WIRELESS COMMUNICATIONS AND MOBILE COMPUTING
Wirel. Commun. Mob. Comput. 2010; **10**:843–856
Published online 5 May 2009 in Wiley InterScience
(www.interscience.wiley.com) DOI: 10.1002/wcm.796

MDPA: multidimensional privacy-preserving aggregation scheme for wireless sensor networks

Xiaodong Lin[1], Rongxing Lu[2] and Xuemin (Sherman) Shen[2]*,†

[1]*Faculty of Business and Information Technology, University of Ontario Institute of Technology, Oshawa, Ontario L1H 7K4, Canada*
[2]*Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada*

Summary

In this paper, we propose a novel multidimensional privacy-preserving data aggregation scheme for improving security and saving energy consumption in wireless sensor networks (WSNs). The proposed scheme integrates the super-increasing sequence and perturbation techniques into compressed data aggregation, and has the ability to combine more than one aggregated data into one. Compared with the traditional data aggregation schemes, the proposed scheme not only enhances the privacy preservation in data aggregation, but also is more efficient in terms of energy costs due to its unique multidimensional aggregation. Extensive analyses and experiments are given to demonstrate its energy efficiency and practicability. Copyright © 2009 John Wiley & Sons, Ltd.

KEY WORDS: wireless sensor network; multidimensional; privacy-preserving data aggregation; energy efficiency

1. Introduction

Wireless sensor networking has been subject to extensive research efforts in recent years, and has been well recognized as a ubiquitous and general approach for some emerging applications such as real-time traffic monitoring, ecosystem and battlefield surveillance [1–3]. A wireless sensor network (WSN) is usually composed of a large number of sensor nodes which are interconnected through wireless links to perform distributed sensing tasks. Each sensor node is cheap and with low battery power and computation capacity, but is equipped with sensing, data processing, and communicating components. When sensor nodes receive a certain query from the data collection unit (also known as *sink*), the sensor nodes will report their sensing results through predetermined paths.

Due to the large scale of WSNs and resource constraints of the sensor nodes, reporting the raw data sensed by each sensor node may significantly increase the energy consumption for communication. In case that each sensor node can compress and aggregate the sensed data before launching it in the network, the communication overhead can be largely decreased

*Correspondence to: Xuemin (Sherman) Shen, Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada.
†E-mail: xshen@bbcr.uwaterloo.ca

at the expense of some computation efforts, which, nonetheless, will result in overall energy saving at each sensor node [4]. Therefore, data aggregation[‡] is considered as an important operation at each sensor node in a large-scale WSN for saving energy consumption and prolonging the network lifetime.

Recently, several data aggregation schemes have been proposed for WSNs [4–9]. The nature of unattendedness makes each sensor node susceptible to capture and compromise by a malicious adversary. If there is no privacy protection in a data aggregation scheme, the compromised nodes can overhear the transmissions and obtain the sensitive data. Therefore, privacy preservation has become an important security requirement for secure data aggregation in WSNs [10]. In most previously reported secure data aggregation schemes, some nodes are chosen and each in charge of data aggregation for a number of other nodes nearby. Since the data is encrypted, an aggregation node has to first decrypt the received data before aggregation according to the corresponding aggregation function, and then it has to encrypt the aggregated data and send it to the *sink*. Such a privacy protection strategy is straightforward and effective, but it takes high computation cost at the aggregation node. Concealed Data Aggregation (CDA) was introduced in Reference [9] to mitigate the limitation by adopting homomorphic encryption ciphers and allowing efficient aggregation of encrypted data without decryption involved in each aggregation node. However, since the scheme has all sensor nodes to share a common secret key with the *sink*, an adversary can get the secret key and access to the encrypted aggregated data by compromising any one of the sensor nodes. Clustered-based Private Data Aggregation (CPDA) scheme was proposed to overcome the weakness [10], however the aggregated data will be known to the aggregation node. Most recently, two perturbation-based data aggregation schemes have been presented [11,12], which can improve the privacy preservation. However, the studies of [11,12] are limited to the traditional single-dimensional data aggregation. In reality, a sensor node may be in charge of more than one data, i.e., multidimensional sensing. For example, humidity sensors from EnviroMon [13] are capable of sensing both humidity and temperature.

In order to further improve the performance and enhance the privacy preservation, in this paper, we propose a novel MultiDimensional Privacy-preserving data Aggregation scheme (MDPA) for WSNs. The proposed MDPA scheme integrates the super-increasing sequence and perturbation techniques [11] into data aggregation, and has the ability to combine more than one aggregated data into one, i.e., multidimensional data aggregation, to improve not only the energy efficiency but also the privacy preservation of data aggregation. The main contributions of this paper are as follows:

First, we propose a novel privacy-preserving multidimensional data aggregation scheme. With the scheme, when the aggregation result arrives at the *sink*, the *sink* can recover all aggregated data from the single result. To the best of our knowledge, this is the first effort on multidimensional privacy-preserving data aggregation.

Second, we study the theoretical upper limit for parameter k in k-neighbor aggregation, i.e., for any n sensor nodes' uniform deployment, $\max(\mathsf{k})$ is $\frac{1}{2} + \sqrt{2n - \frac{7}{4}}$. This result is very useful in system initialization of WSNs.

Third, we develop a Java simulator to study the contamination issue of aggregated data caused by sensor node compromise attack.

The remainder of this paper is organized as follows. In Section 2, we introduce the system model and design goal. In Section 3, we present the MDPA scheme. The security and performance analyses are given in Sections 4 and 5, respectively. Finally, we draw our conclusions in Section 6.

2. System Model

In this section, we characterize the multidimensional privacy-preserving data aggregation in WSNs and identify the design goal.

2.1. System Model

We consider a heterogenous sensor network which consists of a *sink* and large numbers of, i.e., n, sensor nodes arbitrarily deployed in a certain area, as shown in Figure 1. These sensor nodes are further divided into two categories: ordinary sensor nodes $\mathcal{N} = \{N_1, N_2, \ldots, N_{n_1}\}$ and aggregation (sensor) nodes $\mathcal{A} = \{A_1, A_2, \ldots, A_{n_2}\}$, where $N_i, A_j \in \{0, 1\}^*$ are uniquely identifiers for each ordinary sensor node and aggregation node, respectively, and $n_1 + n_2 = n$. Each ordinary sensor node $N_i \in \mathcal{N}$ is stationary, monitoring

[‡]The data aggregation under consideration in this paper refers to compressed data aggregation, i.e., 'sum' and 'average'.

DOI: 10.1002/wcm

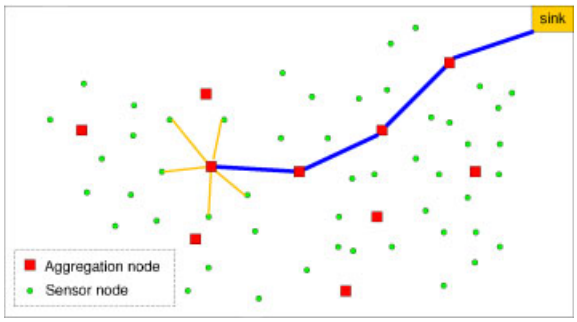

Fig. 1. Heterogenous sensor network under consideration.

the immediate surroundings and periodically collecting the sensed data. To save energy, the sensed data will be first sent to an aggregation node $A_j \in \mathcal{A}$ for aggregation with the other received data, and then the aggregation node A_j will report the aggregated data to the *sink* over a predefined path. We assume that each aggregation node $A_j \in \mathcal{A}$ is different from the ordinary sensor node in heterogenous sensor networks [14]. The tasks of an aggregation node mainly include aggregating the sensed data and forwarding the result to the *sink*. The *sink*, compared to the sensor nodes, is a trust and powerful device, which is either static or mobile, responsible for collecting the information with a certain query to the sensor nodes.

2.2. Design Goal

Before describing our design goal, we first make some necessary assumptions in our model, which are similar as those in Reference [14].

Assumption 1. The *sink* is expensive and actively powered, and will not be compromised by the adversary.

Assumption 2. The number of the aggregation nodes $\mathcal{A} = \{A_1, \ldots, A_{n_2}\}$ is much smaller compared with the ordinary sensor nodes, and thus each aggregation node $A_j \in \mathcal{A}$ can be powerful and equipped with tamper-resistant hardware. Consequently, it is reasonable to assume an adversary cannot compromise an aggregation node.

