Research Article

PDA: A Novel Privacy-Preserving Robust Data Aggregation Scheme in People-Centric Sensing System

Ziling Wei, Baokang Zhao, and Jinshu Su

School of Computer Science, National University of Defense Technology, Changsha, Hunan, China

Correspondence should be addressed to Baokang Zhao; bkzhao@nudt.edu.cn

Received 27 June 2013; Accepted 4 October 2013

Academic Editor: Qin Xin

Copyright © 2013 Ziling Wei et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the advancement of wireless communication technologies, mobile phones, PDAs, and car embedded devices are equipped with sensors, such as sound and image. People can apply these devices to form a new sensing network called people-centric sensing network. And this network offers new opportunities for cooperative sensing applications. However, it introduces some challenges, including security challenge and robust challenge. As sensor nodes need to send their individual sensed data to an aggregator node and these data are related to users’ real life, privacy-preserving data aggregation is a challenge issue. As a node could become offline or a message could be lost before reaching the aggregator, retaining the correctness of the aggregate computed is important. In this paper, we present the design of PDA, a novel privacy-preserving robust data aggregation scheme in people-centric sensing system. Based on $K$-anonymity, homomorphic encryption, and secret sharing, PDA can support a wide range of statistical additive and non-additive aggregation functions such as Sum, Subtraction, Average, Count, Max/Min, and Median without leaking individual sensed data. Moreover, PDA is robust to node failure and data loss. We also evaluate the efficacy and efficiency of PDA. The result shows that our scheme can achieve the security and robust goal under a reasonable cost.

1. Introduction

Nowadays, technological advances in sensing, computation, storage, and wireless communications turn the mobile devices carried by people into a global mobile sensing device [1]. For example, currently smartphone is equipped with accelerometers, audio, location, and image sensors. So the mobile device’s owner becomes sensor custodian that can use the device to collect meaningful context data [2]. Therefore, people as individuals or special interest groups can apply the new sensing devices to form sensing networks which are called people-centric sensing networks [3]. This transformation provides a chance to create intelligence systems that collect data from widespread public participations [4]. There are many systems in present, such as BikeNet [5], CitySense [6], Mobiscopes [7], Urban Sensing [8], SenseWeb [9], and CarTel [10]. They could be used to monitor environmental pollution, temperature, or noise intensity of urban areas. In order to declare the impact of people-centric sensing networks, we will introduce the CitySense. CitySense passively “senses” the most popular places based on actual real-time activity and displays a live heat map. The application intelligently leverages the inherent wisdom of crowds without any change in existing user behavior, in order to navigate people to the hottest spots in a city [6]. So we could find the most popular places to visit easily through CitySense. That is, people-centric sensing networks have a great potential to provide large-scale, flexible, and global sensing network services.

Even though people-centric sensing networks can offer many benefits to users, there are still some challenges. One of them is that the mobile device collects context information closely related to user’s real life. So the concern of user privacy is a challenge for people-centric sensing networks. As users usually do not gain a direct benefit from reporting data, users are unwilling to share their data if their privacy is at risk [11]. The procedure of data aggregation is the most likely to be attacked by adversary. The other challenge is that a node could become offline or a message could be lost before reaching the aggregator. So we should consider a solution that ensures that these problems could not affect the correctness of
the aggregate computed. These challenges motivate our design of PDA, a novel privacy-preserving robust data aggregation scheme in people-centric sensing system.

Although the people-centric sensing network is becoming popular, there is relatively little work focusing on its privacy aspects and robust to node or communication failure. The shortage of prior work can be concluded in three points. The first point is that prior work has generally focused on preserving user’s privacy to a certain extent [11–14]. However, these researches are not robust to node or communication failure. The second point is that these researches rarely address the privacy issues of data aggregation which is the most likely to be attacked by adversary [15–17]. The third point is that most of prior work did not suit for the peculiarity of this network. For example, the devices are no longer owned by a single authority but belong to individuals; therefore, the devices could be mobile and malicious.

Though the above analysis, we could know that none of the existing schemes could overcome the challenges that one mentioned above in people-centric sensing networks. The contributions of this paper are summarized as follows.

(i) We formulate the threat model in people-centric sensing networks. These models enable researchers to study the privacy-preserving in people-centric sensing networks conveniently.

(ii) We present a novel scheme, PDA, to protect users’ privacy against threats and be robust to node or communication failure in people-centric sensing networks. This scheme is based on K-anonymity, homomorphic encryption, and secret sharing. And it can support a wide range of statistical additive and non-additive aggregation functions such as Subtraction, Sum, Average, Count, Max/Min, and Median.

