Hindawi
Security and Communication Networks
Volume 2023, Article ID 6693296, 13 pages
https://doi.org/10.1155/2023/6693296

Research Article

SEMDA: Secure and Efficient Multidimensional Data Aggregation in Smart Grid without a Trusted Third Party

Zichao Song,[1,2] Weidong Zhong,[1,2] Tanping Zhou,[1,2,3] Dong Chen,[1,2] Yujie Ding,[1,2] and Xiaoyuan Yang[1,2]

[1] *College of Cryptography Engineering, Engineering University of PAP, Xi'an 710086, China*
[2] *Key Laboratory of PAP for Cryptology and Information Security, Xi'an 710086, China*
[3] *TCA Laboratory, State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing 100080, China*

Correspondence should be addressed to Weidong Zhong; wdeast@163.com and Tanping Zhou; tanping2020@iscas.ac.cn

Received 9 December 2022; Revised 9 January 2023; Accepted 12 January 2023; Published 1 February 2023

Academic Editor: Kuo-Hui Yeh

Copyright © 2023 Zichao Song 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.

Smart grids are a combination of traditional power system engineering as well as information and communications technology. Smart grid terminals provide convenient services to users by aggregating their data in real time. However, terminals can derive user privacy information from real-time data on smart devices. Therefore, security data aggregation has been widely studied in the field of smart grid. Most existing schemes are one-dimensional data aggregation or rely on a trusted third party. In reality, multidimensional data (such as a user's electricity consumption or user's main usage time, etc.) makes sense for terminals to flexibly adjust supply and demand strategies. In this paper, we propose an efficient and secure multidimensional data aggregation scheme that supports batch validation without a trusted third party. Firstly, we apply the Chinese remainder theorem to encode the user's multidimensional data and realize the independence of each dimension in terminal decryption. Secondly, we adopt a secure key negotiation protocol that does not require a trusted third party. Finally, based on paillier homomorphic encryption and bilinear pairing, we construct an encryption scheme that can reuse the key and blind factor and support batch verification. The analysis results show that our scheme is secure for users' privacy protection. Experimental results show that, compared with existing 1 dimensional aggregation schemes, our scheme has almost no growth in computational overhead for terminal decryption.

1. Introduction

In recent years, as increasing numbers of countries place more emphasis on the next generation of electricity networks, considerable human, material, and financial resources have been invested in the research and development of intelligent electricity networks. As the next generation of grids, smart grids integrate traditional grids with information and communication technologies for the efficient and reliable generation, transmission, distribution, and control of services [1–3], and these technologies will be used in future work. Smart grids will play an increasingly important role in meeting the needs of users, improving data reliability, and providing power control management [4]. In order to intelligently monitor and control the network in real time, smart grid control centers must continuously collect data on the amount of electricity consumed by users. But smart grid usage data directly reveals where, when, and what people are doing. In the event of a leak, these data can expose individual users' electricity usage patterns. For example, if a user's power consumption is almost zero at a fixed time of day, it can be inferred that the user is out at that time of day and no one else is home. If a smart device terminal obtains this data, it can use big data technology to provide personalized service recommendations to users and improve service quality. However, when an external attacker gains access to that user's electricity data, it can be sold for financial gain or it can be used to commit a crime such as a burglary, while the user is away at a later date. Clearly, exposure to such sensitive information is a threat to the privacy of these users [5]. Thus, it is particularly important to ensure the confidentiality and integrity of the information transmitted in smart grids while protecting the privacy of the users in smart grids from being compromised [6].

Security and privacy are two of the main obstacles to the development of the smart grid. How to monitor regional electricity consumption without divulging the power consumption of a single user has become the direction of scholars. The homomorphic encryption scheme guarantees that the algebraic operation of cryptography is equivalent to the direct operation of plaintext. Therefore, homomorphic encryption schemes are widely used in smart grids to protect users' privacy.

The idea of combining data aggregation with a homomorphic encryption algorithm is proposed to solve the privacy problem in smart grids. Compared with symmetric encryption algorithms, the use of homomorphic encryption algorithms and data aggregation combine to achieve both efficiency and privacy protection and to detect power leakage and fraud by comparing the aggregated results of smart meters with the total meter readings.

In 2012, Lu et al. [7] first proposed an EPPA scheme based on paillier homomorphic encryption and super-incremental sequences to address privacy concerns in smart grids. However, paillier encryption is not as efficient as BGN encryption in very small plaintext spaces; so in 2015, Chen et al. [8] proposed a BGN homomorphic encryption scheme based on MuDA data aggregation to solve the problem of small plaintext spaces and paillier encryption's inefficiency. In order to measure the interval distribution of electricity consumption by users, Li et al., (2018) proposed a PPMA scheme for data aggregation across multiple subsets using super-incremental sequences and paillier homomorphic cryptography [9]. It can calculate the amount of electricity used and the number of users at a given time. In 2019, Saleemd et al. [10] proposed a fault-tolerant FESDA scheme to address the failure of unsuccessfully decrypting meter faults. When a user fails, the Control Center (CC) uses the user's equivalent cipher to decrypt and gain good fault tolerance. However, in 2021 Wu et al. [11] found that the presence of equivalent ciphertext in the FESDA scheme can leak users' private information, attacked the FESDA scheme, and proposed a secure PDFA aggregation scheme using the extended Shamir secret-sharing scheme.

These schemes require the involvement of a trusted third party, who passes the corresponding decryption key to CC, assists in the management of user registration and deregistration, and so on. However, finding a fully trusted third party in the real world is extremely difficult, and even if such a trusted third party does exist, significant human, material, and financial resources need to be devoted to maintaining its absolute security and credibility in the future. Based on these considerations, some scholars have proposed aggregation schemes without trusting third parties. In 2016 Knirsch et al. [12] proposed a time-series spatiotemporal aggregation shield that utilizes key superposition between users to protect individual privacy while still providing sufficient bug resistance and an efficient data aggregation scheme. In 2018, Li et al. [13] proposed a data aggregation scheme without trusted third parties based on BGN homomorphic encryption. And in the same year, Gong et al. [14] proposed a distribution method that does not rely on any trusted third party to generate security parameters that tolerate conspiratorial attacks between n-2 users. In 2019, Liu et al. [15] proposed an EC-Elgamal data aggregation scheme without a trusted third party, which protects the privacy of individual users by constructing a virtual aggregation region based on the level of trust between users. However, these schemes are functionally single and do not allow for the aggregation of multidimensional data.

In the above scheme, the signature of each cryptographic text needs to be verified, but it is less efficient. Several other proposals have been floated. In 2020, Ding et al. [16] proposed a homogenous encrypted data aggregation scheme based on user identity that allows only collectors authorized by electricity service providers to decrypt, thereby protecting the privacy of the user. In 2021, Chen et al. [17] proposed a data aggregation scheme based on dynamic member groups that enables dynamic user management. In 2022, Liu et al. [18] proposed a data aggregation scheme that supports the rapid detection of deceptive users.

These solutions address the different needs of smart grids, but none of them is fully functional. In real life, we expect a scheme that can achieve multidimensional data aggregation without the involvement of a trusted third party while protecting the privacy and integrity of users' electricity data. Our scheme can solve this problem.

Our contributions are as follows:

(i) We apply the Chinese remainder theorem [19] to construct a multidimensional data aggregation scheme. The Chinese remainder theorem makes each dimension independent of the other when the terminal is decrypted, which ensures the correctness of our scheme.

(ii) We design a data aggregation scheme supporting key, blind factor reuses, and batch authentication based on paillier homomorphic encryption [20] scheme and bilinear pairing [21]. In addition, an example of fast detection of incorrectly signed users when bulk verification fails is given using combinatorial mathematical principles [22].

(iii) We apply a secure key negotiation protocol [23] that does not require the participation of trusted third parties. All devices are semi-honest throughout the process, and we prove the privacy of user information at every step.

In Section 1, we introduce the basic situation of smart grids. In Section 2, we introduce related work to the paper. In Section 3, we introduce the basic blocks as the preliminaries of our scheme. In Section 4, we introduced our system model. In Section 5, we describe the construction of our scheme in detail. In Section 6, we analyze our scheme. In Section 7, we conducted the experiment evaluation. Finally, we draw our conclusion in Section 8.

