Multi-dimensional Data Aggregation Scheme Supporting Fault-Tolerant Mechanism in Smart Grid

Yong Chen[1], Feng Wang*[2], Li Xu[3], Zhongming Huang[4]
College of Computer Science and Mathematics, Fujian University of Technology, Fujian, China[1,2,4]
Fujian Provincial Key Laboratory of Network Security and Cryptology, Fujian Normal University, Fujian, China[2,3]

***Abstract*—With the large-scale deployment of smart grids, the scheme of smart grid data aggregation has gradually enriched in recent years. Based on the principle of protecting user privacy, existing schemes usually choose to introduce a trusted third party (TTP) to participate in the collaboration. However, this also increases the risk of privacy exposure as the attacker can target the TTP which provides services to smart grid operators. In addition, many existing schemes do not take into account the operational requirements of smart meters in case of failure. Furthermore, some schemes ignore the control center's demand for analyzing multi-dimensional data, which causes a lot of inconvenience in actual operation. Therefore, a fault-tolerant multi-dimensional data aggregation scheme is proposed in this paper. We have constructed a scheme without TTP participation in collaboration, and also meet the following two requirements. The scheme not only ensures the normal operation of the system when the smart meter fails but also meets the requirements of the control center for multi-dimensional data analysis. Security analysis shows that the proposed scheme can resist external attack, internal attack, and collusion attack. The experimental results show that the proposed scheme improves the fault tolerance and reduces the computational cost compared with the existing schemes.**

***Keywords*—Cryptography; fault tolerance; privacy; multi-dimensional data aggregation; encryption; smart grid**

I. Introduction

The smart grid [1], [2] is a modern power grid that is significantly different from the traditional power grid [3], [4]. Traditional power grids can only transmit power from power plants to users, while smart grids can communicate with other power systems through power data. Therefore, the smart grid can significantly improve the reliability, flexibility, security, efficiency and load adjustment/balancing of the power system, and has the potential to replace the traditional power grid. In addition, an important characteristics of smart grid is that real-time power data can be counted to reflect the personal behavior of power users, such as whether they are bathing, watching TV, and what electrical appliances are being used at home. However, if the user's power information plaintext is maliciously attacked during the transmission of public channels, personal information will be leaked. Therefore, it is an urgent need to protect the user's electricity information so that malicious attackers cannot get the correct information. For this reason, many scholars have introduced smart grid data aggregation schemes to protect user's privacy. However, the following problems are ignored in some schemes.

Firstly, scheme [18] indicates that schemes [17], [23] cannot resist collusion attack. Because in the above two schemes, the encryption key for smart meter users to encrypt power information is only the public key of another entity in the system, and there is no blinding factor embedded in the encryption process of the meter. On the one hand, if the aggregator (AG) colludes with the control center (CC), CC will receive the user's power consumption ciphertext sent by AG. At this time, CC is curious about the electricity data information of a user. It will decrypt the ciphertext using a private key to obtain the user's electricity plaintext, leading to the leakage of user privacy. On the other hand, if the legal person in CC is curious about the user's electricity information, he can obtain the data by eavesdropping and then use CC's decryption key to obtain the user's personal privacy information.

Secondly, schemes [15], [27] choose a trusted third party (TTP) to participate in collaboration when building a system, but this can cause fault tolerance problems or low flexible structures. In the registration phase, TTP generates public parameters and key pairs, distributing them to other entities. To resist collusion attack, TTP distributes blinding factors to each smart meter and CC. Only the correct number of meters and corresponding CC can eliminate the influence of blinding factors. This ensures that CC can only obtain the total power consumption of the entire region, rather than the power information of individual users. However, the system constructed in the above way may have the following problems after deployment. Firstly, this system can no longer add/delete smart meter after all smart meter users have registered. Secondly, if the smart meter fails, CC cannot decrypt the aggregated ciphertext.

Thirdly, smart meters in some scenarios [5], [18] can only report one-dimensional data type. However, in real life, smart meters need to report multi-dimensional data types. For example, these data can be classified by different electrical appliances (air conditioner, refrigerator, washing machine, rice cooker, etc.). By using these multi-dimensional data, the smart grid can make more efficient and reasonable power dispatching [29].