The paper is organized as follows. We present the related work in Section 2. In Section 3, we introduce the basic of K-anonymity, homomorphic encryption, and secret sharing. We present the system architecture, threat model, and design goal in Section 4. We present our scheme in detail in Section 5. Then, we give the evaluation of PDA in Section 6. Conclusion could be found in Section 7.

2. Related Work

Although the people-centric sensing network is becoming popular, there is relatively little work focusing on its privacy aspects and robust to node or communication failure.

In [11], AnonySense architecture for anonymous tasking and reporting has been proposed. Shi et al. [12] present the PriSense scheme to protect privacy during the aggregation process of data. And this scheme support additive and non-additive aggregation functions. Feng et al. [13] present a scheme to realize the data aggregation. However, it only supports additive aggregation functions such as max/min and median at the sacrifice in data accuracy. Zhang et al. [14] also present a scheme which supports both additive aggregation functions and nonadditive ones. These researches can preserve users’ privacy to a certain extent. However, these researches are not robust to node or communication failure.

Wagner [15] present a scheme that protect the data aggregation in the presence of false data injection attack by a few malicious nodes and provide guidelines for selecting appropriate aggregation functions in a sensor network. Chan et al. [16] design an algorithm which the base station could detect if the computed aggregate was falsified. Conti et al. [17] present a scheme to preserve the privacy of robust data aggregation in wireless sensor network. However, these researches did not address the privacy issues of data aggregation which is the most likely to be attacked by adversary. In addition, as the devices are no longer owned by a single authority but belong to individuals in people-centric sensing network, the devices could be mobile and malicious. Therefore, these researches are not suitable for people-centric sensing applications.

Therefore, the novelty and the main differences with the existing work in the literature can be concluded in four points.

(i) PDA is looking at the new sensing network called people-centric sensing network. Most of the existing schemes cannot suit for the peculiarity of this network. For example, the devices are no longer owned by a single authority but belong to individuals; therefore, the devices could be mobile and malicious.

(ii) The problem that PDA solved contains security challenge and robust challenge. To the best of my knowledge, PDA is the first scheme that can solve these two problems of people-centric sensing network. Moreover, we not only consider the external internal threat, but also the internal threat. This is out of reach for a lot of existing work.

(iii) PDA is a clever use of K-anonymity, homomorphic encryption, and secret sharing. There is little research to use these schemes for robust challenge. Moreover, the combination of these schemes works very well.

3. Basic Schemes

In this section, we will introduce the basic schemes which are implemented in our scheme.

3.1. K-Anonymity. Society is experiencing exponential growth in the number and variety of data collections containing person-specific information as computer technology, network connectivity, and disk storage space become increasingly affordable. But the security of the database is a big challenge. Some researches propose that we can get a certain protection by removing some sensitive message. However, in most of these cases, the remaining data can be used to reidentify individuals by linking or matching the data to other data or by looking at unique characteristics found in the released data.

In order to solve this problem, Samarati [18] propose K-anonymity in 1998. In K-anonymity, an attacker cannot distinguish the specific individual who belongs to the private information by adding a certain amount of fake individuals. Therefore, we can prevent the disclosure of personal privacy.
3.2. Homomorphic Encryption. Homomorphic encryption technology was originally developed in 1978 by Rivest [20]. It is an encryption transformation technology that allows operation of the ciphertext. Homomorphic encryption was first used to encrypt the statistics. It is ensured that the user can operate on sensitive data by the homogeneity of the algorithm without revealing the data information. That is, homomorphic encryption is a form of encryption which allows specific types of computations to be carried out on ciphertext and obtain an encrypted result which matches the decrypted result of operations performed on the plaintext. A cryptosystem which supports both addition and multiplication is known as fully homomorphic encryption. Using such a scheme, any circuit can be homomorphically evaluated, effectively allowing the construction of programs which may be run on encryptions of their inputs to produce an encryption of their output. This is equivalent to not knowing that the problem also gives the answer to the problem.