2. Related Work

In this section, we provide an overview of traditional solutions for smart grids as well as some of the latest research findings. In 2010, Li et al. [24] first proposed a distributed incremental data aggregation method, which covers the entire local neighborhood or any specified node set with minimal overhead through carefully constructed aggregation trees. At the same time, homomorphic encryption was used to ensure the security of data during transmission. This approach guarantees that all devices participate in aggregation and that no intermediate or final result is obtained by any device, so this approach is mainly used in privacy-preserving data aggregation schemes in smart grids.

Compared with the privacy-preserving one-dimensional data aggregation scheme, the multidimensional data aggregation scheme contains multiple detailed pieces of information of the user, which can not only extend the application of data aggregation but also meet the needs of the terminal for fine-grained analysis of multidimensional data and facilitate strategy adjustment. In 2012, Lu et al. [7] constructed a multidimensional data aggregation scheme by using a super-increasing sequence and homomorphic encryption scheme to obtain more electricity information of users. However, this scheme needs to encrypt the data of each dimension, which leads to excessive computational overhead and low efficiency during encryption. In 2017, Merad Boudia et al. [25] proposed an ElGamal homomorphic encryption multidimensional data aggregation scheme based on an elliptic curve, which does not require complex encryption operations on the way. However, the smart meter needs to encrypt the data for each dimension, which will greatly reduce the calculation efficiency with the increase of the dimension. In 2018, Li et al. [9] used two super-increasing sequences and paillier homomorphic encryption algorithm to construct a multisubset data aggregation scheme, which could count the electricity consumption and the number of users in a certain period of time. However, when the number of electricity consumption intervals is too large, the smart meter needs to encrypt each electricity consumption interval, which leads to excessive computational overhead and low efficiency. In 2019, Ming et al. [26] proposed to construct a multidimensional data aggregation scheme through an elliptic curve encryption method and a super-increasing sequence. The super-increasing sequence ensures that messages of each dimension will not affect each other when the terminal is decrypted, but the terminal needs to solve the discrete logarithm problem to recover data, so the scheme is not efficient. In 2020, Zuo et al. [27] proposed a privacy-preserving multidimensional data aggregation scheme based on the ElGamal homomorphic cryptosystem with distributed decryption, which can resist joint attacks from the gateway and the control center. However, the ciphertext of each user in this scheme is composed of two parts, which leads to an increase in computation cost, communication cost, and storage cost.

In recent years, with the development of fog computing, many scholars have proposed many solutions in the context of fog computing. In 2018, Lyu et al. [28] proposed a privacy-preserving fog computing aggregation scheme using one-time padded homomorphic encryption, which uses a fog node to collect the transmitted data for efficient processing and calculation. However, in this scheme, the integrity of the user's encrypted data is not guaranteed. In 2019, Liu et al. [29] proposed an aggregation scheme supporting fog computing, which used the double trapdoor encryption scheme to construct a homomorphic encryption scheme supporting service organization query and self-query query. However, encrypting each dimension in this scheme incurs an additional heavy computational cost. In 2020, Merad-Boudia et al. [30] proposed a method of using coding to aggregate multidimensional data, which gives each dimension data a certain space according to certain rules so that each dimension data does not affect each other during aggregation and decryption. However, in this method, the length of the coding used is fixed, which is easy to cause space waste and increase storage overhead.

Compared with the above schemes, our scheme uses the Chinese Remainder Theorem to encode the user's multidimensional data, which overcomes the shortcomings of increasing computational overhead with the increase in data dimension and does not cause a waste of storage space. Therefore, our scheme is more practical.

3. Preliminaries

3.1. Chinese Remainder Theorem. Let $b_1, b_2 \cdots b_n$ be l pairwise co-prime primes, $e_1, e_2 \cdots e_n$ denotes n integers, then the following congruence equations have a unique solution:

$$\begin{cases} x \equiv e_1 \mod \mathrm{b}_1 \\ x \equiv e_2 \mod b_2 \\ \quad \vdots \\ x \equiv e_n \mod b_n \end{cases}. \tag{1}$$

The form of the solution can be expressed as $x = e_1 y_1 B_1 + e_2 y_2 B_2 + \cdots + e_n y_n B_n \mod B$, where $B = b_1 b_2 \cdots b_n$, $B_i = B/b_i$, $y_i = B_i^{-1} \mod b_i$, $1 \leq i \leq n$. By $x \equiv e_i \mod b_i$, we can easily calculate e_i. In our scheme, we make full use of the properties of the Chinese remainder theorem to obtain the aggregation value of multidimensional data.

3.2. Paillier Homomorphic Encryption. The paillier public key encryption algorithm is a popular homomorphic encryption which supports homomorphic addition.

3.2.1. Key Generation. Given security parameters κ, random generation of two large primes p and q, calculates $N = pq$, $\lambda = lcm(p-1, q-1)$. Defining functions $L(x) = x - 1/N$, and randomly selecting the raw elements $g \in \mathbb{Z}_{N^2}^*$, make gcd $(L(g^\lambda \mathrm{mod}\, N^2), N) = 1$ and calculate $\mu = (L)(g^\lambda \mod N^2)^{-1} \mod N$. Then, the public key of the encryption algorithm is $PK = (N, g)$, the private key is $SK = (\lambda, \mu)$.

3.2.2. Encryption. For any plaintext $m \in \mathbb{Z}_N$, select a random integer r, where $0 < r < N, r \in \mathbb{Z}^*_{N^2}$. That is, r has a multiplication inverse in the remainder of N^2. Calculate ciphertext $c = g^m r^N \bmod N^2$.

3.2.3. Decryption. For a given ciphertext $c \in \mathbb{Z}^*_{N^2}$, calculate the plaintext $m = L(c^\lambda \bmod N^2) \cdot \mu \bmod N$.

3.3. Bilinear Pairing. Let G_1, G_2 be a cyclic group that satisfies the order of large prime number P, in which a pairing relation e: $G_1 \times G_1 \longrightarrow G_2$ is defined to meet the following conditions.

3.3.1. Bilinear. For any $g, h \in G_1, a, b \in \mathbb{Z}_p$, there is a $e(g^a, h^b) = e(g, h)^{ab}$.

3.3.2. Computability. For any $g, h \in G_1, a, b \in \mathbb{Z}_p$, there is an efficient polynomial time algorithm to calculate the value of $e(g^a, h^b), e(g, h)^{ab}$.

3.3.3. Nondegeneracy. For any $g, h \in G_1$, there is a $e(g, h) \neq 1_{G_2}$.

3.4. Uniform (k, n) – Set of a Finite Set. (k, n) threshold set is a technique in combinatorial mathematics that plays an important role in the rapid detection of error signatures, as defined below.

Let set $A = \{d_1, d_2, \cdots, d_m\}$, the subsets $A_1, A_2, \cdots A_n \in A$ is called a uniform (k, n) – *set* of A ($m \geq \begin{pmatrix} n \\ k-1 \end{pmatrix}$) if the following three conditions are satisfied:

(i) $|\mathrm{A}_1| = |\mathrm{A}_2| = \cdots = |\mathrm{A}_\mathrm{n}|$.

(ii) For any k subsets $\mathrm{A}_{\mathrm{i},1}, \cdots \mathrm{A}_{\mathrm{i,k}} \in \{\mathrm{A}_1, \mathrm{A}_2, \cdots \mathrm{A}_\mathrm{n}\}$, there is $\cup_{\mathrm{j}=1}^{\mathrm{k}} \mathrm{A}_{\mathrm{i,j}} = \mathrm{A}$.