Finally, in Chen $et\ al.$'s scheme [6], data aggregation is constructed by elliptic curve cryptography (ECC), which protects users' privacy while reducing computational cost and communication cost. However, the scheme has low fault tolerance. Furthermore, the system cannot operate normally in the case of smart meter failure. In the scheme, CC decrypts the ciphertext as follows: $C_{uk} = g_x \cdot \prod_{i=1}^{n} c_{ik} = e\left(H\left(t_i\right), d_x \cdot R_A + \sum_{i=1}^{n} R_{si}\right) \cdot g^{\sum_{i=1}^{n} m_{ik}} = g^{\sum_{i=1}^{n} m_{ik}}$, where d_x is the private key of CC, R_A is the public key of all smart

*Corresponding authors

meters, and $\sum_{i=1}^{n} R_{si}$ is the decryption key of all smart meters. If CC wants to decrypt correctly, it needs to make the equation: $d_x \cdot R_A + \sum_{i=1}^{n} R_{si}=0$. Assuming there existing smart meter fails, so that it cannot report power data ciphertext in a short time. In the result, CC cannot get the equation contained in the ciphertext information $d_x \cdot R_A + \sum_{i=1}^{n} R_{si} \neq 0$. Because the encryption key R_{si} is only known by the smart meter, other entities cannot be obtained, so CC cannot make the bilinear pairing $e\left(H\left(t_i\right), d_x \cdot R_A + \sum_{i=1}^{n} R_{si}\right)=1$, and cannot get the power data $g^{\sum_{i=1}^{n} m_{ik}}$, this represents CC decryption failure.

Based on the above reasons, we propose a multi-dimensional data aggregation scheme supporting fault-tolerant mechanism in smart grid. For ease of description, the proposed scheme is referred to simply as MAFTM in the remainder of the paper. The proposed scheme not only designs fault repair mechanism to improve fault tolerance but also achieves multi-dimensional data aggregation using super incremental sequence. The main contributions of the proposed scheme are as follows.

1) Fault-tolerant mechanism: After the system completes the registration, even if the smart meter fails, the system can still work properly. By using this mechanism, the normal smart meter data is not affected by the damaged meter, and the maximum utilization of the collected data is realized.

2) Multi-dimensional data aggregation: The built system can reports multi-dimensional data types by introducing super-incremental sequences, and CC can perform mean/variance analysis on these power data to better regulate power.

3) No trusted third party (TTP): In order to avoid the adversary attacks against TTP, there is no TTP participating in the proposed scheme. In addition, there is no need to trust external entities.

4) Insider attacks resiliency: Smart meters use independent keys to encrypt power data, and CC cannot decrypt the ciphertext information of a single meter through the private key before receiving the aggregated ciphertext.

The remaining part of this paper consists of six chapters: The Section II describes in detail the research achievements of scholars in data aggregation in recent years, as well as the relevant technologies used. In the Section III, we introduced the techniques used in the solution. In the Section IV, we introduced the system model and security model of the proposed scheme. In the Section V, we detailed the overall process of the system. In the Section VI, we conducted safety analysis on the proposed scheme through four aspects. Finally, in the Section VII, we summarized this article.

II. Related Work

The smart grid has undergone many changes from its initial concept to its current widespread application. The traditional data aggregation scheme can only allow CC to get total power information for the entire HAN area, which is called one-dimensional data aggregation. When CC conducts an fine-grained analysis of one-dimensional power data, the scheme cannot meet the requirements. However, multi-dimensional data aggregation can turn various types of power data into aggregated ciphertext. CC obtains the sum of power plaintext information for a cycle period in the entire region by decrypting the aggregated ciphertext. Because the electricity from power plants cannot be easily stored, CC formulates power scheduling and regulates electricity prices based on data information from each time. Multidimensional data aggregation can better assist CC in performing the above operations and achieve more fine-grained analysis results. Therefore, many scholars have proposed data aggregation schemes.