Homomorphic encryption is built on the foundations of algebra theory. Then we will introduce the basic idea of homomorphic encryption. Assuming that $E$ and $D$ are the encryption and decryption functions, plaintext data is a finite set $M = \{m_1, m_2, \ldots, m_n\}$, $\alpha$ and $\beta$ on behalf of computing. If $D(\alpha(E(m_1), E(m_2), \ldots, E(m_n))) = \beta(m_1, m_2, \ldots, m_n)$ is right, $(E, D, \alpha, \beta)$ is called homomorphism. Homomorphic encryption has attracted many attentions of many scholars in recent years, such as Yu. Yu proposed homomorphic encryption algorithm; Craig Gentry introduced an algorithm that $\alpha$ and $\beta$ are the same operation, and he referred to as the “ideal lattice” mathematical objects that allow people to make operation of the encryption data [21, 22].

3.3. Secret Sharing. Secret sharing distributes, preserves, and restores the secret method. And it is an important tool to achieve secure multiparty computation. Secret sharing refers to method for distributing a secret among a group of participants, each of whom is allocated a share of the secret. The secret can be reconstructed only when a sufficient number, of possibly different types, of shares are combined together; individual shares are of no use on their own [23]. Secret sharing technology extends to many areas and practice systems, such as video, voice, and other fields. It is a technology which combines theory and practice.

The general type of secret sharing is threshold method, generally with $(m, n)$-threshold method. In this type, there are one dealer and $n$ participants $(P_1, P_2, \ldots, P_n)$. The dealer slices the secret $S$ to $(s_1, s_2, \ldots, s_n)$ and shares the slices to the participants. The dealer accomplishes this by giving each participant a share in such a way that any group of $m$ or more participants can together reconstruct the secret, but no group of fewer than $m$ players can. That is, it takes $m$ points to define a polynomial of degree $m - 1$. The method is to create a polynomial of degree $m - 1$ with the secret as the first coefficient and the remaining coefficients picked at random. Next find $n$ points on the curve and give one to each of the players. When at least $m$ out of the $n$ players reveal their points, there is sufficient information to fit a $(m - 1)$th degree polynomial to them, the first coefficient being the secret [24, 25].

The basic method of secret sharing contains Shamir secret sharing method, Blakley secret sharing method, Karin-Greene-Hellman secret sharing method and so on.

4. System Architecture

In this section, we present the system model, threat model, and design goal.

4.1. System Model. The architecture of people-centric sensing system is demonstrated in Figure 1.
4.2. Threat Model. In people-centric sensing networks, the data which is received from MNs unavoidably relates to a lot of personal information. If the information is known by an adversary, the user could suffer many troubles. In this section, we will analyze the threat model. As the procedure of data aggregation is the most likely to be attacked by an adversary, we are mainly concerned with the threat of this procedure. We separate the threats into two classes: the internal threat and the external threat.

(1) The Internal Threat. The main threat of internal attackers is that the ASs and MNs could be malicious. That is to say, the ASs and MNs could be compromised and controlled by adversaries. This model is rational in that the number of ASs and MNs is enormous and the ASs and MNs are owned by third-parties. Therefore, it is hard to guarantee that the entire ASs and MNs are reliable. The malicious ASs and MNs could reveal the privacy easily if we take no measures.

(2) The External Threat. As the internet is an open channel, the communication link is unsafe and unreliable. The external threat mainly contains two parts.

Part 1. The adversary may eavesdrop on communications between MNs and AS.

Part 2. The adversary may attempt to insert some false data to the procedure of data aggregation. That is, the adversary may damage the data integrity.

4.3. Design Goal. Our design goal in our paper is satisfying the following requirements.

(i) As the aggregation server and nodes could be compromised and controlled by adversaries, we should prevent the sensed data of an individual node from being disclosed to the aggregation server and nodes.

(ii) As the communication link is unsafe and unreliable, we should prevent the sensed data being leaked. Moreover, we must guarantee the integrity of sensed data.

(iii) As the nodes could become offline or a message could be lost before reaching the aggregation server, we should ensure that these problems could not affect the correctness of the aggregate computed.

5. Scheme Design

In this section, we describe our PDA scheme in detail. As we focus on the privacy-preserving robust data aggregation of people-centric sensing networks, we should support a wide range of statistical additive and nonadditive aggregation functions safely. The additive and nonadditive aggregation functions include Sum, Subtraction, Average, Count, Max/Min and Median. We will use Sum as an example to declare the scheme’s realization. That is, we should prevent the sensed data of an individual node from being disclosed to the AS and other MNs during the process of computing. Moreover, we should be robust to node or communication failure. We could get the issue model as follow.