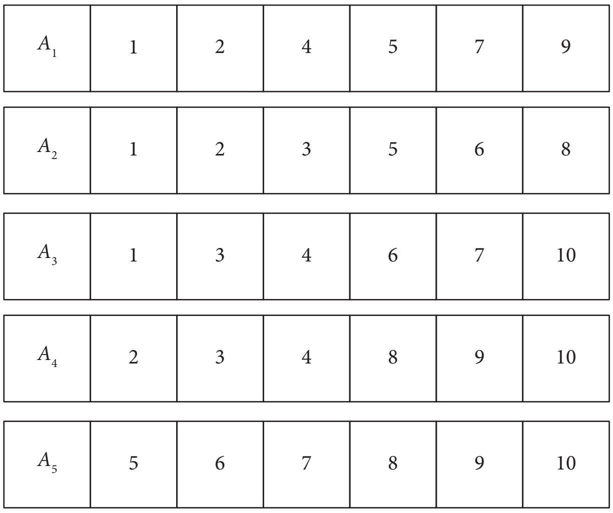
(iii) For any $k-1$ subsets $A_{i,1}, \cdots A_{i,k-1} \in \{A_1, A_2, \cdots A_n\}$, there is $\cup_{j=1}^{k-1} A_{i,j} = A \backslash \{d_i\}$.

Figure 1 shows an example of a (3,5) threshold set for $A = \{1, 2, \cdots, 10\}$.

4. System Design

4.1. System Model. In our scheme, we focus on how to securely aggregate user electricity consumption data and forward user electricity consumption data to the control center (CC) of the smart grid in a privacy-preserving way. Figure 2 shows the system model of the smart grid. There are three main players in the system model: a large number of smart meters (SM), fog nodes (FN), and CC. Their role in smart grid systems is as follows.

4.1.1. Smart Meter (SM). A smart meter is a smart device installed by a utility company on a user's premises. It collects specific real-time data from its surroundings and regularly reports it to the nearest fog node (FN). Obviously, we can easily connect smart meters to the users involved.



A_1	1	2	4	5	7	9
A_2	1	2	3	5	6	8
A_3	1	3	4	6	7	10
A_4	2	3	4	8	9	10
A_5	5	6	7	8	9	10

Figure 1: The uniform (3, 5)-set of $A = \{1, 2, \cdots, 10\}$.

4.1.2. Fog Node (FN). FN is the intermediate node between the control center and the smart meter, with a high level of storage and calculative power. It performs three main functions: (1) verifying the authenticity of the message received; (2) data aggregation of incoming messages; and (3) transferring the aggregated messages to CC.

4.1.3. Control Center (CC). After receiving the FN report, CC first verifies the authenticity and integrity of the message, then decrypts the data. By decrypting the data, CC can learn the status of the entire smart grid in real-time, which can be used to conduct data analysis and make timely strategic adjustments.

4.2. Threat Model. In our scheme, all participants (including CC and FN) were curious but honest. In general, CC and FN will interact exactly as defined, not tampering with calculations but trying to learn sensitive information from uploaded messages. In our threat model, we assume that an attacker has the following capabilities:

(i) Attackers can intercept the communication information between SM, FN, and CC. In addition, the attacker can break into some users in a residential area, which means that the attacker will obtain all the consumption data and the corresponding security parameters of these users and attempt to infer the privacy information of other users through these data.

(ii) Attackers can invade FN and CC databases, steal personal data and security parameters, and even damage FN and CC. Although the CC is powerful in reality, the data stored in it is not completely safe from the risk of being compromised because the attacker is powerful enough under our assumptions.

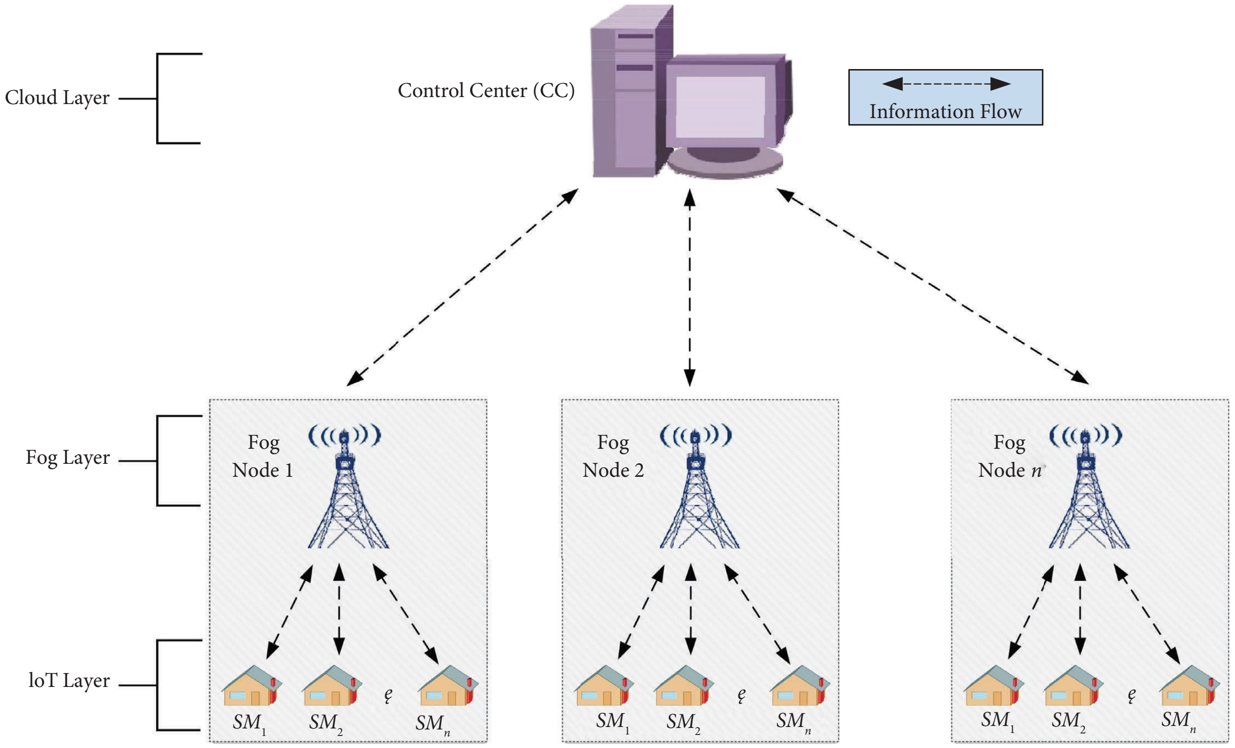


FIGURE 2: Smart grid system model.

Therefore, CC can also be considered a formidable opponent because of its curiosity about user privacy.

(iii) Attackers inject fake data into the system by intercepting messages and forging the identities of authorized users or tampering with data transmitted over a public communication channel to send fake power consumption data to the CC. In addition, attackers could launch attacks that compromise the data integrity of the smart grid.

4.3. Design Goals. Our goal is to design an efficient and stable multidimensional data aggregation solution without a trusted third party, which can protect the privacy of the user while achieving the required functionality. Specifically, our design objectives include the following design goals.

4.3.1. Security. To avoid the leakage of users' privacy, the privacy of users' electricity data should be ensured. Specifically, the electricity consumption data of users should be kept private during the whole communication process from SM to FN and then to CC. External adversaries do not have access to restore the plaintext of individual user data, ensuring that any changes to messages can be checked. The authenticity and integrity of the data transmitted and the identity of illegal users can be checked by FN and CC.

4.3.2. Privacy-Preserving. User's privacy is a key issue in smart grid applications. No one, including CC, has access to a single user's privacy information from real-time energy consumption data.

4.3.3. Functionality. The scheme can achieve the aggregation of user multidimensional data and support the integrity authentication of batch data without the participation of a trusted third party and the reuse of keys and blind factors.

4.3.4. Computing and Communication Efficiency. User meters have limited computing power. Therefore, the high efficiency of computing and communication needs to be considered in smart grids.

4.3.5. Dynamic User Management. Schemes should provide flexible equipment management mechanisms to support users in joining and exiting smart grid systems dynamically.

5. Our SEMDA Scheme

In this section, we introduce our SEMDA scheme in detail. The scheme consists of five stages: system initialization, multidimensional data coding and encryption, batch verification and data aggregation, data decryption, and dynamic user management. The symbols used in SEMDA and their descriptions are shown in Table 1.

5.1. Initialization Phase

Step 1: Given the security parameter k, system users set $\mathbb{U} = \{u_1, u_2, \cdots, u_n\}$, the system randomly selects two large primes p and q to generate the parameters (G_1, G_2, e, h). The group G_1, G_2 satisfies the bilinear relation e: $G_1 \times G_1 \longrightarrow G_2$ and h is the generator of the group G_1.