In the research of multidimensional data aggregation in recent years, homomorphic encryption cryptosystem is a widely used privacy protection technology. Some schemes [7], [8] use homomorphic encryption technology to build systems, and use the characteristics of additive homomorphism to operate ciphertext as well as plaintext directly, but they can do better in terms of communication efficiency. The time efficiency required to calculate and receive data in a smart grid is also one of the factors we need to consider. In order to achieve more efficient computing cost and minimize communication latency as much as possible, Lu $et\ al.$ [9] constructed a more efficient aggregation scheme that can consume less system resources in terms of computational costs named EPPA in 2012. The above scheme utilizes the characteristics of super incremental sequences to construct multidimensional data, enabling CC to separate the total electricity consumption data of different sequences through algorithms when decrypting ciphertext data and utilizes the Paillier homomorphic cryptosystem [10] to encrypt power consumption data. In addition, in order to improve validation efficiency, this scheme achieves batch validation in the aggregation stage based on the Weil pairing [11] proposed. A trusted organization OA is also introduced to guide the system. In 2019, Chen $et\ al.$ [12] constructed an aggregation scheme. The scheme utilizes the Paillier homomorphic Cryptosystem to implement fine-grained data analysis requirements. In this scheme, the user can upload different types of power data through the electricity consumption values for different types of electrical appliances. In addition, CC can also perform variance analysis on multi-dimensional data. In 2019, Ming $et\ al.$ [13] considered that one-dimensional data cannot meet the requirements of power suppliers for fine-grained analysis of power data when scheduling electricity, and proposed a multidimensional aggregation scheme called P2MDA. P2MDA uses super-increasing sequence [14] and ElGamal homomorphic encryption technology to ensure user privacy while completing multi-dimensional data aggregation with less computational cost, so that smart meters can classify power consumption data based on power supply devices. The above aggregation scheme is mainly studied for multi-type data requirements and efficient computing performance. But it is worth noting that they all rely on TTP to build systems, which can provide opportunities for malicious attackers.

In addition, some scholars try to implement multi-dimensional data aggregation without using homomorphic encryption technology. Committed to accelerate the efficiency of authentication and reduce the computational cost, Boudia $et\ al.$ [15] set up an aggregation scheme based on ECC that can

transmit multiple data types in 2017. The scheme completed multidimensional data reporting without the need for pairing operations, which makes it low computational cost. So as to resist human-factor-aware differential aggregation (HDA) attack, Jia $et\ al.$ [16] proposed two different aggregation protocols, one is the basic aggregation protocol, and the other is an improved advanced aggregation protocol in 2017. In this scheme, smart meter users divide data information into M shares when uploading power consumption data. Therefore, only aggregators with the correct key can correctly aggregate power data. However, the disadvantage of this scheme is that the system cannot decrypt normally when facing the problem of meter failure in real life.

Furthermore, some scholars consider that smart meters may fail in real life. Therefore, they study how to improve the fault tolerance of aggregation schemes. Xue $et\ al.$ [17] constructed an aggregation scheme for service outsourcing called PPSO in 2019. In this scheme, CC can respond to the dynamic electricity price demand in real-time through the analysis of aggregated data. PPSO aims to improve system fault-tolerance and flexibility. Considering that smart meters may fail in real life, Wang $et\ al.$ [18] focused their attention on the fault tolerance of the system and proposed a scheme. In order to improve fault tolerance, the scheme uses Paillier homomorphic encryption without the participation of TTP and the blinding factor K negotiated among smart meters. To build a dynamic framework without TTP, Xue $et\ al.$ [19] conducted research on fault Tolerance and proposed a scheme in 2020. The scheme uses Paillier homomorphic encryption and built a dynamic secret sharing to improve fault tolerance. In the above scheme, smart meter users can ensure that the system will not collapse due to the failure of some smart meter through dynamic secret sharing, which improves the fault tolerance of the system. The disadvantage is that Xue $et\ al.$'s scheme [17] unable to defend collude attacks, while Wang $et\ al.$'s scheme [18] requires additional computational cost to negotiate and preserve the information of the blinding factor K. Moreover, Xue $et\ al.$'s scheme [19] cannot perform multi-dimensional data aggregation.