Issue Model. There are n MNs \(\{P_1, P_2, \ldots, P_n\}\). Each of them has a sensed data \(x_i, i \in [1, n]\). These data should be computed (such as Sum). And the result should be sent to the AS. However, these MNs do not want to leak anything about their data to the other MNs and AS. Through the analysis of the problem, we know that the above problem can be solved as long as we can ensure the security of two-party computing problem. Therefore, the above model can be simplified as follow. Suppose that there are two MNs (say, Alice and Bob); each of them has a sensed data (say, data_Alice and data_Bob). And one or more MNs (called third-party) \(M = \{M_1, M_2, \ldots, M_r\}\) exist in the model. And these third-parties are untrusted. Alice and Bob send their sensed data to the AS though these third-parties. Then the AS computes the sum of \(A\) and \(B\). At the same time, \(A\) and \(B\) could not be disclosed to the AS and the third-parties. Moreover, we should be robust to node or communication failure. At last the result should be sent to the server.

In order to solve the issue, we present a novel scheme, PDA. The whole idea of scheme is shown in Figure 2. In order to understand the scheme intuitional, we will introduce an instance. In the instance, Alice and Bob have sensed data for computing. We assume \(\text{data}_\text{Alice} = 3\), \(\text{data}_\text{Bob} = 4\).

There are five steps to realize our scheme.
5.1. Service Registration. Before the aggregation process of data, each MN needs to register for the service at our scheme. The service registration contains initialization and MN’s registration.

During the initialization, the server should periodically generate fake IDs for K-anonymity and store them in a fake ID pool. The fake IDs can be generated using a cryptographic hash function, such as SHA-1. The equation is given as follows: fake ID = SHA(fake ID_{i-1} ⊕ salt).

During MN’s registration, the MN should send a request to the server. After receiving the request, the server verifies if the MN is legal or not. After authentication, the server should pick k fake IDs from its fake ID pool. And one of them is used to replace the identity of the MN. Let this fake ID be RID. The other k – 1 fake IDs are used to confuse the adversaries. Therefore, the server stores an entry as (RID, ID_1, ID_2, ..., ID_{k-1}) in a fake ID table. The number of fake IDs (k) depends on the MN’s reporting frequency. At last, the server sends an OK message which contains the fake ID entry to the MN. The function of fake ID is realizing the anonymous transmission which is based on K-anonymity.

5.2. Authentication. As the feature of people-centric sensing network, the MNs are always moving from an AS to another one. To let the AS authenticate their identity, the MNs should send an authentication request to the AS. The request contains the MN’s ID table. After receiving the authentication request, the AS forwards this request to the server. Upon receiving the message, the MS should verify the ID table’s legality. If the verification succeeds, it sends a reply to the AS. On the reception of the reply, the AS sends an OK message to the MN which contains some parameters. These parameters contain a prime number P for homomorphic encryption, a prime number Q, and a random number m for secret sharing. The meaning of these parameters will be recommended in later subsection. After authentication, the communication link is established between the MN and the AS. Then the MNs could send the sensed data to the AS after receiving a task.

In our instance, we take P = 11, Q = 19, and m = 3.

5.3. Homomorphic Encryption. After the MNs receive a task from the server, they should send the sensed data to the AS. Before sending data, the MNs will execute the data processing, homomorphic encryption. In this paper, the important point is proposing PDA, so we will use a simple algorithm to achieve homomorphic encryption. The implementation process is shown as follows. A large prime P getting from the AS is used as the key. a and b are sensed data. The encryption algorithm is A = a + (r * P), r is a random integer number, A is the ciphertext, and a is the plaintext. Therefore, the data of a and b which have been encrypted can be expressed as (A + B) = (a + (r1 * P)) + (b + (r2 * P)). Then, the process of decryption is P modulo (A + B) and the result is (a + b) [26, 27]. So we achieve a simple homomorphic encryption algorithm. In PDA, both participants should operate their own data using the homomorphic encryption algorithm before sending to the AS. Though the operation, the AS get the sensed data which was encrypted. However, it could compute the sum of the sensed data. That is, the aggregation functions (such as Sum) could be completed without leaking individual sensed data.

In our instance, Alice and Bob use homomorphic encryption algorithm to encrypt data_Alice and data_Bob according to the prime number P which is negotiated. Then, we could get the ciphertext. For Alice, we take an integer r = 3; ciphertext A is A = a + (r * P) = 3 + 3(11) = 36. For Bob, we take an integer r = 2; ciphertext B is B = 4 + (2 * 11) = 26.