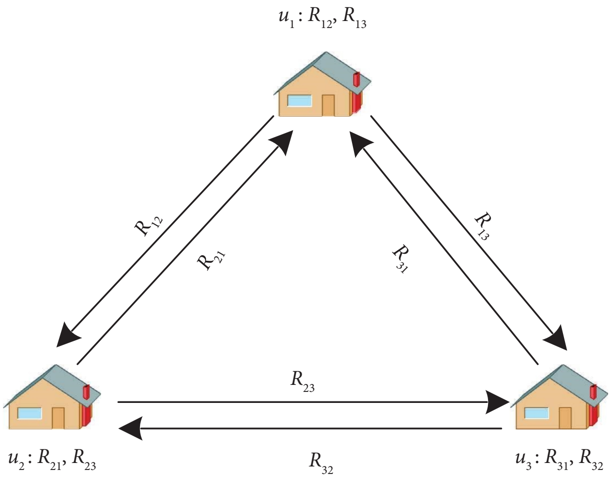
Table 1: Notations in SEMDA scheme.

Notation	Description
SM_i	Smart meters
FN	Fog node
CC	Control center
$d_{i,l}$	Fine-grained user data
m_i	The encoded plaintext message value
x_i	Key signed by the user
Y_i	User authenticated key
ID_i, ID_f	User and FN identity
c	Ciphertext after FN aggregation
M	Plaintext after CC decryption
sk_i	The private key to encrypt
β_i	User blind factor
$\Bbbk_i$	User's key set
T, T_s	Current time and time slot
H_1, H_2	H_1: $\{0,1\}^* \longrightarrow G_1, H_2$: $\{0,1\}^* \longrightarrow Z_N{}^*$
σ_i, σ_f	Signatures of user and FN
λ	CC's private key
b_i	Large prime selected by Chinese remainder theorem

Step 2: First of all, CC calculates $N = p \cdot q$, $\lambda = lcm(p-1, q-1)$; and for ease of calculation choose $g = N+1$, choose l prime numbers $\text{P} = \{b_1, b_2, b_3 \cdots b_l\}$, where $b_i \geq nd_{\max}$, $0 \leq i \leq l$, $d_{\max}$ is the upper bound of a single dimension of a user's electricity data, n is the number of users; and two Hash functions H_1: $\{0,1\}^* \longrightarrow G_1$, H_2: $\{0,1\}^* \longrightarrow Z_N{}^*$. Secondly, CC calculates $B = b_1 b_2 \cdots b_l$, $B' = \{B_i \mid B_i = B/b_i, 1 \leq i \leq l\}$, $y = \{y_i \mid y_i = B_i{}^{-1} \text{mod} b_i, 1 \leq i \leq l\}$. Then, CC randomly selects n blind factors $\beta_i \in_R Z_N{}^*, 1 \leq i \leq n$ and assigns to n users. Finally, CC public parameters $\{l, N, g, G_1, G_2, h, e, H_1, H_2, B, B'\}$, and set the private key as λ.

Step 3: User $u_i \in \mathbb{U}$ randomly selects $x_i \in_R Z_N{}^*$ as his private key and calculates $Y_i = h^{x_i}$ as his public key. Then, FN randomly select $x_f \in_R Z_N{}^*$ as his private key and calculate $Y_f = h^{x_f}$ as his public key.

Step 4: User $u_i \in \mathbb{U}$ randomly selects $R_{ij} \in_R Z_{N^2}{}^*$, where $1 \leq j \leq n-1$. $\{R_{i1}, \cdots, R_{i,n-1}\}$ form the user's shared key set. Through the secure channel, users send R_{ij} to user j. User $u_i \in \mathbb{U}$ receives the shared key of other users, which forms the key set of users $\{R_{1i}, \cdots, R_{ni}\}$. The user's key set $\{R_{i1}, \cdots, R_{i,n-1}, R_{1i}, \cdots, R_{ni}\}$ is recorded $\Bbbk_i$, and these user $u_i \in \mathbb{U}$ uses his key set to calculate the encrypt key $sk_i = R_{i1} + \cdots + R_{i,n-1} - R_{1i} - \cdots - R_{ni}$, clearly $\sum_{i=1}^n sk_i = 0$. To illustrate this, three user examples are given, as shown in Figure 3.



u_1 computes secret key : $sk_1 = R_{12} + R_{13} - R_{21} - R_{31}$

u_2 computes secret key : $sk_2 = R_{21} + R_{23} - R_{12} - R_{32}$

u_3 computes secret key : $sk_3 = R_{31} + R_{32} - R_{13} - R_{23}$

therefore $sk_1 + sk_2 + sk_3 = 0$

Figure 3: The process of generating the user key.

5.2. Reporting Phase. When a user $u_i \in \mathbb{U}$ needs to send smart meter data to FN at the interval T_s, he will implement two steps. Step one: the user collects his multidimensional data $d_i = (d_{i,1}, d_{i,2}, \cdots, d_{i,l})$ and uses the Chinese remainder theorem to compute multidimensional messages $m_i = d_{i,1}B_1 + d_{i,2}B_2 + \cdots + d_{i,l}B_l \text{mod}\, B$. Step two: the user encrypts message m_i by $c_i = g^{m_i} H_2(T_s)^{N(sk_i + \beta_i)} \bmod N^2$ and generates the signature value $\sigma_i = H_1(ID_i \| T \| c_i)^{x_i}$ with his private key x_i, where T is the current time, which can used to defend against a replay attacks. After that, the user sends the data (c_i, σ_i, T, ID_i) to the nearest FN.

5.3. Reading Phase. Firstly, FN verifies the legitimacy and integrity of the message received from the users. FN verifies that the following formula $e(\sigma_i, h) \overset{?}{=} e(H_1(ID_i \| T \| c_i), Y_i)$:

$$\begin{aligned} e\left(\sum_{i=1}^{n} \sigma_i, h\right) &= e\left(\sum_{i=1}^{n} H_1(ID_i \| T \| c_i)^{x_i}, h\right) \\ &= \sum_{i=1}^{n} e(H_1(I)D_i \| T \| c_i)^{x_i}, h) \quad . \\ &= \sum_{i=1}^{n} e(H_1(ID_i \| T \| c_i), Y_i) \end{aligned} \tag{2}$$

If the verification fails, FN asks the user to resend the data.

Secondly, after the batch verification is passed, FN aggregates the data. Due to $\sum_{i=1}^n sk_i = 0$, we have

$$\begin{aligned} c &= \prod_{i=1}^{n} c_i \\ &= g^{\sum_{i=1}^{n} m_i} H_2(T_s)^{\sum_{i=1}^{n} N(sk_i + \beta_i)} \bmod N^2 \\ &= g^M H_2(T_s)^{\sum_{i=1}^{n} N\beta_i} \bmod N^2, \end{aligned} \tag{3}$$

where $M = \sum_{i=1}^n m_i$ represents the value of all users' message aggregations. Then, FN signs the aggregated value with its private key and generates the signature value

$\sigma_f = H_1(ID_f \| T \| c)^{x_f}$. Finally, FN sends (c, σ_f, T, ID_f) to CC.

5.4. Decryption Phase. After receiving the report from FN, CC first verifies that the following formula is true for $e(\sigma_f, h) \overset{?}{=} e(H_1(ID_f \| T \| c), Y_f)$. Then, CC calculates that β_0 satisfies $\sum_{i=1}^{n} \beta_i + \beta_0 = 0 \bmod \lambda$ to decrypt the cipher c using its private key λ.

$$
\begin{aligned}
M' &= c \cdot H_2(T_s)^{N\beta_0} \bmod N^2 \\
&= g^M H_2(T_s)^{\sum_{i=1}^{n} N\beta_i} \cdot H_2(T_s)^{N\beta_0} \bmod N^2 \\
&= g^M H_2(T_s)^{\sum_{i=1}^{n} N\beta_i} \bmod N^2 \\
&= g^M \bmod N^2 \\
&= (N+1)^M \bmod N^2 \\
&= 1 + MN, \\
M &= \frac{M' - 1}{N} \\
&= \sum_{i=1}^{n} m_i \\
&= \sum_{i=1}^{n} \left(d_{i,1}B_1 + d_{i,2}B_2 + \cdots + d_{i,l}B_l\right) \bmod B \\
&= \sum_{i=1}^{n} \left(\sum_{t=1}^{l} d_{i,t} B_t \right) \bmod B \\
&= \sum_{t=1}^{l} \left(\sum_{i=1}^{n} d_{i,t} B_t \right) \bmod B \\
&= e_1 B_1 + e_2 B_2 + \cdots + e_l B_l \bmod B.
\end{aligned}
\tag{4}
$$

Finally, the Chinese remainder theorem is used to calculate $e_i = M \cdot y_i \bmod b_i, 1 \leq i \leq l$, and $\{e_1, e_2, \cdots, e_l\}$ of each dimension of all users, which can be used for statistical analysis and strategy adjustment.