Finally, some scholars have improved the performance of data aggregation schemes by combining different technologies. Wu $et\ al.$ [20] utilized fog assistance to enhance the scheme's fault tolerance and protect user privacy in 2021. Lu $et\ al.$ [21] introduced blockchain into the smart grid in 2021, utilizing the characteristics of blockchain to improve verification efficiency. In addition, Zhang $et\ al.$ [24] implemented dual message encryption using a modified BGN homomorphic system and improved fault tolerance of the scheme using secret sharing technology in 2022. Zhao $et\ al.$ [25] designed an smart and practical aggregation scheme based on the Fog server in 2020, protecting users' privacy and security.

In summary, in recent years, research on data aggregation schemes in smart grids has focused on multidimensional data types, whether TTP participates in collaboration, and fault tolerance. But most of the schemes are based on Paillier homomorphic encryption, the communication cost is high. On the one hand, some schemes consider reporting multi-dimensional data when building systems using TTP. However, these schemes ignore the fault tolerance of the system. On the other hand, some schemes improve the fault tolerance of the system but cannot report multi-dimensional data types. In contrast, the proposed scheme uses elliptic curve cryptography [26] to construct the system, which can perform calculations more efficiently and effectively reduce computational costs. In addition, the fault-tolerant mechanism designed in this paper can ensure that CC can also obtain the power ciphertext of other normal meters when the smart meter fails.

III. Preliminaries

In this section, the related concepts used in the smart grid data aggregation scheme are mainly introduced.

A. Bilinear Pairing Map

The bilinear mapping pairing $e : G_1 \times G_1 \rightarrow G_T$ used in this paper is defined based on the elliptic curve over finite field $GF(q)$, where q is a large prime. In the above definition, where G_1 is an additive cyclic group and G_T is a multiplicative cyclic group, both G_1 and G_T of orders p. In addition, the bilinear mapping pairing $e : G_1 \times G_1 \rightarrow G_T$ also meets the following three conditions [30]

1) $Bilinearity$: For any $P, Q \in G_1$ and $x, y \in Z_p^*$, we have $e(xP, yQ) = e(P, Q)^{xy}$.
2) $Non-degeneracy$: There are two elements $P, Q \in G_1$ satisfying $e(P, Q) \neq 1_{G_T}$, where 1_{G_T} is the identity element of G_T.
3) $Computability$: For all $P, Q \in G_1$, there exists a polynomial-time algorithm to compute $e(P, Q)$.

B. Superincreasing Sequence

In practical situations, CC needs to analyze multiple data types in order to better regulate electricity. In order to achieve the above goals, one of the key technologies used in this scheme is superincreasing sequence. A superincreasing sequence consists of a series of positive real numbers $s_1, s_2, \ldots,$ And this sequence also satisfies the requirement that the newly selected elements are much larger than all previously selected elements. In addition, we can write it in this form [31]: $s_{n+1} > \sum_{j=1}^{n} s_j$.

C. Elliptic Curve Cryptography

Koblitz and Miller proposed the definition of discrete logarithm problem on a set of points of an elliptic curve in 1985. Like RSA, ECC also belongs to a type of asymmetric key mechanism. ECC is an efficient cryptosystem for resource constrained devices. This is mainly attributed to ECC's ability to achieve better security performance with smaller key size, lower power consumption, and lower computational cost compared to other algorithms such as RSA.

In addition, the mathematical base of ECC lies on the hardness of the Elliptic Curve Discrete Logarithm Problem (ECDLP). ECDLP states that if there are two points $P, Q \in E(p)$ (where $E(p)$ is an elliptic curve) then it is mathematically difficult to find an integer n such that $Q = nP$. In our proposed scheme, the hardness of ECDLP is taken into consideration [32].

IV. Models

In this section, we made assumptions about two pieces of content. In the former, we define the three major entities in the system. In the latter, we assume the trust level of entities in the system and provide a brief introduction to security performance.

A. System Model

In this paper, the system requires the collaboration of three entities to function properly, which includes: Control Center (CC), Aggregator (AG) and Smart Meter (SM), where an aggregator and several smart meters form a Home Area Network (HAN). This paper mainly studies the power consumption of all smart meters in an HAN, we use n to express the total number of smart meter, namely $SM_1, SM_2, ..., SM_n$. In addition, the image representation Fig. 1.

Control Center (CC): Throughout the entire system, CC, which is the highest management agency of the system, has powerful data analysis ability, computing power and huge storage space. CC is responsible for decrypting the power aggregate ciphertext information sent by the AG. Moreover, when the power information cannot be sent due to the fault of the meter, CC is also needed to make the system operate normally to ensure that the aggregated ciphertext information of the remaining smart meters is not affected.