Assumption 3. The number of sensor nodes $\mathcal{N} = \{N_1, \ldots, N_{n_1}\}$ is large, and each sensor node $N_i \in \mathcal{N}$ can sense more than one data. However, due to cost constraints, these sensor nodes are not equipped with tamper-resistant hardware. Then, assume that if an adversary compromises a sensor node, she/he can extract all key materials, data and stored codes.

Assumption 4. The *sink* is in charge of initializing all sensor nodes and aggregation nodes, i.e., each node $\in \mathcal{A} \cup \mathcal{N}$ is time synchronized, uniquely identified, and preloaded with some system parameters, and has a symmetric shared key with the *sink*.

Our design goal is to develop a multidimensional privacy-preserving data aggregation scheme for WSNs. Specifically, the multidimensional aggregation scheme should achieve the desirable requirements on enhanced privacy preservation and better energy efficiency, which are explicitly defined as follows.

1. *Multidimensional aggregation and better energy efficiency:* The goal of data aggregation in WSNs is to reduce the communication overhead. The goal of multidimensional aggregation is to further improve the energy efficiency.
2. *Enhanced privacy preservation:* In multidimensional aggregation, each sensor node *only* knows its own sensed data, and the aggregated data is *only* known to the *sink*. It is hard for an adversary to obtain the aggregated data even though it launches the possible attacks listed in the paper, namely the passive attack and sensor node compromise attack. Since our goal is to enhance the privacy preservation of aggregation data, other active attacks that are irrelative to the privacy preservation, such as some denial-of-service (DoS) attacks are outside the scope of this paper.

3. Proposed MDPA Scheme

In this section, we present the proposed MDPA scheme for WSNs. The scheme consists of four phases: system initialization phase, sensor and aggregation nodes initialization phase, deployment and neighbor-key discovery phase, and multidimensional privacy-preserving data aggregation phase.

3.1. System Initialization

In the system initialization phase, to establish a WSN, the *sink* first runs the similar operations in Reference [15] to generate the bilinear parameter 5-tuple $(q, \mathbb{G}, \mathbb{G}_T, e, P)$, where q is a prime number, $\mathbb{G}$, $\mathbb{G}_T$ are two cyclic groups of the same order q, $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is an admissible bilinear map and P is a generator of $\mathbb{G}$. Then, the *sink* chooses a random number $s \in \mathbb{Z}_q^*$ as the *master key*, and computes the system public key $P_{\text{pub}} = sP$. To achieve

DOI: 10.1002/wcm

multidimensional aggregation, the *sink* also initializes the following parameters:

- p: a large prime, which denotes the required aggregation space. Usually, p is 1024-bit length same as that in digital signature algorithm (DSA) [16].
- n: the aggregation dimension, which means each sensor node can collect and report n dimensional data to the *sink*. In this paper, we assume that $1 \leq \mathsf{n} \leq 5$.
- k: the possible maximal k-neighbor aggregation, which denotes that maximal k ordinary sensor nodes will be involved in an aggregation query.
- d: a constant number, which denotes the maximal value for all data sensed by a node. For example, a group of collected data $(d_1, d_2, \ldots, d_{\mathsf{n}}) \in \mathbb{Z}^{\mathsf{n}}$ will satisfy $1 \leq d_i \leq \mathsf{d}$ for all $1 \leq i \leq \mathsf{n}$. (Note that some collected data may not be integer in its original form, but they can be easily transformed into an integer [12].)
- $\mathbf{a} = (a_1, a_2, \ldots, a_{\mathsf{n}})$: a super-increasing sequence such that $\sum_{j=1}^{i-1} a_j \cdot \mathsf{k} \cdot \mathsf{d} < a_i$ for $i = 1, 2, \ldots, \mathsf{n}$, and $\sum_{i=1}^{n} a_i \cdot \mathsf{k} \cdot \mathsf{d} < p$. That is,

$$\begin{cases} a_1 \cdot \mathsf{k} \cdot \mathsf{d} < a_2 \\ (a_1 + a_2) \cdot \mathsf{k} \cdot \mathsf{d} < a_3 \\ \cdots < \cdots \\ (a_1 + a_2 + \cdots + a_{n-1}) \cdot \mathsf{k} \cdot \mathsf{d} < a_n \\ (a_1 + a_2 + \cdots + a_{n-1} + a_n) \cdot \mathsf{k} \cdot \mathsf{d} < p \end{cases} \quad (1)$$

- H, h: two secure cryptographic hash functions [16], where $H : \{0,1\}^* \to \mathbb{G}$ and $h : \{0,1\}^* \to \{0,1\}^{\log_2 q}$.

At the end of this phase, the *sink* keeps the *master key* s secretly and sets the public parameters as $\mathbf{params} = (q, \mathbb{G}, \mathbb{G}_T, e, P, P_{pub}, p, \mathsf{n}, \mathsf{k}, \mathsf{d}, \mathbf{a}, H, h)$.

Discussion on k-neighbor aggregation. From Equation (1), the parameter k is important, which denotes the possible maximal k-neighbor aggregation. Therefore, given the total number of sensor nodes n in some deployment, it is significant to determine the theoretical upper limit of k.

Lemma 1. *Let n nodes $(n \geq 3)$ $\mathrm{N}_1, \mathrm{N}_2, \cdots, \mathrm{N}_n \in \mathcal{N} \cup \mathcal{A}$ be deployed in an area such that any three nodes $\mathrm{N}_i, \mathrm{N}_{i+1}, \mathrm{N}_{i+2} \in \mathcal{N}$ are not collinear. If, for each node $\mathrm{N}_i \in \mathcal{N} \cup \mathcal{A}$ in some deployment scheme, there are at least k nodes having the same distance to N_i[§], then k subjects to*

$$k \leq \frac{1}{2} + \sqrt{2n - \frac{7}{4}} \quad (2)$$

Proof. See appendix. ■

Example. As shown in Figure 2, when $n = 6$,

$$\mathsf{k} \leq \frac{1}{2} + \sqrt{2 \times 6 - \frac{7}{4}} \approx 3.7 \Rightarrow K = \max(\mathsf{k}) = 3 \quad (3)$$

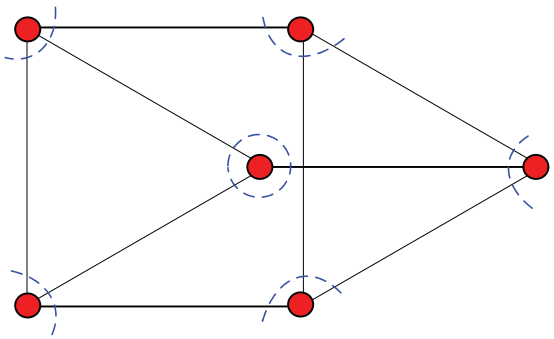

Fig. 2. An example for K-neighbor network topology connection for $n = 6$ and $K = \max(\mathsf{k}) = 3$.

Lemma 1 provides the necessary condition for k-neighbor aggregation but not the sufficient condition. Therefore, for different values of n, the lemma does not ensure the existence of $K = \max(\mathsf{k})$ in Equation (2). In addition, though the deployment of sensor nodes in WSNs is assumed to be uniformly distributed, the local density of sensor nodes may vary throughout the network [17], which also makes it more difficult to deploy the K-neighbor aggregation. Therefore, Lemma 1 just gives us a theoretical upper limit of k. In practical application scenarios, since the upper limit $K = \lfloor \frac{1}{2} + \sqrt{2n - \frac{7}{4}} \rfloor$ could not be researched for a WSN with n sensor nodes, we can choose the parameter k such that $\mathsf{k} = K$ in system initialization phase, which ensures that Equation (1) can adapt to most aggregation nodes in the whole WSNs.

[§]Here, at least k nodes have the same distance to N_i means that at least k nodes are within the transmission range of node N_i and can directly communicate with N_i.

DOI: 10.1002/wcm

3.2. Sensor and Aggregation Nodes Initialization

In this phase, the *sink* initializes all sensor nodes $\mathcal{N} = \{N_1, N_2, \ldots, N_{n_1}\}$ and all aggregation nodes $\mathcal{A} = \{A_1, A_2, \ldots, A_{n_2}\}$, which are going to be deployed. The detailed initialization algorithm is described in Algorithm 1. After all sensor nodes and aggregation nodes are preloaded with their key materials and necessary energy, they can be deployed.