5.5. Dynamic User Management. Our scheme supports dynamic user join and revoke. User failure can be considered as user revocation.

5.5.1. User Revoke. When user $u_{n'} \in \mathbb{U}$ revokes, FN broadcasts to revoke user $ID_{n'}$. Each user removes the shared key $R_{n'i}$ of the revocation user and the shared key $R_{jn'}$ sent to the user in its own key set. Then, others reupdate their set of shared keys, calculating the private key at this point in encryption. CC calculates the $\sum_{i=1}^{n} \beta_i + \beta_0' - \beta_{n'} = 0 \bmod \lambda$ update decryption blind factor β_0'.

5.5.2. User Join. When user $u_{i'}$ joins, FN broadcast adds user identity information $ID_{i'}$. Then, other users update their key sets as initialized. CC randomly generates a blind factor $\beta_{i'} \in_R Z_N^*$, sends it to the user, and updates CC's decryption blind factor β_0' according to $\sum_{i=1}^{n} \beta_i + \beta_0' + \beta_{i'} = 0 \bmod \lambda$.

6. Systems Analysis

6.1. Semantic Security of Encrypted Data. In our scheme, each user encrypts the user's electricity consumption $c_i = g^{m_i} H_2(T_s)^{N(sk_i + \beta_i)} \bmod N^2$ at T_s time and submits it to FN for data aggregation, which we demonstrate to be semantically secure with the following theorem.

Lemma 1. *For given message m_0 and m_1, encrypted ciphers are indistinguishable.*

Proof. First, we randomly select a message in messages m_0 and m_1 with $c_i = g^{m_i} H_2(T_s)^{N(sk_i + \beta_i)} \bmod N^2 (i \in \{0, 1\})$ to encrypt it. For the attacker, the advantage is $adv_A = \text{pr}|\ m_0 \in \{0,1\}^*: c_0 = Enc(m_0)| - \text{pr}|m_1 \in \{0,1\}^*: c_0 = Enc\ (m_0)|$ because the calculated $H_2(T_s)$ and $H_2(T_s) \overset{\$}{\leftarrow}$ are indistinguishable. Therefore, for the attacker, the indistinguishable ciphers $c_i (i \in \{0, 1\})$ and $c_i \overset{\$}{\leftarrow}$ are also indistinguishable. So, the attacker's advantage is $adv_A <$ negligible, the attacker cannot distinguish between encrypted messages m_0 or m_1. Thus, the encryption scheme is semantically secure. End proof. □

6.2. Privacy-Preserving. To prevent user privacy from being compromised, our proposal requires that neither FN nor CC restore user data for individual users.

Lemma 2. *Even if an external attacker steals the communication channel between the user and FN and obtains the relevant data. The attacker cannot get any information about the user's message.*

Proof. External attackers stole messages between users and FN, obtaining (c_i, σ_i, T, ID_i). As our encrypted messages are semantically secure, therefore, the attacker cannot recover messages from users without knowing the secret key sk_i and blind factor β_i. End proof. □

Lemma 3. *Even if an attacker breaks FN, no private data can be inferred from the crypto c uploaded by the user.*

Proof. The message uploaded by the user is an encrypted cipher. FN can't decrypt without the key can only aggregate. The aggregated information is as follows:

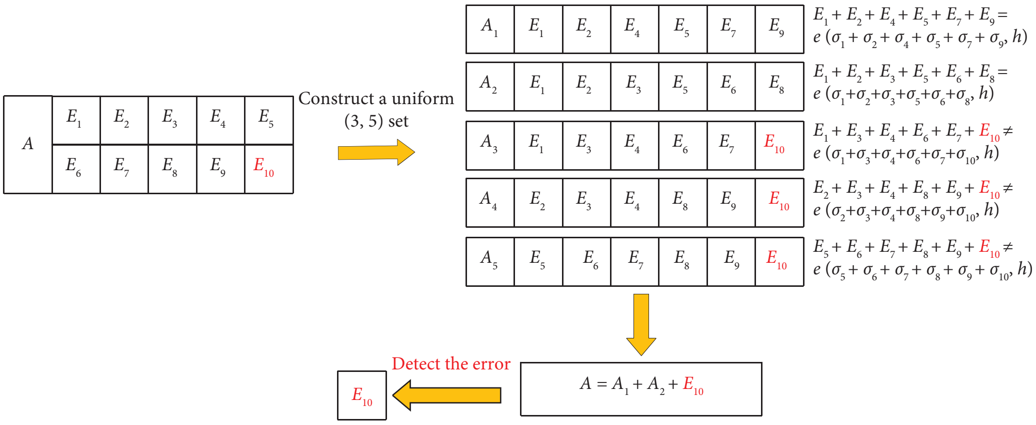

A
E_1 E_2 E_3 E_4 E_5
E_6 E_7 E_8 E_9 E_{10}
Construct a uniform (3, 5) set
A_1 E_1 E_2 E_4 E_5 E_7 E_9
$E_1+E_2+E_4+E_5+E_7+E_9 = e(\sigma_1+\sigma_2+\sigma_4+\sigma_5+\sigma_7+\sigma_9, h)$
A_2 E_1 E_2 E_3 E_5 E_6 E_8
$E_1+E_2+E_3+E_5+E_6+E_8 = e(\sigma_1+\sigma_2+\sigma_3+\sigma_5+\sigma_6+\sigma_8, h)$
A_3 E_1 E_3 E_4 E_6 E_7 E_{10}
$E_1+E_3+E_4+E_6+E_7+E_{10} \neq e(\sigma_1+\sigma_3+\sigma_4+\sigma_6+\sigma_7+\sigma_{10}, h)$
A_4 E_2 E_3 E_4 E_8 E_9 E_{10}
$E_2+E_3+E_4+E_8+E_9+E_{10} \neq e(\sigma_2+\sigma_3+\sigma_4+\sigma_8+\sigma_9+\sigma_{10}, h)$
A_5 E_5 E_6 E_7 E_8 E_9 E_{10}
$E_5+E_6+E_7+E_8+E_9+E_{10} \neq e(\sigma_5+\sigma_6+\sigma_7+\sigma_8+\sigma_9+\sigma_{10}, h)$
Detect the error
E_{10}
$A = A_1 + A_2 + E_{10}$

FIGURE 4: An example of a quick error signature detection.

TABLE 2: Functional comparison of other schemes.

Scheme	Data confidentiality	No trusted authority	Dynamic users	Multidimensional	Key and blind factor reuse
MSDA [31]	Yes	Yes	Yes	No	Yes
EPMDA-FED [18]	Yes	Yes	No	Yes	No
FPDA [11]	Yes	No	Yes	No	Yes
Our	Yes	Yes	Yes	Yes	Yes

$$c = \prod_{i=1}^{n} c_i = g^M H_2(T_s)^{\sum_{i=1}^{n} N\beta_i} \bmod N^2. \quad (5)$$

Without CC's and the private key, the attacker still cannot get any information about the user from the aggregated message. End proof. □

Lemma 4. *The attacker destroys CC to obtain CC's private key and aggregated data. The attacker cannot access any of the user's private information.*

Proof. The attacker destroyed CC and obtained CC's private key. By decrypting the data, the attacker can obtain aggregated plaintext information. But decryption messages are private to a single user's message and cannot analyze specific information about a single user. End proof. □

6.3. Data Integrity. Our scheme has the functions of authentication and data completeness check, using bilinear pairing to sign and send the signature value to FN. The attacker cannot deduce the secret key of the user's signature and forge the signature value. If forged successfully, the difficult problem of bilinear pairing is solved. However, it is impossible, so our scheme can protect the integrity of the data.

