mathbb{I}_r \backslash \mathbb{I}_l)$ and adds
them to $\mathbb{I}_l$;
return $\mathbb{I}_l$ ;
```



$\sqrt{50}$	$\sqrt{41}$	$\sqrt{34}$	$\sqrt{29}$	$\sqrt{26}$	5	$\sqrt{26}$	$\sqrt{29}$	$\sqrt{34}$	$\sqrt{41}$	$\sqrt{50}$
$\sqrt{41}$	$\sqrt{32}$	5	$\sqrt{20}$	$\sqrt{17}$	4	$\sqrt{17}$	$\sqrt{20}$	5	$\sqrt{32}$	$\sqrt{41}$
$\sqrt{34}$	5	$\sqrt{18}$	$\sqrt{13}$	$\sqrt{10}$	3	$\sqrt{10}$	$\sqrt{13}$	$\sqrt{18}$	5	$\sqrt{34}$
$\sqrt{29}$	$\sqrt{20}$	$\sqrt{13}$	$\sqrt{8}$	$\sqrt{5}$	2	$\sqrt{5}$	$\sqrt{8}$	$\sqrt{13}$	$\sqrt{20}$	$\sqrt{29}$
$\sqrt{26}$	$\sqrt{17}$	$\sqrt{10}$	$\sqrt{5}$	$\sqrt{2}$	1	$\sqrt{2}$	$\sqrt{5}$	$\sqrt{10}$	$\sqrt{17}$	$\sqrt{26}$
-5	-4	-3	-2	-1	0	1	2	3	4	5
$\sqrt{26}$	$\sqrt{17}$	$\sqrt{10}$	$\sqrt{5}$	$\sqrt{2}$	-1	$\sqrt{2}$	$\sqrt{5}$	$\sqrt{10}$	$\sqrt{17}$	$\sqrt{26}$
$\sqrt{29}$	$\sqrt{20}$	$\sqrt{13}$	$\sqrt{8}$	$\sqrt{5}$	-2	$\sqrt{5}$	$\sqrt{8}$	$\sqrt{13}$	$\sqrt{20}$	$\sqrt{29}$
$\sqrt{34}$	5	$\sqrt{18}$	$\sqrt{13}$	$\sqrt{10}$	-3	$\sqrt{10}$	$\sqrt{13}$	$\sqrt{18}$	5	$\sqrt{34}$
$\sqrt{41}$	$\sqrt{32}$	5	$\sqrt{20}$	$\sqrt{17}$	-4	$\sqrt{17}$	$\sqrt{20}$	5	$\sqrt{32}$	$\sqrt{41}$
$\sqrt{50}$	$\sqrt{41}$	$\sqrt{34}$	$\sqrt{29}$	$\sqrt{26}$	-5	$\sqrt{26}$	$\sqrt{29}$	$\sqrt{34}$	$\sqrt{41}$	$\sqrt{50}$

Figure 4: Illustration of All Possible Distances

Given the array $D = \{D[k]\}_{k\in[1,K]}$ including all the possible distances, the goal of the SSC is to find the privacy window $[w_{left}, w_{right}]$ with the minimum size $|w_{right} - w_{left}| \geq w_{min}$ where $\mathbb{I}_l$ denotes the MDOs with the distance below $pivot^t$ of which the number if less than k, and $\mathbb{I}_r$ denotes the MDOs with the distance below $pivot^t$ of which the number is more than k . The first iteration starts with $[w_{left}, w_{right}] = [0, K]$ and $pivot^{(t)} = \lfloor \frac{w_{left} + w_{right}}{2} \rfloor$. In the t-th iteration, if there exists more than k linear inequality with '$<$' sign, w_{right} should be decreased by setting $w_{right} = pivot^{(t)}$ in the $(t + 1)$-th iteration;

otherwise, w^{left} should be increased by setting $w_{left} = pivot^{(t)}$ in the $(t+1)$-th iteration. The iterations stop when there exists k linear inequality with the '$<$' sign or the privacy window size has shrunk to w_{min} indicated by $Diff_{min}(pivot^{(t)}, \mathbb{P}^{(t-1)}) \leq w_{min}$, which denotes the minimum difference between $pivot^{(t)}$ and any element of $\mathbb{P}^{(t-1)}$. Aside from the $\mathbb{I}_l$ selected MDOs, the SSC randomly selects another $k - |\mathbb{I}_l|$ MDOs from the MDOs whose distances fall within $[D[w_{left}], D[w_{right}]$, because we assume selecting any $k - \mathbb{I}_l$ MDOs with distance within $[D[w_{left}], D[w_{right}]$ approximate the real $k - \mathbb{I}_l$ MDOs out of the actual k-nearest MDOs. Figure 5 illustrates in the search process of 6-nearest neighbors how the privacy window $\vec{w} = \{w_{left}, w_{right}\}$ shrinks from $\{0, 20\}$ at the beginning to $\{0, 10\}$ in the 1st iteration, and then $\{5, 10\}$ in the 2nd iteration, and ends up with $\{5, 7\}$ in the 3rd iteration.