3.3. Deployment and Neighbor-Key Discovery

In this phase, all sensor nodes $\mathcal{N} = \{N_1, N_2, \ldots, N_{n_1}\}$ and aggregation nodes $\mathcal{A} = \{A_1, \ldots, A_{n_2}\}$ will be deployed at a geographical area by the *sink* in various ways such as by air or by land. Given the rich literature in sensor nodes deployment, here we do not address the detailed deployment operations. Without loss of generality, we assume that all nodes will be almost uniformly distributed after deployment as shown in Figure 1, then most aggregation nodes in $\mathcal{A}$ have k neighboring sensor nodes, where $k \leq \mathsf{k}$.

Algorithm 1: Initialization of sensor nodes

Input: n_1 un-initialized sensor nodes and n_2 un-initialized aggregation nodes
Output: n_1 initialized sensor nodes $\mathcal{N} = \{N_1, \cdots, N_{n_1}\}$ and n_2 initialized aggregation nodes $\mathcal{A} = \{A_1, \cdots, A_{n_2}\}$
begin
  for $i = 1$ *to* n_1 **do**
    compute the private key $S_i = sH(N_i)$
    preload sensor node N_i with $(S_i, \mathbf{params})$ and energy
  end
  for $j = 1$ *to* n_2 **do**
    compute the private key $S_j = sH(A_j)$
    preload aggregation node A_j with $(S_j, \mathbf{params})$ and energy
  end
  return $\mathcal{N} = \{N_1, \cdots, N_{n_1}\}$ and $\mathcal{A} = \{A_1, \cdots, A_{n_2}\}$
end

To guarantee the secure subsequent data transmissions, after identifying the closest aggregation node $A_j \in \mathcal{A}$, each sensor node $N_i \in \mathcal{N}$ computes the neighbor key $\mathbf{key}_{ij}$ as

$$\mathbf{key}_{ij} = h(e(S_i, H(A_j))) \tag{4}$$

Likewise, after identifying its k neighbors $\{N_1, \ldots, N_k\} \in \mathcal{N}$, each aggregation node $A_j \in \mathcal{A}$ will generate its corresponding k neighbor keys $\mathcal{K}_j = \{\mathbf{key}_{j1}, \ldots, \mathbf{key}_{jk}\}$, where

$$\mathbf{key}_{ji} = h(e(S_j, H(N_i))) \quad \text{for } 1 \leq i \leq k \tag{5}$$

Here, each neighbor key $\mathbf{key}_{ji}$ is symmetrically shared by A_j and N_i, since

$$\begin{aligned}\mathbf{key}_{ji} &= h(e(S_j, H(N_i))) = h(e(sH(A_j), H(N_i))) \\ &= h(e(H(A_j), sH(N_i))) \quad (\because \text{bilinearity of } e) \\ &= h(e(sH(N_i), H(A_j))) = \mathbf{key}_{ij}\end{aligned} \tag{6}$$

At the same time, due to the hardness of *Bilinear Diffie-Hellman* (BDH) problem [15], each neighbor key $\mathbf{key}_{ji}$ is secure against the external attacks.

Discussion. Since the pairing operation is a time-consuming operation, the neighbor key establishment requires more energy than that with the traditional key-pool based key-distribution schemes [18]. However, compared with the key-pool based key distribution scheme, the neighbor key establishment protocol saves much more storage spaces. For example, in a random key distribution mechanism where the size of key pool is $\mathbf{K} = 5000$, and each sensor node randomly chooses $\mathbf{N}_k$ keys from the key pool, the probability that any two pair of sensor nodes hold at least one common key is

$$\begin{aligned}&\Pr(\text{the number of shared keys} \geq 1) \\ &= 1 - \frac{\binom{\mathbf{K}}{\mathbf{N}_k}\binom{\mathbf{K}-\mathbf{N}_k}{\mathbf{N}_k}}{\binom{\mathbf{K}}{\mathbf{N}_k}\binom{\mathbf{K}}{\mathbf{N}_k}} = 1 - \frac{((\mathbf{K}-\mathbf{N}_k)!)^2}{(\mathbf{K}-2\mathbf{N}_k)! \cdot \mathbf{K}!}\end{aligned} \tag{7}$$

Figure 3 shows the probability Pr varies with the number of chosen keys $\mathbf{N}_k$, when the size of key

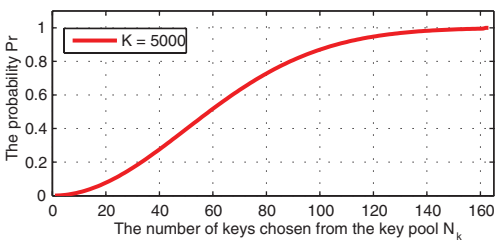

Fig. 3. The probability Pr *versus* the number of chosen keys $\mathbf{N}_k$, when the size of key pool is $\mathbf{K} = 5000$.

DOI: 10.1002/wcm

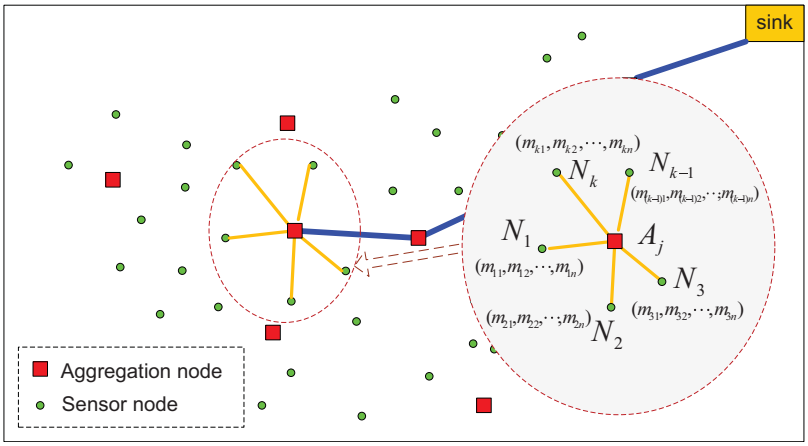

Fig. 4. Multidimensional privacy-preserving data aggregation in WSNs.

pool is $\mathbf{K} = 5000$. We can see when the probability reaches 99.5%, each sensor node should be preloaded at least 160 keys. Assume the size of each key is [128, 160] bits, the storage space for 160 keys in key pool based schemes is about [2560, 3200] bytes, while the method adopted here only requires 500 bytes or so (cf. Section 5.1). In addition, since the pairing operations are only executed in non-interactive neighbor key establishment, the energy costs of pairing is therefore not critical to the latter privacy-preserving data aggregation in WSNs. Recently, Zhang *et al.* [19] conducted a comprehensive security analysis on neighbor key establishment, which strengthens our inference.

3.4. Multidimensional Privacy-Preserving Data Aggregation

As shown in Figure 4, the proposed multidimensional data aggregation scheme mainly consists of two parts: private data aggregation (focusing on additive aggregation) and recovery of aggregated data. We describe the two parts below.

3.4.1. Private data aggregation

When the *sink* queries an aggregation node $A_j \in \mathcal{N}$ with an aggregation request E, where E is an unique identifier only for this query, the aggregation node A_j takes the role of an aggregator and broadcasts E to its k neighbor sensor nodes $\{N_1, N_2, \ldots, N_k\}$.

After receiving E, each neighbor sensor node N_i, $1 \le i \le k$, will send its sensed data $(m_{i1}, m_{i2}, \ldots, m_{in})$ to the aggregation node A_j. The sending procedure is shown in Figure 5, and the detailed steps are described as follows:

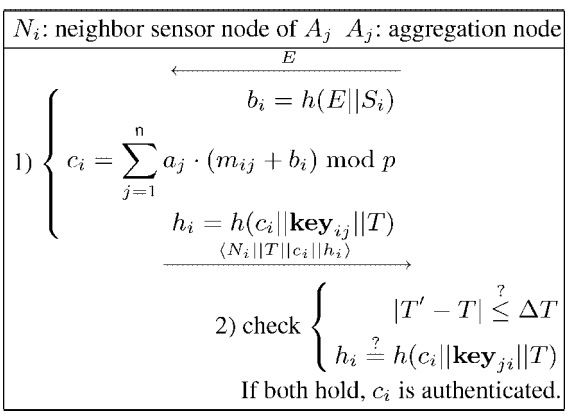

Fig. 5. Neighbor sensor node N_i sending its sensed data $(m_{i1}, m_{i2}, \ldots, m_{in})$ to the aggregation node A_j.