To better illustrate how to detect errors quickly, as shown in Figure 4, an example of a quick detection error signature is given. In our example, $A = \{E_1, E_2, \cdots, E_{10}\}$, $E_i = e(H_1(I\ D_i\|T\ \|c_i), Y_i)$, assuming that the wrong signature is randomly distributed and the wrong signature is σ_{10}. We construct a (3, 5) threshold set. Depending on the nature of the threshold set, we can easily identify the wrong signature σ_{10}. Compared to the 10 times it takes to detect an error signature by detecting it one by one, we only need to count up to five times to find the wrong signature σ_{10}, which improves the efficiency of the calculation. For other details, please refer to [18].

6.4. Functional Analysis. In our scheme, according to the property of paillier homomorphic encryption scheme, the ciphertext of the same plaintext is indistinguishable, so our key and blind factor can be reused. As is shown in Table 2, we compare the functionality of our scheme with that of other schemes. Our scheme is functionally superior.

7. Performance Evaluation

In this section, we evaluate our scheme in terms of computational overhead and communication overhead. Considering the computing power of smart meters and fog nodes. In literature [7, 9–11, 29–35], paillier homomorphic encryption scheme has been widely applied in data aggregation schemes of the smart grid. In literature [10, 11, 13, 29, 30, 32, 33, 35–38], fog nodes are used as intermediate nodes to process data transmitted by smart meters. Therefore, our scheme is feasible in reality. Our experiment is based on the JPBC library and uses a computer configured with an AMD R7-580H CPU@ 3.2 GHz and 16 GB of RAM with the Windows 11 operating system. We assume that there are k FNs in a smart grid system and n users in each FN. The relative parameters of the bilinear pairing are shown in Table 3, and the size of the experimental selection parameters is shown in Table 4. In order to improve the accuracy of the test, we took an average of 30 times as the test result. As is shown in Table 5, we present the computational overhead for each stage; the encryption overhead is the computational overhead of a user. As is shown in

Table 3: Parameters of the type A curve.

Name	Value
q	87807107996633125224377819847540498158068831994142082110286533992664756308802229570786251794226622214231558587695823174592777133673174813249251299982247 91
h	12016012264891146079388821366740534204802954401251311822919615131047207289359704531102844802183906537786776
r	730750818665451621361119245571504901405976559617

Table 4: Implementation parameters.

Parameter	Length (bits)
p, q	512
ID_i, ID_f	32
T, T_s	64
H_1, H_2	1024
b_i	100
$d_{i,l}$	64

Table 5: Computing overhead at each stage.

Users = 500	Stage			
Dimension = 10	Encryption	Aggregation	Batch verification	Decryption
Time (m·s)	4.37	4.50	1847	3.22
Proportion	0.24%	0.24%	99.35%	0.17%

Figure 5, we tested the computational overhead of a different number of users in the same dimension in an FN. In addition, as is shown in Figure 6, we tested the computational overhead of different dimensions for the same number of users. Since the cost of FN signature verification is primarily linear to the number of users, we will not show the cost of signature verification in Figure 6.

7.1. Computation Overhead. First of all, we mainly consider the cost of the data operation, because the cost of the system initialization stage does not affect the delay of our system. In addition, T_p to represent the computational cost of bilinear pairing. T_{m_1} to represent the multiplication of elements in group G_1. T_{m_2} to represent the multiplication of elements in group G_2. T_{m_3} to represent the multiplication of elements in group $\mathbb{Z}_{N^2}$. T_{a_1} to represent the cost of adding elements in group G_1. T_{a_2} to represent the cost of adding elements in group G_2. T_{a_3} to represent the cost of adding elements in group $\mathbb{Z}_{N^2}$. T_{h_1} represents the cost of mapping elements to group G_1 through Hash. T_{h_2} represents the cost of mapping elements to group $\mathbb{Z}_{N^2}$ through Hash. T_{e_1} to represent the cost of exponential operations in group G_1. T_{e_2} to represent the cost of exponential operations in group G_2. T_{e_3} to represent the cost of exponential operations in group $\mathbb{Z}_{N^2}$.

In our scheme, each user needs to spend $T_{e_3} + T_{h_2} + T_{m_3}$ to encrypt message, spend $T_{h_1} + T_{e_1}$ to generate his signature. So, the total computing cost of user is $T_{h_1} + T_{e_1} + T_{e_3} + T_{h_2} + T_{m_3}$. For FN, it needs to verify n user's signature in bulk, the computation cost is $nT_{a_1} + (n+1)T_p + nT_{h_1} + (n-1)T_{a_2}$, the computation cost of data aggregation is $(n-1)T_{m_3}$, and the computation cost of signatures is $T_{h_1} + T_{e_1}$, so the total computing cost of FN is $nT_{a_1} + (n+1)T_p + (n+1)\ T_{h_1} + (n-1)T_{a_2} + (n-1)T_{m_3} + T_{e_1}$. The experimental results show that dimensions have negligible impact on decryption costs. For CC, it spends $2T_p + T_{h_1}$ to authenticate the signature and spends $T_{h_2} + T_{m_3} + T_{e_3}$ to decrypt ciphertext, so the total computing cost of CC is $(2T_p + T_{h_1} + T_{h_2} + T_{m_3} + T_{e_3})k$.

In the MSDA scheme [31], each user needs to spend $3T_{e_3} + T_{h_2} + 2T_{m_3}$ to encrypt message. In the scheme, there is no signature process. Therefore, FN aggregation costs is $(n-1)T_{m_3}$. CC decryption cost is $(T_{e_3} + T_{m_3})k$.

In the EPMDA-FED scheme [18], each user needs to spend $2T_{a_3}$ to encrypt message, spend $T_{a_3} + T_{h_1} + T_{m_1}$ to generate his signature, so the total computing cost of user is $3T_{a_3} + T_{h_1} + T_{m_1}$. For FN, it needs to verify n user's signature in bulk, the computation cost is $(n-1)(T_{a_1} + T_{a_3} + T_{h_1} + T_{m_1} + T_{m_2}) + nT_p$, the computation cost of data aggregation is $(n-1)T_{a_3}$, and the computation cost of signatures is $T_{h_1} + T_{m_1}$, so the total computing cost of FN is $(n-1)(T_{a_1} + +T_{m_2}) + n(T_{h_1} + T_{m_1} + T_p) + 2(n-1)T_{a_3}$. For CC, it spends $2T_p + T_{h_1}$ to authenticate the signature and $(n-1)T_{m_3} + 2T_{a_3}$ to decrypt ciphertext, so the total computing cost of CC is $(2T_p + T_{h_1} + (n-1)T_{m_3} + 2T_{a_3})k$.

In the FPDA scheme [11], each user needs to spend $T_{h_2} + T_{e_3} + T_{m_3}$ to encrypt message. In the scheme, there is no signature process. Therefore, FN aggregation costs is $(n-1)T_{m_3}$. CC decryption cost is $(T_{h_2} + T_{e_3} + T_{m_3})k$.

As shown in Table 6, we compared the calculation costs of different schemes. It can be seen that the computation is still efficient because we include signature authentication in our scheme. But batch verification is done mainly in FN and does not impose an additional computational burden on CC decryption.

7.2. Communication Overhead. Communication overhead is calculated based on the size of the messages that the smart device sends to the fog node ($SM - to - FN$) and the fog node sends to the control center ($FN - to - CC$).

In our scheme, each SM reports the message (c_i, σ_i, T, ID_i) to FN at each interval of time, so $SM - to - FN$'s communication cost is $(|c_i| + |\sigma_i| + |T| + |ID_i|) = 3168n$. Before FN communicates with CC, FN aggregates the user's data and sends aggregated data (c, σ_f, T, ID_f) to CC. So $FN - to - CC$'s communication cost is $(|c| + |\sigma_f| + |T| + |ID_f|) = 3168k$.