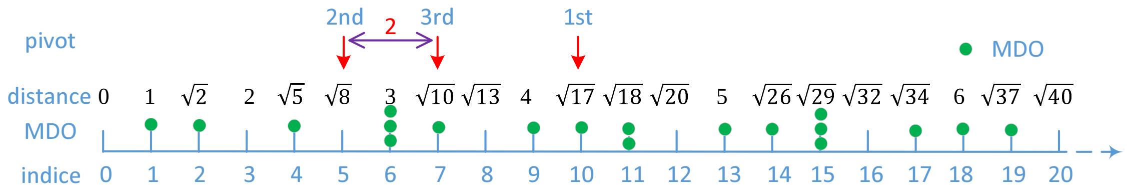

pivot
2nd
2
3rd
1st
● MDO
distance 0 1 √2 2 √5 √8 3 √10 √13 4 √17 √18 √20 5 √26 √29 √32 √34 6 √37 √40
MDO
indice 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

Figure 5: Search Process

5.3. Security Analysis

The AKN2P2 offers semantic security against privacy attacks. The security of AKN2P2 can be further proved by the theorem 3 below:

THEOREM 3. *The AKN2P2 ensures the SSC's query privacy for the POO* (x_0, y_0) *and* k *against the honest-but-curious MDOs even if they collude with each other, and it also protects MDOs' data privacy at the same time.*

Proof of Theorem 3. *The MDOs receive and operate only on ciphertexts* $E = (E(x_0^2 + y_0^2, pk), E(x_0, pk), E(y_0, pk))$*, and they do not have the SSC's private key. As such, the computationally constrained MDOs cannot learn the POO* (x_0, y_0) *through decryption due to the semantic security of the Paillier cryptosystem. In addition, the SSC keeps the value* k *and the signs of all the linear inequalities in the iterative kNN search process hidden from the MDOs, and thus the MDOs cannot get the knowledge of the approximate distances from the POO* (x_0, y_0)*, thereby preventing them from narrowing down the* *possible range of (x_0, y_0) by collusion. Therefore, the AKN2P2 ensures the SSC's sensing query privacy.*

Meanwhile, $MDO_i (i \in [1, n])$ generates random $\gamma_{i1}^{(t)}$ and $\gamma_{i2}^{(t)}$ in the t-th iteration to mask the value of $\Delta_i - D^2[pivot^{(t)}]$ where $\gamma_{i1}^{(t)} > \gamma_{i2}^{(t)}$. As a result, it is impossible for the SSC to derive Δ_i with the known $D^2[pivot^{(t)}]$ due to the lack of knowledge of $\gamma_{i1}^{(t)}$ and $\gamma_{i2}^{(t)}$. Furthermore, the malicious SSC might manipulate $[w_{left}, w_{right}]$ in the t-th iteration in attempt to get the knowledge of the exact distance from (x_0, y_0) to each MDO. However, as $[w_{left}, w_{right}]$ and $pivot^{(t)}$ are made public in each iteration, any MDOs can easily check if the SSC honestly meet the following security requirements: i) $[w_{left}, w_{right}]$ and $pivot^{(t)}$ are updated in a divide-and-conquer manner correctly; ii) the size of the privacy window $Diff_{min}(pivot^{(t)}, \mathbb{P})$ decreases by iterations; iii) $Diff_{min}(pivot^{(t)}, \mathbb{P}) \geq w_{min}$. If not, this malicious behavior can be detected immediately and any MDOs can decline to continue the process. Therefore, the AKN2P2 also ensures the SSC's sensing query privacy.

6. Experiment

The proposed privacy-preserving schemes are emulated on the DEll OPTIPLEX 390 desktop with Intel(R) Core(TM) i3-2100 CPU at 3.10GHz and 4GB memory running 64-bit Windows 7 Enterprise. The whole processes are programmed based on the java version of Paillier cryptosystem provided by Kun Liu in UMBC[19]. The emulations assess the computational cost of different scheme in terms of timing or the number of iterations without considering the communication cost.

Figure 6 illustrates the results of the average computational cost in PPSQ. In the emulation, a random number $|\mathbb{I}^t|$ of the real target MDOs $\mathbb{I}^t$ are selected from the cloaking identification set $\mathbb{I}^*$ with the size $|\mathbb{I}^*|$ where $|\mathbb{I}^t| \in \{1, 2, 3, \cdots 18\}$ and $|\mathbb{I}^*| \in \{20, 30, 40\}$. The average computational cost of each customized selection is derived by running random selections with the same $|\mathbb{I}^t|$ and $|\mathbb{I}^*|$ for 10 times. It can be learned that the average computation cost almost remains constant regardless of the number of the actually selected MDOs when the size of the cloaking identification set $\mathbb{I}^*$ is fixed. This also proves the privacy preservation as the malicious MDOs cannot estimate the number of the actually selected MDOs based on the computational cost. In addition, the average computation cost increases with the size of the cloaking identification set $\mathbb{I}^*$ as more MDOs get involved into the computation process.