Step 1. N_i first applies its private key S_i and $\mathbf{a} = (a_1, a_2, \ldots, a_n)$ preloaded by the *sink* to computes $c_i = \sum_{j=1}^{n} a_j \cdot (m_{ij} + b_i) \bmod p$, where $b_i = h(E\|S_i)$. N_i then gains the current timestamp T and computes the hash value $h_i = h(c_i\|\mathbf{key}_{ij}\|T)$. In the end, N_j sends the message formatted as $\langle N_i\|T\|c_i\|h_i\rangle$ to the aggregation node A_j.

Step 2. When receiving the message $\langle N_i\|T\|c_i\|h_i\rangle$ from the neighbor sensor node N_i at timestamp T', the aggregation node A_j first checks $|T' - T| \le \Delta T$, where ΔT is the expected valid time interval for transmission delay. If it holds, the aggregation node A_j proceeds to

DOI: 10.1002/wcm

the next operation, and stops otherwise. A_j then uses the neighbor key $\mathbf{key}_{ji}$ to verify $h_i = h(c_i \| \mathbf{key}_{ji} \| T)$. If it holds, the aggregation node A_j accepts the encrypted message c_i, and rejects otherwise.

After receiving k valid encrypted data $(c_1, \ldots, c_k)$ from its neighbor sensor nodes $\{N_1, N_2, \ldots, N_k\}$, the aggregation node A_j runs the following steps:

Step 1. Compute $c = \sum_{i=1}^{k} c_i \bmod p$, set $\mathsf{NI} = \{N_1 \| \cdots \| N_k\}$ as the neighbor information of A_j, gain the current timestamp T, and compute $h_j = h(c \| \mathsf{NI} \| S_j \| T)$.

Step 2. Use the self-encryption technique [20] to encrypt the plaintext message $m = \langle E \| A_j \| \mathsf{NI} \| T \| h_j \rangle$ as $(c'_1 = rP, c'_2 = h(rP_{pub}) \cdot m \bmod p)$, where $r \in \mathbb{Z}_q^*$. Note that since $(rP, h(rP_{\text{pub}}))$ are irrelative to m, they can be offline computed for improving online efficiency. Send the formatted encrypted aggregation message $C = \langle c \| c'_1 \| c'_2 \rangle$ to the *sink*.

Discussion. We can see the self-encryption technique can hide the nodes information with P_{pub}, then no one, except the *sink*, can know the aggregation area information from C. Additionally, when an aggregation node A_j sends out an encrypted aggregation message C, no one can distinguish C is generated by A_j itself or received from the downstream aggregated nodes. Therefore, only if the adversary is not a global adversary, the location privacy of aggregation area, which may be sensitive in some applications, is achieved.

3.4.2. Recovery of aggregated data

After receiving the message $C = \langle c \| c'_1 \| c'_2 \rangle$ at the timestamp T', the *sink* performs the following steps to recover the aggregated data.

- Recover $m = \langle E \| A_j \| \mathsf{NI} \| T \| h_j \rangle$ by computing $m = \frac{c'_2}{h(s \cdot c'_1)} = \frac{h(rP_{\text{pub}}) \cdot m}{h(rP_{\text{pub}})} \bmod p$ to identify the message C comes from the aggregation node A_j concerning the event E.
- Check $|T' - T| \leq \Delta T$. If it does not hold, the message c will be rejected to resist the replay attack.
- Use A_j's private key $S_j = sH(A_j)$ to compute $h'_j = h(c \| \mathsf{NI} \| S_j \| T)$ and compare it with h_j. If h'_j is not equal to h_j, the message c is rejected too.
- According to the neighbor information $\mathsf{NI} = \{N_1 \| \cdots \| N_k\}$ of A_j, determine their private keys $(S_1, S_2, \ldots, S_k)$ of k neighbor sensor nodes, and compute $(b_1, b_2, \ldots, b_k)$, where $b_i = h(E \| S_i)$, for $1 \leq i \leq k$.
- Compute $M = c - \sum_{j=1}^{\mathsf{n}} a_j \cdot \sum_{i=1}^{k} b_i \bmod p$ and invoke the Algorithm 2 with M to recover all aggregated sum values $\mathtt{SUM}(m_{*j}) = \sum_{i=1}^{k} m_{ij}$ and the corresponding aggregated average values $\mathtt{AVG}(m_{*j}) = \frac{\sum_{i=1}^{k} m_{ij}}{k}$, for $1 \leq j \leq \mathsf{n}$.

Algorithm 2: Recover all aggregated values

Input: super-increasing sequence $\mathbf{a} = (a_1, a_2, \cdots, a_{\mathsf{n}})$ and the aggregated data M
Output: aggregated sum value $\mathtt{SUM}(m_{*j})$ and average value $\mathtt{AVG}(m_{*j})$, for $j = 1, \cdots, \mathsf{n}$.
begin
  set $\mathbb{U}$ as *null set* and $x = M$
  for $j = \mathsf{n}$ *to* 1 **do**
    $\mathtt{SUM}(m_{*j}) = \frac{x - x \bmod a_j}{a_j}$, $\mathtt{AVG}(m_{*j}) = \frac{\mathtt{SUM}(m_{*j})}{k}$
    $\mathbb{U} = \mathbb{U} \cup \{(\mathtt{SUM}(m_{*j}), \mathtt{AVG}(m_{*j}))\}$
    , $x = x \bmod a_j$
  end
  return $\mathbb{U}$
end

Correctness. From the recovery phase, we know

$$
\begin{aligned}
M &= c - \sum_{j=1}^{\mathsf{n}} a_j \cdot \sum_{i=1}^{\mathsf{k}} b_i \bmod p \\
&= \sum_{i=1}^{k} c_i - \sum_{j=1}^{\mathsf{n}} a_j \cdot \sum_{i=1}^{k} b_i \bmod p \\
&= \sum_{i=1}^{k} \sum_{j=1}^{\mathsf{n}} a_j \cdot (m_{ij} + b_i) - \sum_{j=1}^{\mathsf{n}} a_j \cdot \sum_{i=1}^{k} b_i \bmod p \\
&= \sum_{j=1}^{\mathsf{n}} a_j \cdot \sum_{i=1}^{k} (m_{ij} + b_i) - \sum_{j=1}^{\mathsf{n}} a_j \cdot \sum_{i=1}^{k} b_i \bmod p \\
&= \sum_{j=1}^{\mathsf{n}} a_j \cdot \sum_{i=1}^{k} m_{ij} \bmod p \qquad (8)
\end{aligned}
$$

DOI: 10.1002/wcm

Since $\sum_{i=1}^{k} m_{ij} < k \cdot \mathsf{d} \leq \mathsf{k} \cdot \mathsf{d}$, according to Equation (1), we have $M < p$. Therefore

$$M = \sum_{j=1}^{\mathsf{n}} a_j \cdot \sum_{i=1}^{k} m_{ij} = \sum_{j=1}^{\mathsf{n}} a_j \cdot \mathtt{SUM}(m_{*j}) \tag{9}$$

Because $\mathbf{a} = (a_1, a_2, \ldots, a_{\mathsf{n}})$ is a super-increasing sequence, i.e., $\sum_{j=1}^{i-1} a_j \cdot k \cdot \mathsf{d} \leq \sum_{j=1}^{i-1} a_j \cdot \mathsf{k} \cdot \mathsf{d} < a_i$ for $i = 1, 2, \ldots, \mathsf{n}$, the correctness of the Algorithm 2 follows.

Parameters discussion. It is well known that for a 1024-bit p, it will have the same ciphertext space as that of the classical 1024-bit RSA algorithm [16]. We demonstrate that this ciphertext space can adapt to our multidimensional data aggregation. Let total $n = 30\,000$ nodes be deployed at some area and each sensor node $N_i \in \mathcal{N}$ can at most support $\mathsf{n} = 5$ dimensional data aggregation (since large n will increase the costs of sensor nodes), and each aggregation node $A_j \in \mathcal{A}$ has at most $\mathsf{k} = 2^8$ neighbor sensor nodes based on the relation in Equation (2). Then, from Equation (1), we have the following relations

$$\begin{cases} a_5(1 + 2^8 \cdot \mathsf{d}) < 2^{1024}, & a_4(1 + 2^8 \cdot \mathsf{d}) < a_5 \\ a_3(1 + 2^8 \cdot \mathsf{d}) < a_4, & a_2(1 + 2^8 \cdot \mathsf{d}) < a_3 \\ a_1 \cdot 2^8 \cdot \mathsf{d} < a_2 \end{cases} \tag{10}$$

Furthermore, we have

$$a_1 \cdot (2^8 \cdot \mathsf{d})^5 < a_1 \cdot 2^8 \cdot \mathsf{d} \cdot (1 + 2^8 \cdot \mathsf{d})^4 < 2^{1024} \tag{11}$$

$$\Rightarrow \log_2(a_1 \cdot (2^8 \cdot \mathsf{d})^5) < \log_2 2^{1024} \tag{12}$$

$$\Rightarrow \log_2 a_1 + 5 \cdot \log_2 d < 984 \tag{13}$$

From Equation (13), for $a_1 \approx 2^{160}$ considering security, we have $d \approx 2^{164}$, which is large enough to support most practical application scenarios. Therefore, the proposed MDPA is feasible.