5.4. Secret Sharing. As the MNs could become offline or a message could be lost before reaching the AS, we should ensure that these problems could not affect the correctness of the aggregate computed. In our paper, we could achieve this goal through secret sharing.

The general type of secret sharing is threshold method, generally with (m,n)-threshold method. In this type, the dealer accomplishes this by giving each participants a share in such a way that any group of m or more participants can together reconstruct the secret but no group of fewer than m players can. In this scheme, in order to send the data after encryption to some third-parties and process the data by the third-party, we will improve the Shamir secret sharing algorithm. Shamir secret sharing algorithm is proposed by Shamir in 1979 [28]. The system relies on the idea that you can fit a unique polynomial of degree m – 1 to any set of m points that lie on the polynomial. It takes two points to define a straight line, three points to fully define a quadratic, four points to define a cubic curve, and so on. That is, it takes m points to define a polynomial of degree m – 1. The principle of the algorithm is shown in Figure 3. Any two points in a straight line can define a straight line. So we can take any
of two points in \((x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\) to obtain the equation of the line [29].

The procedure of the algorithm is (assuming the secret is \(K\)) as follows.

Selecting a Polynomial. At first, we should select a polynomial \(F(x)\) based on a prime number \(Q\). The prime number is got from the parameter negotiation. \(F(x) = (a_{m-1}x^{m-1} + a_{m-2}x^{m-2} + \cdots + K) \mod Q\), and \(a_{m-1}, a_{m-2}, \ldots, a_1 \in [1, Q]\).

Getting the Variables. Then, we should select \(n\) groups data \(x_1, x_2, \ldots, x_n, x_i \neq 0\).

Computing the Point. At last, we should compute the point \((x_i, y_i)\), which meets the polynomial.

Therefore, we split the secret \(K\) to \((x_i, y_i), i \in [1, n]\). We can reconstruct the secret when getting more than \(m\) parts.

After homomorphic encryption, the MN will operate the data which was encrypted by homomorphic encryption through secret sharing. That is, the MN will slice the data to \(t\) points. Then, the MN will send these points to \(t\) third-parties. At the same time, the MN should pick an ID from its fake ID table and send it to the third-parties.

In our instance, Alice and Bob will perform secret sharing operation for the ciphertext. We assume that the polynomial \(F(x)\) for Alice is \(F(x) = (2x^2 + 4x + 36) \mod 19\). At first, Alice will compute the point of polynomial, based on \((x_1, x_2, x_3, x_4)\). Then, Alice could get four points \([1, 4, 2, 14, 3, 9, 4, 8]\). At last, Alice will send the four points to four third-parties. Similarly, we assume that the polynomial \(F(x)\) for Bob is \(F(x) = (x^2 + 2x + 26) \mod 19\). Bob will compute the point of polynomial, \([1, 10, 2, 15, 3, 3, 4, 12]\) [30]. At last, Bob will send the four points to four third-parties. At the same time, Alice and Bob should send its fake ID to the third-parties.

5.5. AS Operation. After the third-parties receive the points and the ID, they should forward the information to the AS. Upon receiving the points and the ID, the AS should take three steps to get the sum of the sensed data.

Step 1 (recovering the data). The AS picks the points which have the same ID as a group. And it recovers the data for each group based on secret sharing. The procedure of recovering is as follow. At first, the AS should pick \(m\) points and select a polynomial \(F(x) = (a_{m-1}x^{m-1} + a_{m-2}x^{m-2} + \cdots + K) \mod Q\). Then, the AS substitutes the \(m\) points into \(F(x)\). So we can get the data of \((a_{m-1}, a_{m-2}, \ldots, K)\). At last, we could get the data when setting \(x = 0\). In our instance, we pick three points \((1, 4, 2, 14, 3, 9)\) and substitute them into \(F(x)\) for Alice. Then, we could get that \(a_2 = 2, a_1 = 4, \text{ and } K = 36\). So the data is 36 for Alice. Similarly, the data is 26 for Bob.

Step 2 (computing the sum of data). After recovering the data, we could compute the sum of data. In our instance, the sum of data is \(R = 36 + 26 = 62\).

Step 3 (getting the result). As we conduct the data using homomorphic encryption, the data after computing is not the final result. In order to get the result, we should take the process of decryption which is using \(P\) modulo of the data. In our instance, the result is \(R \% P = 62 \% 11 = 7\). As data\_Alice + data\_Bob = 3 + 4 = 7, we know that the result is correct. At last, the result will be sent to the server. The server can provide data to some applications which provide guidance of the social environment or people's social activities.