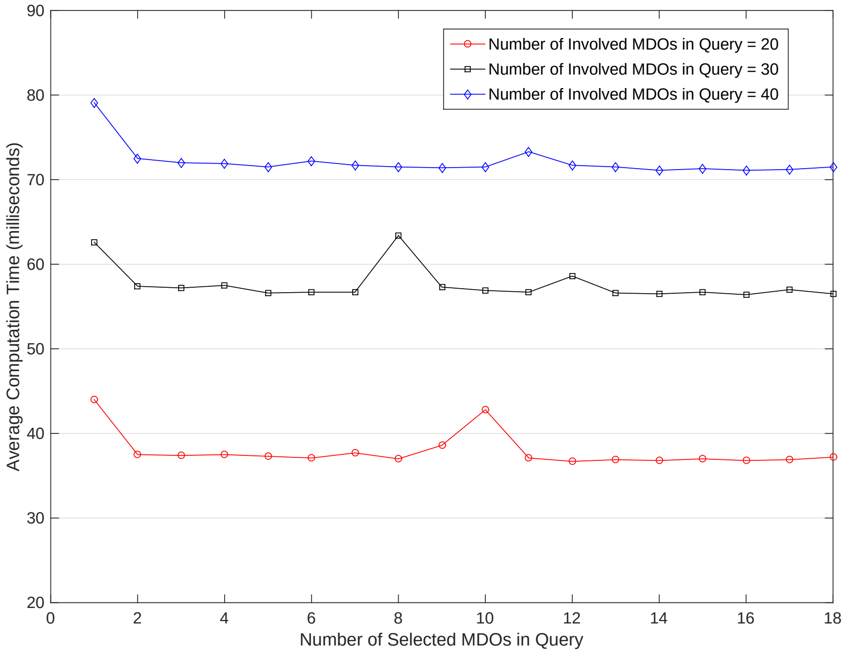


Figure 6: The Average Computational Cost Regarding Different Number of Target MDOs

Figure 7 illustrates the results of the average computational cost in PP-DRC. The emulation runs 10 times for each designated number of involved MDOs. It can be learned that the average computation cost increases with the number of the involved MDOs, as the addition and the comparison of the sensing readings of more MDOs would take more time.

Figure 8 illustrates the results of the average number of iterations needed in AKN2P2. The emulation utilizes the distance array $D = \{0, 1, \sqrt{2}, 2, \sqrt{5}, \sqrt{8}, 3, \sqrt{10}, \sqrt{13}, \cdots, 60\sqrt{2}\}$ with 1446 elements under the assumption of 50 MDOs in total. The ranges for the uniform distribution of x and y coordinates grow from $[0, 10]$ to $[0, 60]$, and the number of the nearest neighbors grows from 1 to 45. The emulation derives the average number of iterations by running 10 times for each specified k and (x, y), which are randomly selected from each specified coordinate range. It can be learned that the average number of iterations needed gradually decreases as the range for the uniform distribution of the coordinate distance grows from $[0, 10]$ to

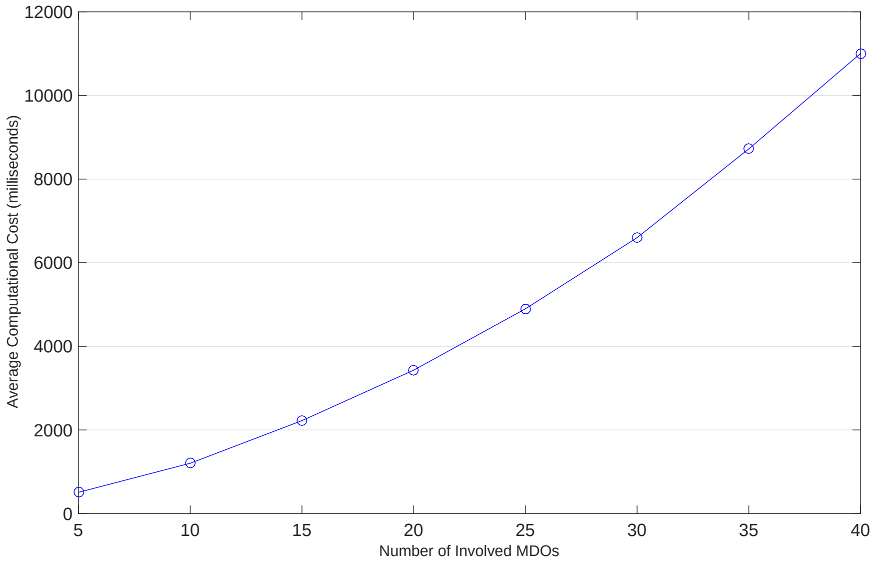


Figure 7: The Average Computational Cost Regarding Different Number of Involved MDOs

$[0, 60]$, because the density of MDOs would be reduced given the fixed number of $MDOs$, and it is easy for the divide-and-conquer approach to identify the k-nearest neighbors. Given a designated range of the uniform distribution of coordinate distance and a fixed number of $MDOs$, the average number of iterations fluctuates around a certain value due to the $O(log(n))$ complexity of the divide-and-conquer approach.