3.5. Numerical Example

To demonstrate the implementation of MDPA, we present a numerical example as follows:

- *Parameters setting:* Assume that the system parameters are initialized as

$$\begin{cases} (p, \mathsf{n}, \mathsf{k}, \mathsf{d}, s) = (153763, 3, 3, 10, 600) \\ \mathbf{a} = (a_1, a_2, a_3) = (5, 160, 4960) \end{cases} \tag{14}$$

For a specific query E, there are three sensor nodes $N_1, N_2, N_3 \in \mathcal{N}$ reporting their sensed data to the *sink*. The data sensed by N_1 is $(m_{11}, m_{12}, m_{13}) = (4, 6, 3)$ and assume $b_1 = h(E \| S_1) = 105$, then

$$\begin{aligned} c_1 &= \sum_{j=1}^{3} (m_{1j} + b_1) \cdot a_j \bmod p \\ &= (4 + 105) \cdot 5 + (6 + 105) \cdot 160 \\ &\quad + (3 + 105) \cdot 4960 \bmod 153763 \\ &= 553985 \bmod 153763 = 92696 \end{aligned} \tag{15}$$

The data sensed by N_2 is $(m_{21}, m_{22}, m_{23}) = (2, 4, 5)$ and assume $b_2 = h(E \| S_2) = 156$, then

$$\begin{aligned} c_2 &= \sum_{j=1}^{3} (m_{2j} + b_2) \cdot a_j \bmod p \\ &= (2 + 156) \cdot 5 + (4 + 156) \cdot 160 \\ &\quad + (5 + 156) \cdot 4960 \bmod 153763 \\ &= 824950 \bmod 153763 = 56135 \end{aligned} \tag{16}$$

The data sensed by N_3 is $(m_{31}, m_{32}, m_{33}) = (6, 8, 7)$ and assume $b_3 = h(E \| S_3) = 185$, then

$$\begin{aligned} c_3 &= \sum_{j=1}^{3} (m_{3j} + b_3) \cdot a_j \bmod p \\ &= (6 + 185) \cdot 5 + (8 + 185) \cdot 160 \\ &\quad + (7 + 185) \cdot 4960 \bmod 153763 \\ &= 984155 \bmod 153763 = 61577 \end{aligned} \tag{17}$$

- *Aggregation:*

$$\begin{aligned} c &= \sum_{i=1}^{3} c_i \bmod p \\ &= 92696 + 56135 + 61577 \bmod 153763 \\ &= 56645 \end{aligned} \tag{18}$$

DOI: 10.1002/wcm

- *Recovery:*

$$\begin{aligned} M &= c - \sum_{j=1}^{3} a_j \sum_{i=1}^{3} b_i \cdot \bmod p \\ &= 56645 - (5 + 160 + 4960) \\ &\quad \cdot (105 + 156 + 185) \bmod 153763 \\ &= (56645 - 2285750) \bmod 153763 \\ &= 77340 \end{aligned} \tag{19}$$

The following results are obtained by running Algorithm 2.

	$x = M = 77340, a_3 = 4960$
$\mathtt{SUM}(m_{*3})$ $\mathtt{AVG}(m_{*3})$	$= (x - (x \bmod a_3))/a_3 = 15$ $= \mathtt{SUM}(m_{*3})/3 = 5$
	$x = x \bmod a_3 = 2940, a_2 = 160$
$\mathtt{SUM}(m_{*2})$ $\mathtt{AVG}(m_{*2})$	$= (x - (x \bmod a_2))/a_2 = 18$ $= \mathtt{SUM}(m_{*2})/3 = 6$
	$x = x \bmod a_2 = 60, a_1 = 5$
$\mathtt{SUM}(m_{*1})$ $\mathtt{AVG}(m_{*1})$	$= (x - (x \bmod a_1))/a_1 = 12$ $= \mathtt{SUM}(m_{*1})/3 = 4$

4. Security Analysis

In this section, we analyze the security of the proposed MDPA scheme, especially on the privacy preservation of data aggregation. In addition, we discuss the contamination issue of aggregated data caused by sensor node compromise attack.

4.1. Privacy Preservation on Data Aggregation

We first discuss the data aggregation in the proposed MDPA scheme is privacy-preserving against passive attack and sensor node compromise attack.

- *Privacy-preserving against passive attack:* In MDPA, each sensor node N_i encrypts the sensed data $(m_{i1}, m_{i2}, \ldots, m_{i\mathsf{n}})$ into

$$\begin{aligned} c_i &= \sum_{j=1}^{\mathsf{n}} a_j \cdot (m_{ij} + b_i) \\ &= \underbrace{\sum_{j=1}^{\mathsf{n}} a_j \cdot m_{ij}}_{\text{part 1}} + \underbrace{\sum_{j=1}^{\mathsf{n}} a_j \cdot b_i \bmod p}_{\text{part 2}} \end{aligned} \tag{20}$$

with the randomized sequence $\mathbf{a} = (a_1, a_2, \ldots, a_{\mathsf{n}})$ and a shared key $b_i = h(E \| S_i)$. Due to the perturbation of part 2, without knowing the value of $b_i = h(E \| S_i)$, it is impossible for a passive adversary to recover part 1. Similarly, for the aggregated ciphertext

$$c = \sum_{i=1}^{k} c_i = \underbrace{\sum_{i=1}^{k} \sum_{j=1}^{\mathsf{n}} a_j \cdot m_{ij}}_{\text{part 1}} + \underbrace{\sum_{i=1}^{k} \sum_{j=1}^{\mathsf{n}} a_j \cdot b_i \bmod p}_{\text{part 2}} \tag{21}$$

without knowing all shared keys $b_i = h(E \| S_i)$, where $1 \le i \le k$, it is also impossible for a passive adversary to get part 2 and recover part 1. From these analyses, we can know only the *sink*, due to having the *master key s*, can recover the aggregated values, any other nodes including the aggregation node, cannot break the privacy of the aggregated data. As a result, the proposed MDPA scheme is privacy-preserving against the passive attack.

- *Privacy-preserving against sensor node compromise attack:* In MDPA, since the sensor nodes are inexpensive, an active adversary could compromise some sensor nodes. Thus, we discuss the effect of this kind of attack on privacy-preserving data aggregation.

1. *Compromising a sensor node doesn't disclose the data sensed by other nodes or aggregated data.* With CDA in Reference [9], all sensor nodes share the same secret key and use the secret key to encrypt the sensed data. Therefore, once a sensor node is compromised by an adversary, the adversary can break the privacy of each sensed data and aggregated data. Different from CDA, MDPA has each sensor node N_i to share a unique secret key $S_i = sH(N_i)$ with the *sink*. Therefore, even though an adversary compromises one of the sensor nodes, still cannot

DOI: 10.1002/wcm

know other sensor nodes' sensed data and the aggregated data.