Through the above analysis, we compute the sum of sensed data during the procedure of aggregation. Similarly, we could support a wide range of statistical additive and nonadditive aggregation functions [12].

6. Evaluation

In this section, we evaluate the security and performance of PDA.

6.1. Security Evaluation. In this part, we analyze the security of our scheme based on the threat model discussed in Section 3. We assume that the threats contain that (1) the ASs could be compromised by adversaries; (2) the MNs could be malicious; (3) the adversary could eavesdrop all of the communications in the network; (4) the adversary could attempt to insert some false data to the procedure of data aggregation.

For ease of threat, we explain the security features of our scheme.

(i) The ASs could be compromised by adversaries. In PDA, the AS only gets the sensed data which has been encrypted by homomorphic encryption algorithm. As the AS cannot get the parameter \(r\), it could not decrypt the data. Moreover, the AS cannot get the MNs which own the sensed data real ID as fake IDs. Therefore, the attacker obtains no useful information from the AS to compromise a single node’s privacy.
(ii) The MNs could be malicious. In PDA, the MN acts as one of third-part in the procedure of data aggregation. Each MN can only get one part of the other MNs' sensed data based on secret sharing. And fewer than \( m \) players cannot reconstruct the data. Moreover the sensed data is encrypted by homomorphic encryption algorithm. Therefore, we can compromise the privacy of a non-captured node leveraging the information.

(iii) The adversary could eavesdrop all of the communications in the network. In PDA, each link between AS and MN transfers only one part of sensed data based on secret sharing. And fewer than \( m \) players cannot reconstruct the data. If the MN eavesdropped all of the communications in the network, it can reconstruct the sensed data. However, the sensed data is encrypted by homomorphic encryption algorithm. As the adversary cannot get the parameter \( r \), it could not decrypt the data. Therefore, the adversary obtains no useful information to compromise a single node's privacy.

(iv) The adversary could attempt to insert some false data to the procedure of data aggregation. In PDA, each sensed data is attached with the owner's ID. And the ID is picked from the MN's fake ID table randomly. As the adversary cannot get the fake ID table, he does not know the valid ID. The false data which was sent by the adversary is not accepted by the AS. Therefore, the adversary cannot damage the data integrity.

Through the above analysis, PDA is able to meet the security requirements of privacy-preserving data aggregation.

### 6.2. Performance Evaluation

In this section, we focus on the efficiency of PDA from robustness, computational complexity, and communication complexity.

#### 6.2.1. Robustness

As the MNs could become offline or a message could be lost before reaching the AS, we should ensure that these problems could not affect the correctness of the aggregate computed. In our paper, we could achieve this goal though secret sharing. As the AS get a share from \( n \) third-parties, it can reconstruct the data but no group of fewer than \( m (m < n) \) can. That is, the AS can reconstruct the sensed data when it received no less than \( m \) parts. Therefore, if the number of node failure or data loss is less than \( n - m \), we could ensure the correctness of the aggregate computed. Though the analysis, the AS can robust to node or communication failure.

#### 6.2.2. Computational Complexity

For the people-centric sensing system, it is assumed that there are \( n \) MNs for computing. In order to compute all of the MNs' data, we should compute each two MNs' data. Therefore, the computing complexity of this process is \( O(n^2) \). For each computation processing, the computation complexity is mainly determined by homomorphic encryption and secret sharing. The operations of homomorphic encryption and secret sharing are the simple algebraic operations. So the computation complexity is \( O(k) \) that is not relevant to the size of input. The analysis shows that computational complexity of PDA is \( O(n^2) \).

#### 6.2.3. Communication Complexity

The primary aspect to measure the performance of a scheme is its computational complexity. But, for the problem of privacy-preserving robust data aggregation, computational complexity cannot fully describe the performance of a scheme. As the participants and third-parties will communicate each other, the communication complexity is also an important aspect to measure the performance of PDA.

The communication complexity contains the cost of secret sharing and the third-party operation. It is assumed that the number of MNs is \( m \) and the number of third-parties is \( n \). As the MNs will send the secret input data to the third-parties and the third-parties will send the result to the AS, the communication complexity of PDA is \( O(m \times n) \).

#### 6.3. Comparison

In this section, we summarize the features of our proposal compared with other mainly relevant algorithms in Table 2.