7. Related Work

The privacy preservation issues have been extensively studied in sensing scenarios. Information access control techniques [22, 38] require a trusted middle-ware service lying between location-based applications and the mobile users to enforce access control over geo-spatial data by rule-based access policies. Specifically, LocServ [22] answers queries of three types: i) requests for user location identified by the users' unique identifiers; ii) enumeration requests to return lists of users at specific locations; iii) asynchronous requests to notify events when users enter or leave areas of interest. The enforcement mechanism in [38] consists of a spatio-temporal module, an encoder and

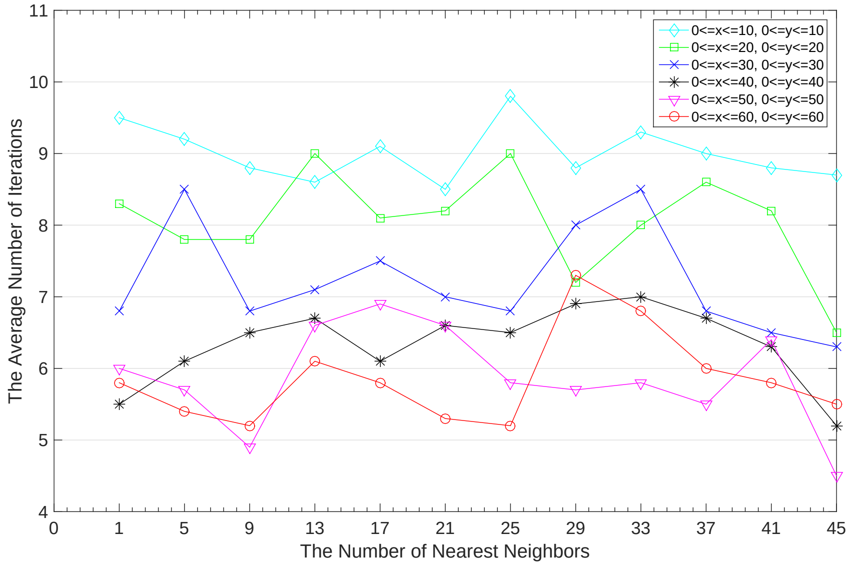


Figure 8: The Average Number of Iterations Regarding Different Number of Nearest Neighbors

the ASM-trie, and it follows a hierarchical access control model to enable adaptive search and support positive and negative location-based data access. While the techniques stated above depend on the third party to enforce the access control over the location-based data, they are vulnerable to the malicious behaviors of the third party.

Mix-zones [3, 27] also utilize middle-ware between the mobile users and the location-based application such that the location-based applications receive and reply pseudonymous messages from the mobile users. Specifically, the middle-ware assures the unlinkability of their pseudonyms by assigning a new, unused pseudonym to the mobile users when they enter a mix zone, and thus the location-based application cannot link the users emerging from the mix zone to the ones going into the mix zone. MobiMix [27], an application of mix-zones over road networks, develop a suite of construction methods to protect the location privacy of the mobile users. By the same token, it is subject to malicious behaviors of the third party middle-ware, as the middle-ware functions as a proxy to anonymize the locations of mobile users.

K-anonymity [32] ensures the information for each person in a release cannot be distinguished from at least $k-1$ individuals whose information are also in the same release. As a result, an attacker can identify a user based on the location information with probability no more than $1/k$. PRIVACYGRID [2] provides effective cloaking algorithms for location k-anonymity and l-diversity in a mobile environment wherein the mobile users communicate with the location-based service servers via location anonymization servers. Casper [21] uses the location anonymizer server to blur the users' exact information into cloaked spatial regions based on user-specified privacy requirements, and the privacy-aware query processor of the database only deal with the cloaked spatial areas instead of the exact location. In [9], a mobile user has to collaborate with $k-1$ peers to cloak her exact location into a spatial region before querying the location-based database server. All these k-anonymity schemes rely on the assumption that the third party is entrusted, which is usually infeasible in real-world settings.

Comparatively, the dummy location approaches [17, 31] eliminate the need of a third party server by generating redundant location-related data. In [17], a mobile user sends true location data with several dummy location data to a location-based service provider. The dummy locations are generated randomly and they are not real user locations as in the k-anonymity schemes. The privacy of the user location is protected as the location-based service provider cannot distinguish the true location from several dummy locations. SibilQuery [31] allows the user to generate $k-1$ Sybil queries to achieve k-anonymity, such that the location-based server is unable to distinguish between the user's real query and the Sybil queries. As the dummy location approaches rely on data redundancy rather than third-party anonymizer, they significantly increase the system overhead and complexity. At the same time, the location-based service provider can narrow down to the sub-space of the exact location, thereby resulting in a weak privacy.