2. *Compromising a sensor node also does not disclose other sensor nodes' private key.* During an aggregation query E, a sensor node N_i reads $(m_{i1}, m_{i2}, \ldots, m_{i\mathsf{n}})$, and its neighboring node N_{i+1} possibly senses the same data $(m_{(i+1)1}, m_{(i+1)2}, \ldots, m_{(i+1)\mathsf{n}})$, i.e., $m_{(i+1)j} = m_{ij}$, for $j = 1, 2, \ldots, \mathsf{n}$. In this special scenario, we consider whether N_{i+1}'s private key will be disclosed when N_i is compromised. Since

$$\begin{cases} c_i = \sum_{j=1}^{\mathsf{n}} a_j \cdot m_{ij} + \sum_{j=1}^{\mathsf{n}} a_j \cdot b_i \bmod p \\ c_{i+1} = \sum_{j=1}^{\mathsf{n}} a_j \cdot m_{(i+1)j} + \sum_{j=1}^{\mathsf{n}} a_j \cdot b_{(i+1)} \bmod p \\ m_{(i+1)j} = m_{ij}, \text{ for } j = 1, 2, \ldots, \mathsf{n} \end{cases} \tag{22}$$

we have

$$b_{i+1} = b_i + \frac{c_{i+1} - c_i}{\sum_{j=1}^{\mathsf{n}} a_j} \tag{23}$$

If b_{i+1} is computed only from N_{i+1}'s private key S_{i+1}, i.e., $b_{i+1} = h(S_{i+1})$ is applied to all queries. Then, only if the above scenario takes place, N_{i+1}'s private key $b_{i+1} = h(S_{i+1})$ will be disclosed. However, in MDPA, b_{i+1} is actually derived not only from N_{i+1}'s private key S_{i+1} but also from E, i.e., $b_{i+1} = h(E \| S_{i+1})$. Since one-wayness of $h()$ and the identifer E is distinct in different query, the above scenario does not affect N_{i+1}'s private key S_{i+1}. Therefore, compromising a sensor node also does not disclose other sensor nodes' private keys.

4.2. Contamination Issue on Aggregated Data

Based on the above analyses, compromising a sensor node does not disclose the aggregated data. However, if an adversary compromises at least one sensor nodes involved in the query E, then the compromised node could report false data. Since the compromised node still holds a valid private key, the aggregation node cannot detect the false data, then the aggregated data will be contaminated. Therefore, in this subsection, we will study the contamination in the proposed MDPA scheme, using a simulator built in Java.

In this simulation, 2500 ordinary sensor nodes and 150 aggregation nodes with a transmission radius of 20 m are deployed to cover an interest area of 200 m × 200 m. We divide the total simulation time into 10 time slots. At each time slot, we assume the compromise rate (CR) is given, saying $\mu = 5, 10, 15, 20$ ordinary sensor nodes are compromised per time slot. We test three kinds of experiments with parameter $k = 5, 10, 15$, respectively. Each experiment conducts the simulation 10 000 times with different random value seeds. Therefore, each of the following results are an average of repetition.

Figure 6 shows the contamination probability of aggregated data under different time slots with compromise rate $\mu = 5, 10, 15, 20$ and neighbors $k = 5, 10, 15$, from which we can make the following observations: (1) the contamination probability increases along with the increase of time slots; (2) when the compromise rate μ increases, the contamination probability will quickly increase; (3) the bigger the parameter k, the larger the contamination probability. However, for the AVG aggregation, the bigger the parameter k, the higher accurate the aggregated data. Therefore, there is a tradeoff between the accuracy of average value and the contamination when choosing the parameter k.

5. Performance Evaluation

In WSNs, energy saving is a key issue for multidimensional data aggregation. In this section, the performance of the proposed MDPA is evaluated in terms of energy efficiency.

5.1. Time and Energy Cost in Neighbor Key Establishment

Compared with the symmetric key cryptography, the public key cryptography is more expensive in terms of computation, and has been taken not suitable in WSNs in the past. However, recent reports [21] showed that the public key techniques with only software implementation is very viable on sensor nodes, and a number of subsequent studies following a public key technique have appeared in References [19,21–23].

In the proposed MDPA scheme, although the ID-based public key techniques are adopted, the expensive public key operations are only performed in non-interactive neighbor key establishment in the sensor node deployment phase. Therefore, the adoption of public key infrastructure does not increase the load of the privacy-preserving data aggregation in WSNs. In our design on neighbor key establishment, the bilinear parameters $(q, \mathbb{G}, \mathbb{G}_T, e, P)$ in Reference [24] are adopted with $|q| = 160$ bits, $|P| = 512$ bits. According to Reference [15], such bilinear parameters can achieve the same security level as 1024-bit RSA. We assume

DOI: 10.1002/wcm

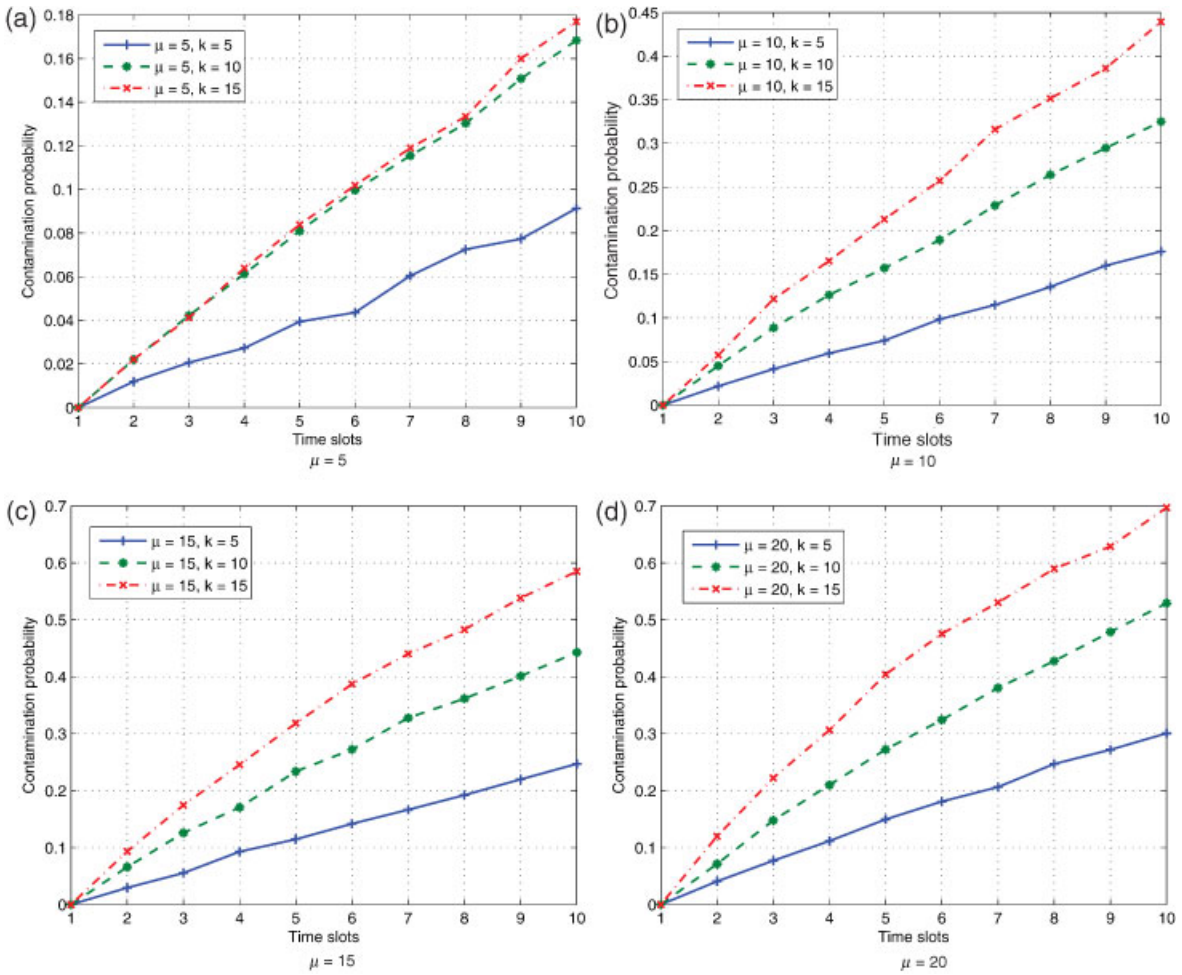

Fig. 6. The contaminations probability of aggregated data *versus* time slots with different compromise rate μ and k.

that each sensor node is equipped with a low-power high-performance 32-bit Intel PXA255 processor at 400 MHz [25]. According to References [19,22], this type of processor has been widely applied in many sensor products such as Sensoria WINS 3.0 and Crossbow Stargate. According to Reference [25], the typical power consumption of PXA 255 in active and idle modes are 411 and 121 mW, respectively. Also, it was reported in Reference [26] that the processor takes 752 ms to compute the Tate pairing with similar parameters as ours on a 32-bit ST 22 smartcard microprocessor at 33 MHz. Therefore, the computation of Tate pairing on PXA255 can be roughly estimated as $33/400 \times 752 \approx 62.04$ ms, and the energy consumption E_p is approximately 25.5 mJ. According to the Algorithm 1, the Tate pairing evaluation e by far takes the most running time of one neighbor key establishment. Therefore, the evaluation on pairing is taken to approximate the time cost and energy cost of each sensor node and aggregation node. Table I thus shows the time cost and energy cost for each sensor node and aggregation node.