In the MSDA scheme [31], each SM reports the message c_i to FN at each interval of time, so $SM - to - FN$'s communication cost is $|c_i| = 2048n$. Before FN communicates with CC, FN aggregates the user's data and sends the

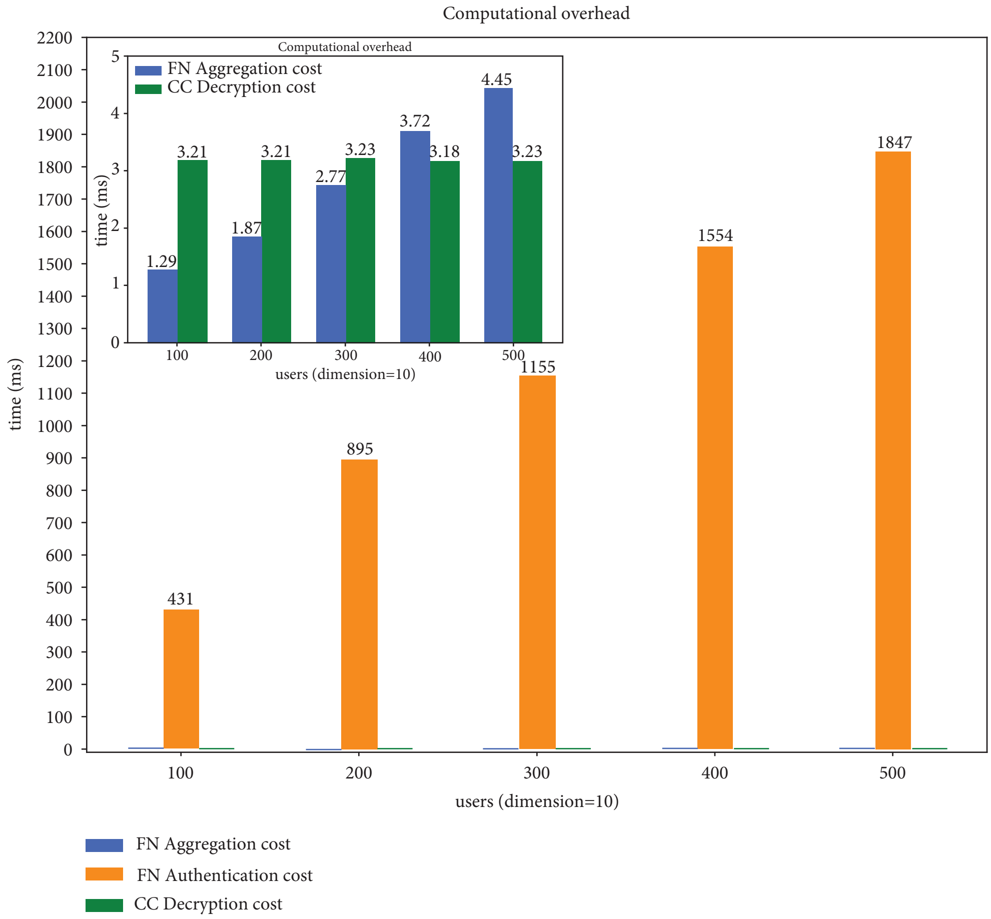


Figure 5: Calculation overhead for different user numbers in the same dimension.

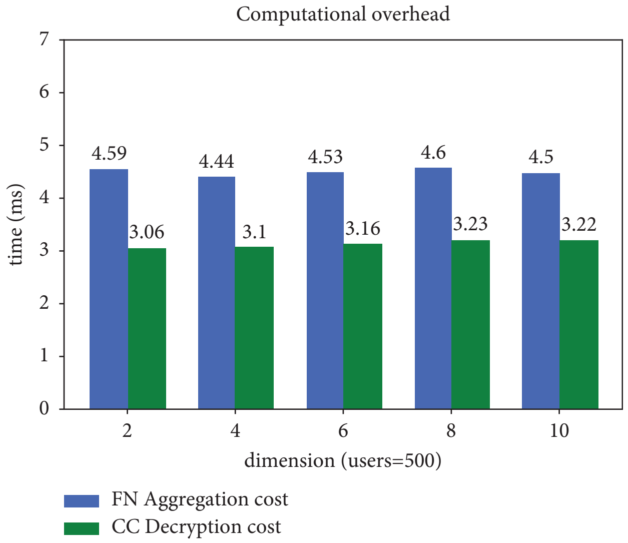


Figure 6: Calculation overhead for different dimensions of the same number of users.

aggregated data c to CC. So $FN - to - CC$'s communication cost is $|c| = 2048k$.

In the EPMDA-FED scheme [18], each SM reports the message $(c_i, \sigma_i, T, ID_i, ID_f)$ to FN at each interval of time, so $SM - to - FN$'s communication cost is $(|c_i| + |\sigma_i| + |T| + |ID_i| + |ID_f|) = 3200n$. Before FN communicates with CC, FN aggregates the user's data and sends aggregated data (c, σ_f, T, ID_f) to CC. So $FN - to - CC$'s communication cost is $(|c| + |\sigma_f| + |T| + |ID_f| + |ID_{cc}|) = 3200k$.

Table 6: Comparison of computation overhead.

Scheme	SM	FN	CC
MSDA [31]	$3T_{e_3} + T_{h_2} + 2T_{m_3}$	$(n-1)T_{m_3}$	$(T_{e_3} + T_{m_3})k$
EPMDA-FED [18]	$3T_{a_3} + T_{h_1} + T_{m_1}$	$(n-1)(T_{a_1} + +T_{m_2}) + n(T_{h_1} + T_{m_1} + T_p)$ $+ 2(n-1)T_{a_3}$	$(2T_p + T_{h_1} + (n-1)T_{m_3} + 2T_{a_3})k$
FPDA [11]	$T_{h_2} + T_{e_3} + T_{m_3}$	$(n-1)T_{m_3}$	$(T_{h_2} + T_{e_3} + T_{m_3})k$
Our	$T_{h_1} + T_{e_1} + T_{e_3} + T_{h_2} + T_{m_3}$	$nT_{a_1} + (n+1)T_p + (n+1)T_{h_1} + (n-1)T_{a_2}$ $+ (n-1)T_{m_3} + T_{e_1}$	$(2T_p + T_{h_1} + T_{h_2} + T_{m_3} + T_{e_3})k$

Table 7: Comparison of communication overhead.

Scheme	$SM - to - FN$ (n)	$FN - to - CC$ (k)
MSDA [31]	2048	2048
EPMDA-FED [18]	3200	3200
FPDA [11]	2048	2048
Our	3168	3168

In the FPDA scheme [11], each SM reports the message c_i to FN at each interval of time, so $SM - to - FN$'s communication cost is $|c_i| = 2048n$. Before FN communicates with CC, FN aggregates the user's data and sends aggregated data c to CC. So $FN - to - CC$'s communication cost is $|c| = 2048k$.

As is shown in Table 7, we compared the communication costs of other schemes. As can be seen from the table, the increase in communication cost of our scheme compared with the MSDA scheme and the FPDA scheme is mainly the communication cost of signature authentication. However, for the sake of the authenticity and integrity of the data. Therefore, our scheme's increased communication costs are necessary.

8. Conclusion

In this paper, we propose an efficient privacy-protecting multidimensional data aggregation scheme that does not require a trusted third party. The multidimensional data is encapsulated by the Chinese residual theorem and then encrypted securely, which can effectively resist the collusive attack. In addition, the scheme also supports blind factor reusability and batch data validation, so it can better adapt to the requirements of multidimensional data aggregation in complex scenarios. Experimental results show that the scheme is feasible and efficient.

Data Availability

The JPBC database used in this article is from https://gas.dia.unisa.it/projects/jpbc/#.Y4NA5DFMQ7d.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Weidong Zhong and Zichao Song contributed equally to this work.

Acknowledgments

This work was supported by the Innovative Research Team in Engineering University of PAP (KYTD201805), National Natural Science Foundation of China (Grant nos. 62172436 and 62102452).

References

[1] S. M. Muyeen and S. Rahman, *Communication, Control and Security Challenges for the Smart Grid*, Institution of Engineering and Technology, Rathipur, Odisha India, 2017.