Data transformation involves data owners, data users and clouds. The data owner uses certain encoding methodology to transform his data sets before outsourcing them to the cloud, while the data user attempts to retrieve the encrypted data sets with queries and perform decryption with the transformation keys derived from the data owner. The cloud is honest-but-curious about the encrypted data sets, and the cloud can perform search over the encrypted data sets based on the data user's encoded queries although they are unreadable. OPES [1] encrypts the data in an order-preserving manner to enable distance comparison operations. Wong *et al.* [34] allows kNN processing by using a secure point transformation to preserve the distances of points of interests relative to any query points. Khoshgozaran *et al.* [16] proposed to use Hilbert transformation to transform the points while the parameters (e.g., scale, order) keep secret. The data transformation techniques are vulnerable to access pattern attacks, and they are not scalable in real-world settings, because each data user has to get the transformation key from the data owner, thereby resulting in unbearable offline communication overhead.

Personal Information Retrieval (PIR) [25, 14, 15, 23, 25, 29, 30, 33] protocols allow a user to obtain the i-th record from the database server without disclosing which record he is obtaining. First, the user pins down his location index in the database via the location-based service provider without revealing his whereabouts. In this phase, Paulet *et al.* [30] utilize oblivious transfer while Ghinita *et al.* [15, 14] employ homomorphic encryptions. After that, the user relies on PIR protocols to retrieve points of interests associated with the index from the location-based service provider. Trusted hardware [29] can be used to generate the secret key and permute the database. Generally speaking, PIR protocols are secure against access pattern attacks, whereas they are too costly to be applied in real-world settings.

At the same time, l-diversity and t-closeness has been taken into consideration in MCS to increase anonymity and enhance privacy of the MDOs who contribute data. In k-anonymity, any quasi-identifier present in the released data set must appear in at least k records. Nevertheless, k-anonymity does not protect privacy when sensitive values in an equivalence class lack diversity or the attacker has some background knowledge. Hence, it is necessary to diversify sensitive attributes within each quasi-identifier equivalence class to achieve l-diversity, such that each equivalence class has at least l well-represented sensitive class. But the sensitive values are usually not evenly distributed, so l-diversity does not prevent probabilistic inference attacks. Therefore, the distribution of sensitive attributes in each quasi-identifier group should be "close" to the entire data set to achieve t-closeness. As a result, the complexity of data preprocessing is significantly raised while the data utility is seriously reduced due to masked data.

Pseudonym-based methods help offer anonymity to the MDOs, but they also bring significant cost and potential risk. A user uses a pseudonym for a while, then discards it and switch to a new one. Short-lived and frequently-changed pseudonyms bring better privacy. As a result, the user is obliged to maintain a repository of pseudonyms certified by trusted third-parties, thereby resulting in both higher cost and potential privacy leakage due to the compromised third-parties. Freudiger *et al.* [12] study the utilization of pseudonyms in a network consisting of autonomous mobile nodes, and these mobile nodes have to register with a central authority to obtain a new set of pseudonyms when the old ones expire. The critical conditions for the emergence of location privacy are obtained by analyzing the pseudonym ages, and it is suggested for a mobile node to cooperate with other nodes in proximity for pseudonym change. CPN[28] presents the Cooperative Pseudonym change scheme based on the number of Neighbors as a general cooperative framework. A node should have at least k neighbors to trigger the pseudonym exchange with its neighbors, and more neighbors provide better anonymity. The corresponding anonymity is analyzed and the simulation results indicate that CPN performs better than the non-cooperative scheme. Nevertheless, the privacy would be breached if any of the neighbors involved in pseudonym exchange are attackers. In addition, complicated procedures have to be designed to provide unlinkability of pseudonyms and preserve anonymity, thereby significantly increasing system overhead.

The group signature [7, 5] allows the verifier to validate that a message signature comes from a member in a specific group without knowing the identity of the signer. Nonetheless, it brings about significant system overhead, because the group setup requires the prior cooperation with other group members. In addition, the group manager is able to identify any group member, thereby putting the privacy of the group members at risk when the group manager is compromised. Comparatively, all of our schemes fit an ad-hoc network settings and they do not depend on any trusted third-parties. In addition, the group signature schemes attempt to hide identities and only disclose group membership information, and thus make fine-grained participant selection difficult, while our schemes hide the message content but disclose user identities without causing difficulty for participant selection.

Data perturbation is useful in reducing the risk of compromising privacy, the MDOs and the SSCs tend to submit perturbed sensing data and queries with generalized context to the CPs and the SSPs respectively. Consequently, the system becomes less efficient and obtain reduced utility, which represents the usefulness of sensing tasks, because the CPs may have to task a larger pool of participants and the SSPs need to conduct more computation to reach a certainty like a non-private process. It is self-evident that the two goals are hard to be optimized at the same time. Without a detailed knowledge of the context and raw data, it is hard to select the best participants and filter out noise and corrupt data to obtain the maximal utility. Our system should optimize the MCS processes to maximize the expected utility while subject to privacy concerns concurrently.