Table I. Time and energy costs in key establishment.

	Time cost	Energy cost
Sensor node	62.04 ms	25.5 mJ
Aggregation node	$k \times 62.04$ ms	$k \times 25.5$ mJ

5.2. Energy Cost in Privacy-Preserving Data Aggregation

To evaluate the energy cost, the hash function $h: \{0,1\}^* \rightarrow \{0,1\}^{160}$ is implemented by SHA-1, which has good energy consumption performance with $E_h = 5.9\,\mu$J/byte [21]. We also omit the energy cost on the

DOI: 10.1002/wcm

modulus addition operation and assume the energy costs on the modulus multiplication operations is $E_m = 2\,\text{mJ}$. In addition, we denote E_{tr} and E_{re} by the hop-wise energy consumption for transmitting and receiving one byte, respectively. Based on Reference [21], when the effective data rate is 12.4 kb/s, we have $E_{\text{tr}} = 28.6\,\mu\text{J/byte}$ and $E_{\text{re}} = 59.2\,\mu\text{J/byte}$.

Let the size of the identifier of a sensor node, the event E, and the timestamp T, be all 8 bytes. Thus, $\langle N_i \| T \| c_i \| h_i \rangle$ is 164 bytes, and $C = \langle c \| c_1' \| c_2' \rangle$ is 320 bytes. Each sensor node then performs the following three activities that are considered in the energy consumption evaluation: receiving the event E, encrypting the sensed data, and transmitting the sensed data. Therefore, the total energy consumption is given by $E_{\text{sensor}} = 8 \times E_{\text{re}} + \mathsf{n} \times E_m + 156 \times E_h + 164 \times E_{\text{tr}} = 2 \times \mathsf{n} + 6.08\,\text{mJ}$.

For the energy consumption at the aggregation node, the aggregator first costs $8 \times (E_{\text{re}} + E_{\text{tr}}) = 0.70\,\text{mJ}$ on receiving E from the *sink* before broadcasting the received data to its k neighbor sensor nodes. Then, the aggregator takes $k \times (164 \times E_{\text{re}} + 156 \times E_h) = k \times 9.74\,\text{mJ}$ on receiving and authenticating the encrypted sensed data from its k neighbor sensor nodes, and $(k \times 8 + 156) \times E_h + E_m + 320 \times E_{\text{tr}} = k \times 0.05 + 12.07\,\text{mJ}$ on aggregating and sending the aggregated data to the *sink*. Therefore, the total amount of energy consumed at the aggregation node is given by $E_{\text{agg}} = k \times 9.79 + 12.07\,\text{mJ}$.

Assume the aggregated data requires α hops to reach the *sink*. Then, we get the total energy consumed by intermediate aggregation nodes is $E_{\text{hop}} = \alpha \times 320 \times (E_{\text{tr}} + E_{\text{re}}) = \alpha \cdot 28.10\,\text{mJ}$. The total energy consumed for one aggregation query in MDPA is

$$E_{\text{MDPA}} = k \times E_{\text{sensor}} + E_{\text{agg}} + E_{\text{hop}} \tag{24}$$

$$= k \cdot (\mathsf{n} \cdot 2 + 15.87) + \alpha \cdot 28.10 + 12.07\,\text{mJ} \tag{25}$$

Figure 7 shows the energy costs in MDPA n-dimensional aggregation as well as in the tradition data aggregation when $k = 8$ and $\alpha = 10$. n reflects aggregation dimension. From the figure we can see that the energy costs are same when $\mathsf{n} = 1$. With the increase of n, MDPA costs almost the same energy, the slight difference is caused by the modulus multiplication operations in involved sensor nodes; while the energy costs in the traditional data aggregation will increase linearly. This result demonstrates the MPDA is particularly efficient.

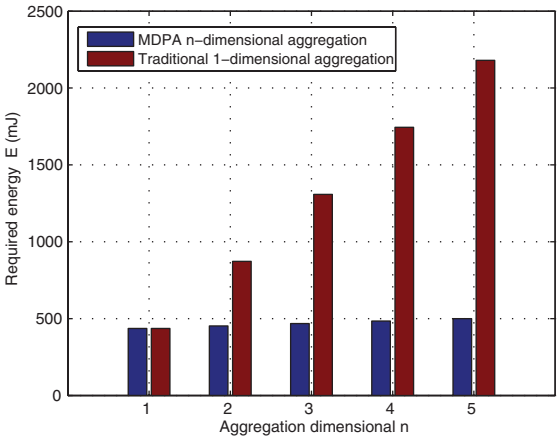

Fig. 7. Energy comparisons between MDPA n-dimensional aggregation and traditional aggregation, $k = 8$, $\alpha = 10$.

6. Conclusions

In this paper, we have proposed a novel multidimensional privacy-preserving data aggregation scheme for WSNs, named MDPA. The proposed MDPA employs the super-increasing sequence and perturbation techniques to achieve multidimensional aggregation, which not only enhances the confidentiality but also improves the communication performance. Extensive analyses and experiments have also been conducted to demonstrate that the proposed MDPA is more secure and efficient than those existing aggregation schemes.

Appendix

We will show that the relation $\mathsf{k} \leq \frac{1}{2} + \sqrt{2n - \frac{7}{4}}$ in Lemma 1 holds. Consider n nodes as n vertices of a graph $G(V, E)$, i.e., $V = \{\mathtt{N}_1, \mathtt{N}_2 \ldots, \mathtt{N}_n\}$. Assume that for any two vertices $N_i, N_j \in V$, the edge $\mathtt{N}_i\mathtt{N}_j \in E$ always exists, which means they can directly communicate with each other. Then, $G(V, E)$, denoted by K_n, is a complete graph that has total

$$\binom{n}{2} = \frac{n(n-1)}{2} \tag{25}$$

edges. We can also consider the graph K_n from another view of point. For any vertex $\mathtt{N}_i \in V$, there are at least k neighboring vertices in V having the same distance to $\mathtt{N}_i$. Then, these at least k vertices forms the *neighbors* of $\mathtt{N}_i$, denoted by $\mathcal{N}(\mathtt{N}_i)$. Because K_n is a complete

DOI: 10.1002/wcm

graph, $\mathcal{N}(\mathtt{N}_i)$ then totally contains

$$\binom{\mathsf{k}}{2} = \frac{\mathsf{k}(\mathsf{k}-1)}{2} \tag{26}$$

edges. For any two distinct vertices $\mathtt{N}_i, \mathtt{N}_j \in V$, each vertex in $\mathcal{N}(\mathtt{N}_i) \cap \mathcal{N}(\mathtt{N}_j)$ has the same distance to vertices $\mathtt{N}_i$ and $\mathtt{N}_j$. Then, the vertex number in $\mathcal{N}(\mathtt{N}_i) \cap \mathcal{N}(\mathtt{N}_j)$ could be 0, 1 and 2. Therefore, there is at most 1 edge in K_n contained in $\mathcal{N}(\mathtt{N}_i) \cap \mathcal{N}(\mathtt{N}_j)$.

Because there are total n vertices in K_n, the total edge number in these *neighbors* is then at least

$$n \cdot \binom{\mathsf{k}}{2} - 1 \cdot \binom{n}{2} = \frac{n\mathsf{k}(\mathsf{k}-1)}{2} - \frac{n(n-1)}{2} \tag{27}$$

Since the relation between Equation (25) and (27) is Equation (27) $\leq$ Equation (25), we have

$$\frac{n\mathsf{k}(\mathsf{k}-1)}{2} - \frac{n(n-1)}{2} \leq \frac{n(n-1)}{2} \tag{28}$$