The feature additive and nonadditive aggregation indicates if the scheme support a wide range of statistical additive and nonadditive aggregation functions such as Sum, Subtraction, Average, Count, Max/Min, and Median. In columns 2, 3, and 4, these features indicate if the scheme supports privacy against outside eavesdropper, other MNs, or the AS. The features of node failure and data loss resilience refer to whether the AS can compute the correct aggregate when a few MNs become offline or a few messages could be lost. In columns 6 and 7, these features denote the scheme's computational and communication complexity. The last feature indicates if the scheme is suitable for the people-centric sensing networks.

Through the above analysis, our scheme is secure, scalable, and resilient to node failure and data loss.
7. Conclusion

We present PDA, a novel privacy-preserving robust data aggregation scheme in people-centric sensing system. Based on \(K\)-anonymity, homomorphic encryption, and secret sharing, PDA can support a wide range of statistical additive and nonadditive aggregation functions such as sum, subtraction, average, count, max/min and median without leaking individual sensed data. Moreover, PDA is robust to node failure and data loss. We also evaluate the efficacy and efficiency of PDA. The result shows that our scheme can achieve the security and robust goal under a reasonable cost. We believe a privacy-aware system will make people-centric sensing networks more acceptable.

Acknowledgments

The work described in this paper is partially supported by the Grants of the National Basic Research Program of China (973 project) under Grants nos. 2009CB320503, 2012CB315906; the project of National Science Foundation of China under Grants nos. 61070199, 6103189, 6103194, 6103182, 61202488, and 61272482; the National High Technology Research and Development Program of China (863 Program) nos. 2011AA01A103, 2012AA01A506, and 2013AA013505; the Research Fund for the Doctoral Program of Higher Education of China under Grants nos. 20114307110006, 20124307120032; the program for Changjiang Scholars and Innovative Research Team in University (no. IRT1012); Science and Technology Innovative Research Team in Higher Educational Institutions of Hunan Province (network technology); and Hunan Province Natural Science Foundation of China (11J7003).