Homomorphic Encryptions provide an important solution to privacy preservation for multiple-party data computation and data sharing in sensing scenarios. Lin *et al.* [18] address the issue of securing two-parties' data comparison in a privacy preserving manner by exploiting ElGamal encryption, Paillier encryption, 0-Encoding and 1-Encoding. Erkin *et al.* [18] explore relatively efficient cryptographic privacy techniques based on Paillier cryptosystem to allow spatial and temporal aggregation of smart meter measurements. Privacy-preserving face recognition is also studied in [11] and extensive experiments are done by running the standard Eigenfaces recognition algorithm. Bilogrevic *et al.* [4] propose two privacy-preserving algorithms for the fair render-vous point problem with transformation functions based on homomorphic encryption for location-based services. However, the distances to certain users are exposed to multiple mobile users, and these mobile users can collude together to pinpoint the exact coordinates of the target victim. Ghinita *et al.* [13] first address the issue of private comparison of two integers, and accordingly proposed approximate and exact hybrid algorithms for private nearest-neighbor queries based on multi-level index infrastructure and Voronoi cells. Nonetheless, the two algorithms rely on the trusted location server to partition the two-dimensional plane into small convex polygon areas first. Yi *et al.* [36] study how to preserve the privacy of k-nearest neighbor queries from the semi-honest location-based service provider based on Paillier cryptosystem. All the location data info have to be revealed to the location-based service provider. The whole area is divided into small regions, and the location-based service provider returns a cluster of cells with points of interest that could be more than k. Zhang *et al.* [39] study fine-grained profile matching in proximity-based mobile social networking based on Paillier crypotosystems, Choi *et al.* [8] address the issue of secure mutual proximity zone enclosure evaluation with homomorphic encryption and order-preserving encryption. Specifically, it proposes two protocols, such that a client can securely determine her location is enclosed in the proximity zone of a target, and the client learns if the target's location is within the client's proximity zone, respectively. CityWatch[20] provides an urban-scale data sensing and dissemination framework through participatory sensing by taking the user-led design process with city stakeholders. It first presents two design processes: FloodWatch and GreenWatch. FloodWatch aims to address real-time data collection for disruptive events. It focuses on understanding the city stakeholders experiences and insights regarding the floods in 2011, the required types of data from sensing participation, the operational and decision making hierarchies, etc. FloodWatch finds out the citizens are unwilling to install applications for emergency events as they rarely happen. GreenWatch attempts to motivate citizens to use the application in the daily life for environmental purposes. It investigates the utilization of the crowd and the existing fixed sensors to enhance green initiatives and reduce wasteful practices. It finds out the city stakeholders are more willing to get involved with daily sustainable behavior. Accordingly, CityWatch comes out with a framework with salient features to address some design challenges such as data storage, scalability, security and so on. This framework consists of the CityWatch middleware and the CityWatch application server. The CityWatch middleware is an intermediate layer between data providers and consumers, and the CityWatch application server provides further functionalities, such as user management, report handling, reputation tracking and so on. The implementation is done based on the ongoing user trial on top of the CityWatch framework, and the result demonstrates that the user participation follows power-law based distribution as a few users have the most interactions with the application. NoiseTubePrime[10] provided a simple but effective distributed computation algorithm to generate grid-based maps for a target area and timeframe to protect the users' privacy. Each user participates in the sensing campaigns through a personal cloud software agent to preserve privacy and perform computation. As such, the users data are always online to be collected by different campaign contributors, and the issues of limited local computation resources and intermittent data connectivity are resolved by the resource-abundant cloud, thereby bringing high availability, scalability, ease of deployment and privacy. It expects the honest users traversing a specific fixed grid location to add its own measurement to the encrypted sum of measurements and increase the encrypted number of measurements, such that the average value can be derived by division when decrypted. In addition, it has implemented an online demo and undertook computational performance evaluation to demonstrate its feasibility. Comparatively, all the schemes in our paper encompass the ad-hoc network setting and they do not rely on any trusted third-parties. The MDOs are mobile without any geographical constraints, and their sensing readings are not associated with any fixed locations. Additionally, PPS aims to protect the privacy of the sensing service request, and PPDRC focuses on the individual data privacy of MDOs, while AKN2P2 protects both the privacy of the sensing service query and the location privacy of MDOs.

8. Conclusion

In this paper, we studied various ways to protect the privacy of the SSC and the MDOs in different MCS scenarios. We first proposed PPSQ to secure the privacy of the summation query of the SSC regarding the real target MDOs. Next, we put forward PPDRC to protect the data privacy of the MDOs to retrieve the difference rank in the multi-party computation process. As the signs of polynomials can be derived without disclosing the numeric values of the data records of the involved MDOs, given a baseline value d, the difference between the sensing reading of any MDO and d can be compared against a designated proportion of the summation of sensing reading differences of a group of MDOs without revealing the values of d and any sensing readings. Subsequently, we elaborated how to identify K-nearest neighbors around the POO issued by the SSC with minimum error, while keeping the POO and the locations of all involved MDOs secrete. In this solution, all the possible distance values between the PPO and each MDO location are included in a distance range, and all the MDOs agree on the privacy window of the smallest size such that the privacy level regarding the distances of some MDOs is still acceptable even in the worst case. Security analysis are given and the performance evaluations are done at the end.