By computing Equation (28), we have

$$\mathsf{k} \leq \frac{1}{2} + \sqrt{2n - \frac{7}{4}}. \tag{29}$$

This completes the proof. ■

References

1. Akyildiz I, Su W, Sankarasubramaniam Y, Cayirci E. A survery on sensor networks. *IEEE Communications Magazine* 2002; **40**(8): 102–116.
2. Mainwaring A, Polastre J, Szewczyk R, Culler D, Anderson J. Wireless sensor networks for habitat monitoring. *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications (WSNA)*, Atlanta, Georgia, 2002; 88–97.
3. Lu R, Lin X, Zhang C, Zhu H, Ho PH, Shen X. AICN: an efficient algorithm to identify compromised nodes in wireless sensor network. *International Conference on Communications (ICC 2008)*, Beijing, China, May 2008; 1499–1504.
4. Intanagonwiwat C, Estrin D, Govindan R, Hellerstein J. Impact of network density on data aggregation in wireless sensor networks. *Proceedings of the 22nd International Conference on Distributed Computating Systems (ICDCS 2002)*, Vienna, Austria, July 2002; 457–458.
5. Madden S, Franklin MJ, Hellerstein JM. TAG: a tiny aggregation service for ad-hoc sensor networks. *ACM SIGOPS Operating Systems Review* 2002; **36**(SI): 131–146.
6. Przydatek B, Song D, Perrig A. SIA: secure information aggregation in sensor networks. *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys)*, Los Angeles, CA, USA, November 2003; 255–265.
7. Tang X, Xu J. Extending network lifetime for precision constrained data aggregation in wireless sensor networks. *Proceedings of 25th IEEE International Conference on Computer Communications (Infocom 2006)*, Barcelona, Spain, April 2006; 1–12.
8. Yang Y, Wang X, Zhu S, Cao G. SDAP: a secure hop-by-hop data aggregation protocol for sensor networks. *ACM Transactions on Information and System Security (TISSEC)* 2008; **11**(4): 1–43.
9. Westhoff D, Girao J, Acharya M. Concealed data aggregation for reverse multicast traffic in sensor networks: encryption, key distribution, and routing adaptations. *IEEE Transactions on Mobile Computing* 2006; **5**(10): 1417–1431.
10. He W, Liu X, Nguyen H, Nahrstedt K, Abdelzaher T. PDA: privacy-preserving data aggregation in wireless sensor networks. *Proceedings of 26th IEEE International Conference on Computer Communications (Infocom 2007)*, Anchorage, Alaska, USA, May 2007; 2045–2053.
11. Castelluccia C, Mykletun E, Tsudik G. Efficient aggregation of encrypted data in wireless sensor networks. *Proceedings of the Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (IEEE MobiQuitous'05)*, San Diego, CA, USA, July 2005; 2045–2053.
12. Feng T, Wang C, Zhang W, Ruan L. Confidentiality protection for distributed sensor data aggregation. *Proceedings of 27th IEEE International Conference on Computer Communications (Infocom 2008)*, Phoenix, AZ, USA, April 2008; 475–483.
13. Enviromon.net, a division of Netmon Inc. http://www.enviromon.net/.
14. Du X, Xiao Y, Guizani M, Chen H. An effective key management scheme for heterogeneous sensor networks. *Ad Hoc Networks* 2007; **5**(1): 24–34.
15. Boneh D, Franklin M. Identity-based encryption from the Weil pairing. *Proceedings of Advances in Cryptology (Crypto 2001)*, Santa Barbara, California, USA, LNCS 2139. Springer-Verlag: NewYork, 2001; 213–229.
16. Mao W. *Modern Cryptography: Theory and Practice*. Prentice Hall PTR: Upper Saddle, River, New Jersey, 2003.
17. Blough D, Leoncini M, Resta G, Santi P. The k-neighbors approach to interference bounded and symmetric topology control in ad hoc networks. *IEEE Transactions on Mobile Computing* 2006; **5**(9): 1267–1282.
18. Liu D, Ning P. Establishing pairwise keys in distributed sensor networks. *Proceedings of the 10th ACM conference on Computer and communications security (CCS 2003)*, Washington, DC, October 2003; 52–61.
19. Zhang Y, Liu W, Lou W, Fang Y. Location based security mechanisms in wireless sensor networks. *IEEE Journal On Selected Areas In Communications* 2006; **24**(2): 247–260.
20. Lin X, Lu R, Shen X, Nemoto Y, Kato N. SAGE: a strong privacy-preserving scheme against global eavesdropping for ehealth systems. *IEEE Journal of Selected Areas of Communications* (in press).
21. Wander A, Gura N, Eberle H, Gupta V, Shantz S. Energy analysis of public key cryptography on small wireless devices. *IEEE International Conference on Pervasive Computing and Communications (PerCom 2005)*, Hawaii, USA, March 2005; 324–328.
22. Ren K, Lou W, Zhang Y. LEDS: providing location-aware end-to-end data security in wireless sensor networks. *Proceedings of 25th IEEE International Conference on Computer Communications (Infocom 2006)*, Barcelona, Spain, April 2006; 1–12.
23. Oliveira L, Aranha D, Morais E, Daguano F, Lopez J, Dahab R. TinyTate: computing the tate pairing in resource-constrained sensor nodes. *Proceedings of Sixth IEEE International*

DOI: 10.1002/wcm

Symposium on Network Computing and Applications (NCA 2007), Cambridge, Massachusetts, USA, July 2007; 318–323.
24. Barreto P, Kim H, Bynn B, Scott M. Efficient algorithms for pairing-based cryptosystems. *Proceedings of Advances in Cryptology (Crypto 2002)*, Santa Barbara, California, USA, LNCS 2442. Springer-Verlag: NewYork, 2002; 354–368.
25. Intel PXA255 Processor Electrical, Mechanical, and Thermal Specification. Available online at: http://www.intel.com/design/pca/applicationsprocessors/manuals/278780.htm
26. Bertoni G, Chen L, Fragneto P, Harrison K, Pelosil G. Computing Tate pairing on smartcards. *White Paper STMICROELECTRONICS*, 2005. Available online at: http://www.st.com/stonline/products/families/smartcard/ast_ibe.htm.

Authors' Biographies

Xiaodong Lin received the Ph.D. degree in information engineering from Beijing University of Posts and Telecommunications, Beijing, China, in 1998 and the Ph.D. degree (with Outstanding Achievement in Graduate Studies Award) in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2008. He is currently an Assistant Professor of information security with the Faculty of Business and Information Technology, University of Ontario Institute of Technology, Oshawa, ON, Canada. His research interests include wireless network security, applied cryptography, computer forensics, and anomaly-based intrusion detection. Dr Lin was the recipient of a Natural Sciences and Engineering Research Council of Canada (NSERC) Canada Graduate Scholarships (CGS) Doctoral and the Best Paper Award of the IEEE International Conference on Communications (ICC'07)—Computer and Communications Security Symposium.

Rongxing Lu is currently working toward the Ph.D degree with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario, Canada. He is currently a Research Assistant with the Broadband Communications Research (BBCR) Group, University of Waterloo. His research interests include wireless network security, applied cryptography, and trusted computing.

Xuemin (Sherman) Shen received the B.Sc.(1982) degree from Dalian Maritime University (China) and the M.Sc. (1987) and Ph.D. degrees (1990) from Rutgers University, New Jersey (U.S.A.), all in electrical engineering. He is a University Research Chair Professor, Department of Electrical and Computer Engineering, University of Waterloo, Canada. His research focuses on mobility and resource management in interconnected wireless/wired networks, UWB wireless communications networks, wireless network security, wireless body area networks and vehicular *ad hoc* and sensor networks. He is a co-author of three books, and has published more than 400 papers and book chapters in wireless communications and networks, control and filtering. Dr Shen serves as the Tutorial Chair for *IEEE ICC'08*, the Technical Program Committee Chair for *IEEE Globecom'07*, the General Co-Chair for *Chinacom'07 and QShine'06*, the Founding Chair for *IEEE Communications Society Technical Committee on P2P Communications and Networking*. He also serves as a Founding Area Editor for *IEEE Transactions on Wireless Communications*, Editor-in-Chief for *Peer-to-Peer Networking and Application*, Associate Editor for *IEEE Transactions on Vehicular Technology*, *KICS/IEEE Journal of Communications and Networks*, *Computer Networks*, *ACM/Wireless Networks*, and *Wireless Communications and Mobile Computing (Wiley)*, etc. He has also served as Guest Editor for *IEEE JSAC*, *IEEE Wireless Communications*, *IEEE Communications Magazine*, and *ACM Mobile Networks and Applications*, etc. Dr Shen received the Excellent Graduate Supervision Award in 2006, and the Outstanding Performance Award in 2004 and 2008 from the University of Waterloo, the Premier's Research Excellence Award (PREA) in 2003 from the Province of Ontario, Canada, and the Distinguished Performance Award in 2002 and 2007 from the Faculty of Engineering, University of Waterloo. Dr Shen is a registered Professional Engineer of Ontario, Canada. Dr Shen is a Fellow of IEEE.

DOI: 10.1002/wcm