[2] N. S. Nafi, K. Ahmed, M. A. Gregory, and M. Datta, "A survey of smart grid architectures, applications, benefits and standardization," *Journal of Network and Computer Applications*, vol. 76, pp. 23–36, 2016.

[3] S. K. Salman, "Introduction to the smart grid: concepts, technologies and evolution," The Institution of Engineering and Technology, Energy Engineering, 2017, https://www.iresearchbook.cn/f/ebook/detail?id=ec296388fec045faaec9ec0c2ec3e834.

[4] G. Dileep, "A survey on smart grid technologies and applications," *Renewable Energy*, vol. 146, pp. 2589–2625, 2020.

[5] M. Shateri, F. Messina, P. Piantanida, and F. Labeau, "Real-time privacy-preserving data release for smart meters," *IEEE Transactions on Smart Grid*, vol. 11, no. 6, pp. 5174–5183, 2020.

[6] T. W. Chim, S. M. Yiu, V. O. Li, L. C. Hui, and J. Zhong, "PRGA: privacy-preserving recording and gateway-assisted authentication of power usage information for smart grid," *IEEE Transactions on Dependable and Secure Computing*, vol. 12, no. 1, pp. 85–97, 2015.

[7] R. Lu, L Xiaohui, L Xu, L Xiaodong, and S Xuemin, "EPPA: an efficient and privacy-preserving aggregation scheme for secure smart grid communications," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 9, pp. 1621–1631, 2012.

[8] Le Chen, R. Lu, Z. Cao, K. AlHarbi, and X. Lin, "MuDA: multifunctional data aggregation in privacy-preserving smart grid communications," *Peer-to-peer networking and applications*, vol. 8, no. 5, pp. 777–792, 2015.

[9] S. Li, K. Xue, Q. Yang, and P. Hong, "PPMA: privacy-preserving multisubset data aggregation in smart grid," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 2, pp. 462–471, 2018.

[10] A. Saleem, A. Khan, S. U. R. Malik et al., "FESDA: fog-enabled secure data aggregation in smart grid IoT network," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6132–6142, 2020.

[11] L. Wu, M. Xu, S. Fu, Y. Luo, and Y. Wei, "FPDA: fault-Tolerant and privacy-enhanced data aggregation scheme in fog-assisted smart grid," *IEEE Internet of Things Journal*, vol. 9, no. 7, pp. 5254–5265, 2022.

[12] F. Knirsch, G. . Eibl, and D. Engel, "Error-resilient masking approaches for privacy preserving data aggregation," *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 3351–3361, 2018.

[13] X. Li, S. Liu, F. Wu, S. Kumari, and J. J. P. C. Rodrigues, "Privacy preserving data aggregation scheme for mobile edge computing assisted IoT applications," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4755–4763, 2019.

[14] X. Gong, S. H Qiang, Q Lixiang, Y Dongxiao, and J Hai, "Communication-efficient and privacy-preserving data aggregation without trusted authority," in *Proceedings of the IEEE INFOCOM 2018-IEEE Conference on Computer Communications*, April 2018.

[15] Y. Liu, W. Guo, C. I. Fan, L. Chang, and C. Cheng, "A practical privacy-preserving data aggregation (3PDA) scheme for smart grid," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 3, pp. 1767–1774, 2019.

[16] Y. Ding, B. Wang, Y. Wang, K. Zhang, and H. Wang, "Secure metering data aggregation with batch verification in industrial smart grid," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 10, pp. 6607–6616, 2020.

[17] Y. Chen, J. F. Martinez-Ortega, L. Lopez, H. Yu, and Z. Yang, "A dynamic membership group-based multiple-data aggregation scheme for smart grid," *IEEE Internet of Things Journal*, vol. 8, no. 15, pp. 12360–12374, 2021.

[18] Z. Liu, Z. Cao, X. Dong et al., "EPMDA-FED: efficient and privacy-preserving multidimensional data aggregation scheme with fast error detection in smart grid," *IEEE Internet of Things Journal*, vol. 9, no. 9, pp. 6922–6933, 2022.

[19] A. J. Menezes, P. C. Van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, Boca Raton, Fl, USA, 2018.

[20] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques*, Springer, Berlin, Heidelberg, May 1999.

[21] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in *Proceedings of the Annual International Cryptology Conference*, Springer, Berlin, Heidelberg, August 2001.

[22] G. Wang, Z. Cao, and X. Dong, "Improved fault-tolerant aggregate signatures," *The Computer Journal*, vol. 62, no. 4, pp. 481–489, 2019.

[23] K. Xue, B. Zhu, Q. Yang, D. S. L. Wei, and M. Guizani, "An efficient and robust data aggregation scheme without a trusted authority for smart grid," *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 1949–1959, 2020.

[24] F. Li, Bo Luo, and P. Liu, "Secure information aggregation for smart grids using homomorphic encryption," in *Proceedings of the 2010 First IEEE International Conference on Smart Grid Communications*, October 2010.

[25] O. R Merad Boudia, S. M. Senouci, M. Feham, and M. Feham, "Elliptic curve-based secure multidimensional aggregation for smart grid communications," *IEEE Sensors Journal*, vol. 17, no. 23, pp. 7750–7757, 2017.

[26] Y. Ming, X. Zhang, and X. Shen, "Efficient privacy-preserving multi-dimensional data aggregation scheme in smart grid," *IEEE Access*, vol. 7, pp. 32907–32921, 2019.

[27] X. Zuo, L. Li, H. Peng, S. Luo, and Y. Yang, "Privacy-preserving multidimensional data aggregation scheme without trusted authority in smart grid," *IEEE Systems Journal*, vol. 15, no. 1, pp. 395–406, 2021.

[28] L. Lyu, K. Nandakumar, B. Rubinstein, J. Jin, J. Bedo, and M. Palaniswami, "PPFA: privacy preserving fog-enabled aggregation in smart grid," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 8, pp. 3733–3744, 2018.

[29] J.-N. Liu, J. Weng, A. Yang, Y. Chen, and X. Lin, "Enabling efficient and privacy-preserving aggregation communication and function query for fog computing-based smart grid," *IEEE Transactions on Smart Grid*, vol. 11, no. 1, pp. 247–257, 2020.

[30] O. R. Merad-Boudia, S. M. Senouci, and S. Mohammed Senouci, "An efficient and secure multidimensional data aggregation for fog-computing-based smart grid," *IEEE Internet of Things Journal*, vol. 8, no. 8, pp. 6143–6153, 2021.

[31] Z. Zeng, X. Wang, Y. Liu, and L. Chang, "MSDA: multi-subset data aggregation scheme without trusted third party," *Frontiers of Computer Science*, vol. 16, no. 1, pp. 161808–161817, 2022.

[32] A. K. Singh and J. Kumar, "A privacy-preserving multidimensional data aggregation scheme with secure query processing for smart grid," *The Journal of Supercomputing*, pp. 1–21, 2022.

[33] Z. Guan, Y. Zhang, L. Zhu, L. Wu, and S. Yu, "EFFECT: an efficient flexible privacy-preserving data aggregation scheme with authentication in smart grid," *Science China Information Sciences*, vol. 62, no. 3, pp. 32103–32114, 2019.

[34] X. Wang, Y. Liu, and K. K. R. Choo, "Fault-tolerant multisubset aggregation scheme for smart grid," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 6, pp. 4065–4072, 2021.

[35] C. Peng, M. Luo, H. Wang, M. K. Khan, and D. He, "An efficient privacy-preserving aggregation scheme for multidimensional data in IoT," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 589–600, 2022.

[36] Z. Zeng, Y. Liu, and L. Chang, "A robust and optional privacy data aggregation scheme for fog-enhanced IoT network," *IEEE Systems Journal*, pp. 1–11, 2022.

[37] Yu Zhan, L. Zhou, B. Wang, P. Duan, and B. Zhang, "Efficient function queryable and privacy preserving data aggregation scheme in smart grid," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 12, pp. 3430–3441, 2022.

[38] R. Lu, K. Heung, A. H. Lashkari, and A. A. Ghorbani, "A lightweight privacy-preserving data aggregation scheme for fog computing-enhanced IoT," *IEEE Access*, vol. 5, pp. 3302–3312, 2017.