References

[1] Agrawal, R., Kiernan, J., Srikant, R., Xu, Y., 2004. Order preserving encryption for numeric data. In: Proceedings of the 2004 ACM SIGMOD international conference on Management of data. ACM, pp. 563–574.

[2] Bamba, B., Liu, L., Pesti, P., Wang, T., 2008. Supporting anonymous location queries in mobile environments with privacygrid. In: Proceedings of the 17th international conference on World Wide Web. ACM, pp. 237–246.

[3] Beresford, A. R., Stajano, F., 2003. Location privacy in pervasive computing. IEEE Pervasive computing 2 (1), 46–55.

[4] Bilogrevic, I., Jadliwala, M., Kalkan, K., Hubaux, J.-P., Aad, I., 2011. Privacy in mobile computing for location-sharing-based services. In: Privacy Enhancing Technologies. Springer, pp. 77–96.

[5] Boneh, D., Shacham, H., 2004. Group signatures with verifier-local revocation. In: Proceedings of the 11th ACM conference on Computer and communications security. ACM, pp. 168–177.

[6] Chase, M., Chow, S. S., 2009. Improving privacy and security in multi-authority attribute-based encryption. In: Proceedings of the 16th ACM conference on Computer and communications security. ACM, pp. 121–130.

[7] Chaum, D., Van Heyst, E., 1991. Group signatures. In: Workshop on the Theory and Application of of Cryptographic Techniques. Springer, pp. 257–265.

[8] Choi, S., Ghinita, G., Bertino, E., 2014. Secure mutual proximity zone enclosure evaluation. In: Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. ACM, pp. 133–142.

[9] Chow, C.-Y., Mokbel, M. F., Liu, X., 2006. A peer-to-peer spatial cloaking algorithm for anonymous location-based service. In: Proceedings of the 14th annual ACM international symposium on Advances in geographic information systems. ACM, pp. 171–178.

[10] Drosatos, G., Efraimidis, P. S., Athanasiadis, I. N., Stevens, M., DHondt, E., 2014. Privacy-preserving computation of participatory noise maps in the cloud. Journal of Systems and Software 92, 170–183.

[11] Erkin, Z., Franz, M., Guajardo, J., Katzenbeisser, S., Lagendijk, I., Toft, T., 2009. Privacy-preserving face recognition. In: Privacy Enhancing Technologies. Springer, pp. 235–253.

[12] Freudiger, J., Manshaei, M. H., Le Boudec, J.-Y., Hubaux, J.-P., 2010. On the age of pseudonyms in mobile ad hoc networks. In: INFOCOM, 2010 Proceedings IEEE. IEEE, pp. 1–9.

[13] Ghinita, G., Kalnis, P., Kantarcioglu, M., Bertino, E., 2011. Approximate and exact hybrid algorithms for private nearest-neighbor queries with database protection. GeoInformatica 15 (4), 699–726.

[14] Ghinita, G., Kalnis, P., Khoshgozaran, A., Shahabi, C., Tan, K.-L., 2008. Private queries in location based services: anonymizers are not necessary. In: Proceedings of the 2008 ACM SIGMOD international conference on Management of data. ACM, pp. 121–132.

[15] Ghinita, G., Kalnis, P., Skiadopoulos, S., 2007. Prive: anonymous location-based queries in distributed mobile systems. In: Proceedings of the 16th international conference on World Wide Web. ACM, pp. 371–380.

[16] Khoshgozaran, A., Shahabi, C., 2007. Blind evaluation of nearest neighbor queries using space transformation to preserve location privacy. In: Advances in Spatial and Temporal Databases. Springer, pp. 239–257.

[17] Kido, H., Yanagisawa, Y., Satoh, T., 2005. An anonymous communication technique using dummies for location-based services. In: Pervasive Services, 2005. ICPS'05. Proceedings. International Conference on. IEEE, pp. 88–97.

[18] Lin, H.-Y., Tzeng, W.-G., 2005. An efficient solution to the millionaires problem based on homomorphic encryption. In: Applied Cryptography and Network Security. Springer, pp. 456–466.

[19] Liu, K., ???? Paillier cryptosystem.
URL `http://www.csee.umbc.edu/ kunliu1/research/Paillier.html`

[20] Manzoor, A., Patsakis, C., Morris, A., McCarthy, J., Mullarkey, G., Pham, H., Clarke, S., Cahill, V., Bouroche, M., 2014. Citywatch: exploiting sensor data to manage cities better. Transactions on Emerging Telecommunications Technologies 25 (1), 64–80.