References

[1] T. Abdelzaher, Y. Anokwa, P. Boda et al., "Mobiscopes for human spaces," IEEE Pervasive Computing, vol. 6, no. 2, pp. 20–29, 2007.
[2] A. T. Campbell, S. B. Eisenman, N. D. Lane, E. Miluzzo, and R. A. Peterson, “People-centric urban sensing,” in Proceedings of the 2nd Annual International Workshop on Wireless Internet (WICON ’06), ACM, 2006.
[3] A. T. Campbell, S. B. Eisenman, N. D. Lane et al., “The rise of people-centric sensing,” IEEE Internet Computing, vol. 12, no. 4, pp. 12–21, 2008.
[4] I. Krontiris, F. C. Freiling, and T. Dimitriou, “Location privacy in urban sensing networks: research challenges and directions,” IEEE Wireless Communications, vol. 17, no. 5, pp. 30–35, 2010.
[5] S. B. Eisenman, E. Miluzzo, N. D. Lane, R. A. Peterson, G.-S. Ahn, and A. T. Campbell, “BikeNet: a mobile sensing system for cyclist experience mapping,” ACM Transactions on Sensor Networks, vol. 6, no. 1, article 6, 2009.
[6] M. Welsh and J. Bers, “CitySense: an urban-scale sensor network,” in Ecological Urbanism, M. Mostafavi and G. Doherty, Eds., Harvard Graduate School of Design, pp. 164–165, Lars Müller, Zurich, Switzerland, 2010.
[7] Y. Malinovsky and Y. Wang, “Pedestrian travel pattern discovery using mobile bluetooth sensors,” in Proceedings of the 91st Annual Meeting of the Transportation Research Board, 2012.
[8] M. Mun, S. Reddy, K. Shilton et al., "PEIR, the personal environmental impact report, as a platform for participatory sensing systems research," in Proceedings of the 7th ACM International Conference on Mobile Systems, Applications, and Services (MobiSys’09), pp. 55–68, ACM, Wrocław, Poland, June 2009.
[9] "Microsoft Research Sense Web Project," March 2013, http://research.microsoft.com/en-us/projects/senseweb/.
[10] R. Crepaldi, R. Beavers, B. Ehrat, and R. Kravets, “Illinois vehicular project, live data sampling and opportunistic internet connectivity,” in Proceedings of the 3rd ACM International Workshop on Mobile Opportunistic Networks (MobiOpp ’12), pp. 85–86, ACM, Zurich, Switzerland, March 2012.
[11] C. Cornelius, A. Kapadia, D. Potz, D. Peebles, M. Shin, and N. Triandopoulos, “AnonySense: privacy-aware people-centric sensing,” in Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services (MobiSys ’08), pp. 211–224, ACM, Breckenridge, Colo, USA, June 2008.
[12] J. Shi, R. Zhang, Y. Liu, and Y. Zhang, “PriSense: privacy-preserving data aggregation in people-centric urban sensing systems,” in Proceedings of the IEEE International Conference on Computer Communications (INFOCOM ’10), San Diego, Calif, USA, March 2010.
[13] T. Feng, C. Wang, W. Zhang, and L. Ruan, “Confidentiality protection for distributed sensor data aggregation,” in Proceedings of the 27th IEEE International Conference on Computer Communications (INFOCOM ’08), pp. 475–483, Phoenix, Ariz, USA, April 2008.
[14] W. Zhang, C. Wang, and T. Feng, “GP2S: generic privacy-preservation solutions for approximate aggregation of sensor data,” in Proceedings of the 6th IEEE Annual International Conference on Pervasive Computing and Communications (PerCom ’08), pp. 179–184, Hong Kong, March 2008.
[15] D. Wagner, “Resilient aggregation in sensor networks,” in Proceedings of the 4th ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN ’06), pp. 71–82, Alexandria, Va, USA, October 2006.
[16] H. Chan, A. Perrig, and D. Song, “Secure hierarchical innetwork aggregation in sensor networks,” in Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS ’06), pp. 278–287, ACM, Alexandria, Va, USA, November 2006.
[17] M. Conti, L. Zhang, S. Roy, R. di Pietro, S. Jajodia, and L. V. Mancini, “Privacy-preserving robust data aggregation in wireless sensor networks,” Security and Communication Networks, vol. 2, no. 2, pp. 195–213, 2009.
[18] P. Samarati and L. Sweeney, “Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression,” Tech. Rep., SRI International, 1998.
[19] L. Sweeney, “k-anonymity: a model for protecting privacy,” International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, vol. 10, no. 5, pp. 557–570, 2002.
[20] R. L. Rivest, L. Adleman, and M. L. Dertouzos, “On data banks and privacy homomorphisms,” Foundations of Secure Computation, vol. 32, no. 4, pp. 169–178, 1978.
[21] M. van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan, “Fully homomorphic encryption over the integers,” in Advances in Cryptology—EUROCRYPT, 2010, H. Gilbert, Ed., vol. 6100 of Lecture Notes in Computer Science, pp. 24–43, Springer, Berlin, Germany, 2010.
[22] C. Fontaine and F. Galand, “A survey of homomorphic encryption for nonspecialists,” *Eurasip Journal on Information Security*, vol. 2007, Article ID 13801, 2007.

[23] P. Feldman, “A practical scheme for non-interactive verifiable secret sharing,” in *Proceedings of the 28th IEEE Annual Symposium on Foundations of Computer Science (FOCS ’87)*, 1987.

[24] L. Harn, “Secure secret reconstruction and multi-secret sharing schemes with unconditional security,” *Security and Communication Networks*, 2013.

[25] Y. Yan and H. Hu, “Research and realization of security electronic voting plan based on homomorphic commitment verifiable secret sharing,” *Applied Mechanics and Materials*, vol. 263–266, pp. 1673–1676, 2013.

[26] E. Prouff and T. Roche, “Higher-order glitches free implementation of the AES using secure multi-party computation protocols,” in *Cryptographic Hardware and Embedded Systems—CHES*, B. Preneel and T. Takagi, Eds., vol. 6917 of *Lecture Notes in Computer Science*, pp. 63–78, Springer, Berlin, Germany, 2011.

[27] K.-M. Chung, Y. Kalai, and S. Vadhan, “Improved delegation of computation using fully homomorphic encryption,” in *Advances in Cryptology—CRYPTO, 2010*, T. Rabin, Ed., vol. 6223 of *Lecture Notes in Computer Science*, pp. 483–501, Springer, Berlin, Germany, 2010.

[28] R. L. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems,” *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[29] V. A. Oleshchuk and Z. Vladimir, “Secure multi-party computations and privacy preservation: results and open problems,” *Telektronikk*, vol. 103, no. 2, article 20, 2007.

[30] S. L. From and T. Jakobsen, *Secure multi-party computation on integers [M.S. dissertation]*, Datalogisk Institut, Aarhus Universitet, Aarhus, Denmark, 2006.