[21] Mokbel, M. F., Chow, C.-Y., Aref, W. G., 2006. The new casper: query processing for location services without compromising privacy. In: Proceedings of the 32nd international conference on Very large data bases. VLDB Endowment, pp. 763–774.

[22] Myles, G., Friday, A., Davies, N., 2003. Preserving privacy in environments with location-based applications. IEEE Pervasive Computing 2 (1), 56–64.

[23] Naor, M., Pinkas, B., 1999. Oblivious transfer with adaptive queries. In: Advances in CryptologyCRYPTO99. Springer, pp. 573–590.

[24] Naor, M., Pinkas, B., Reingold, O., 1999. Distributed pseudo-random functions and kdcs. In: International Conference on the Theory and Applications of Cryptographic Techniques. Springer, pp. 327–346.

[25] Ostrovsky, R., Skeith III, W. E., 2007. A survey of single-database private information retrieval: Techniques and applications. In: Public Key Cryptography–PKC 2007. Springer, pp. 393–411.

[26] Paillier, P., 1999. Public-key cryptosystems based on composite degree residuosity classes. In: Advances in cryptologyEUROCRYPT99. Springer, pp. 223–238.

[27] Palanisamy, B., Liu, L., 2011. Mobimix: Protecting location privacy with mix-zones over road networks. In: Data Engineering (ICDE), 2011 IEEE 27th International Conference on. IEEE, pp. 494–505.

[28] Pan, Y., Li, J., 2013. Cooperative pseudonym change scheme based on the number of neighbors in vanets. Journal of Network and Computer Applications 36 (6), 1599–1609.

[29] Papadopoulos, S., Bakiras, S., Papadias, D., 2010. Nearest neighbor search with strong location privacy. Proceedings of the VLDB Endowment 3 (1-2), 619–629.

[30] Paulet, R., Kaosar, M. G., Yi, X., Bertino, E., 2014. Privacy-preserving and content-protecting location based queries. Knowledge and Data Engineering, IEEE Transactions on 26 (5), 1200–1210.

[31] Shankar, P., Ganapathy, V., Iftode, L., 2009. Privately querying location-based services with sybilquery. In: Proceedings of the 11th international conference on Ubiquitous computing. ACM, pp. 31–40.

[32] Sweeney, L., 2002. k-anonymity: A model for protecting privacy. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 10 (05), 557–570.

[33] Wang, S., Ding, X., Deng, R. H., Bao, F., 2006. Private information retrieval using trusted hardware. In: Computer Security–ESORICS 2006. Springer, pp. 49–64.

[34] Wong, W. K., Cheung, D. W.-l., Kao, B., Mamoulis, N., 2009. Secure knn computation on encrypted databases. In: Proceedings of the 2009 ACM SIGMOD International Conference on Management of data. ACM, pp. 139–152.

[35] Yao, A. C.-C., 1982. Protocols for secure computations. In: FOCS. Vol. 82. pp. 160–164.

[36] Yi, X., Paulet, R., Bertino, E., Varadharajan, V., 2014. Practical k nearest neighbor queries with location privacy. In: Data Engineering (ICDE), 2014 IEEE 30th International Conference on. IEEE, pp. 640–651.

[37] Yiu, M. L., Jensen, C. S., Huang, X., Lu, H., 2008. Spacetwist: Managing the trade-offs among location privacy, query performance, and query accuracy in mobile services. In: Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on. IEEE, pp. 366–375.

[38] Youssef, M., Atluri, V., Adam, N. R., 2005. Preserving mobile customer privacy: an access control system for moving objects and customer profiles. In: Proceedings of the 6th international conference on Mobile data management. ACM, pp. 67–76.

[39] Zhang, R., Zhang, R., Sun, J., Yan, U., 2012. Fine-grained private matching for proximity-based mobile social networking. In: INFOCOM, 2012 Proceedings IEEE. IEEE, pp. 1969–1977.

Zhijie Wang received his B.S. and M.S. degree from Beijing University of Posts & Telecommunications, in 2007 and 2010, respectively. He is a Ph.D. at Arizona State University. His research interests include wireless networking, applied cryptography, network security and cloud computing.

Dijiang Huang (Member 2000, Senior Member 2011) received his B.S. degree from Beijing University of Posts & Telecommunications, China 1995. He received his M.S., and Ph.D. degrees from the University of Missouri-Kansas City, in 2001 and 2004, respectively. He is an Associate Professor in the School of Computing Informatics and Decision System Engineering at the Arizona State University. His current research interests are computer networking, security, and privacy. He is an associate editor of the Journal of Network and System Management(JNSM) and an editor of IEEE Communications Surveys & Tutorials. He has served as an organizer for many International conferences and workshops. Dr. Huang's research is supported by NSF, ONR, ARO, NATO, and Consortium of Embedded System (CES). He is a recipient of ONR Young Investigator Program (YIP) Award.