Aggregator (AG): AG is the second layer in the system. Compared with CC, AG has lower computing power, lower security level and is more vulnerable to enemy attacks. During the communication process, AG will collect the power ciphertext information of all smart meters in HAN in real-time. If a smart meter fault is detected, AG will perform a fault repair mechanism. Otherwise, AG will directly aggregate the ciphertext. Finally, the aggregated ciphertext is sent to CC.

Smart Meter (SM): SM is the third layer in the system and can communicate bidirectionally with AG. In the registration phase, each SM will select a random number as its blinding factor, then uses the blinding factor and the public key of other entities to calculate the encryption key. During the communication process, SM periodically records the user's power information and encrypts it with an encryption key. Then, SM send this encrypted data to AG. In addition, SM may fail in the system, making data reporting impossible for a short time.

B. Security Model

In this paper, there are three different entities, and not all of them are fully trusted (such as some outsourced service providers), so it is necessary to define each entity. We assume that the CC and the AG entities in the system are $honest\text{-}but\text{-}curious$. This represents that CC and AG will work according to the aggregation protocol process, but they will also be curious about the uploaded data content after completing the work. In addition, SM users $SM_1, SM_2, ..., SM_n$ are $honest$. For each smart meter, they will collect data according to the process every cycle and then encrypt and upload them. They will not try to obtain the power information of other smart meter, nor will they cooperate with other entities to obtain the private data in the system.

Data transmission through insecure communication channels is vulnerable to various attacks, such as external attacks. More seriously, attackers may also steal users' power data by invading the databases of AG and CC. What needs to be ensured is that user privacy information is not stolen by malicious enemies This scheme aim to resist external attack, internal attack, and collusion attack.

V. Our Proposed Maftm Scheme

We have divided the execution process of the MAFTM scheme into the following five steps: System Initialization, Entity Registration, SM Data Reporting, AG Data Aggregation, CC Decryption Ciphertext. As shown in Fig. 2, we also provided a schematic diagram of the execution process of the Fault-Tolerant Mechanism.

A. System Initialization

At this stage, CC generates the parameters required for elliptic curve cryptography, selects a secure hash function, sets the super increment sequence, and finally CC publishes the public parameters to other entities in the system.

1) CC generates a bilinear pairing map $e : G_1 \times G_1 \to G_T$, where G_1 is an additive cyclic group, G_T is a multiplicative cyclic group and both G_1 and G_T of orders q. Then CC will select P as the random generator of G_1 and g as the random generator of G_T.
2) CC selects a secure hash function $H : \{0,1\}^* \to G_1$.
3) CC defines d as the maximum value for each data type, n as the number of smart meters SM_i, and then selects a super increasing sequence $\vec{a} = (a_1, a_2, ..., a_k)$, where $a_1, a_2, ..., a_k$ are large primes and satisfy $\sum_{j=1}^{i-1} a_j \cdot n \cdot d < a_i$, $i = 1, 2, 3, ..., k$ and $\sum_{i=1}^{k} a_i \cdot n \cdot d < q$. CC calculates $g_\phi = g^{a_\phi}$, $\phi = 1, 2, 3, ..., k$ and gets $(g_1, g_2, ..., g_k)$.
4) Finally, CC will disclose the parameter

$$pp = \{q, G_1, G_T, e, P, g, \vec{a}, H, g_1, ..., g_k\}$$

to other entities in the system.

B. Entity Registration

At this stage, each entity in the system will independently select random numbers and generate corresponding public and private key pairs, and then they will negotiate and calculate a public-private key pair for encryption/decryption of power data. Suppose the registration message is sent over a private and secure channel, which means that the adversary cannot launch an attack during the registration phase.

1) CC selects a random number $sk_x \in Z_q^*$ as the private key and calculates that the public key is $pk_x = sk_x \cdot P$, CC sends pk_x to AG.
2) Meter SM_i selects a random number $sk_i \in Z_q^*$ as the private key, calculates the public key as $pk_i = sk_i \cdot P$, SM_i sends the public key pk_i to AG.

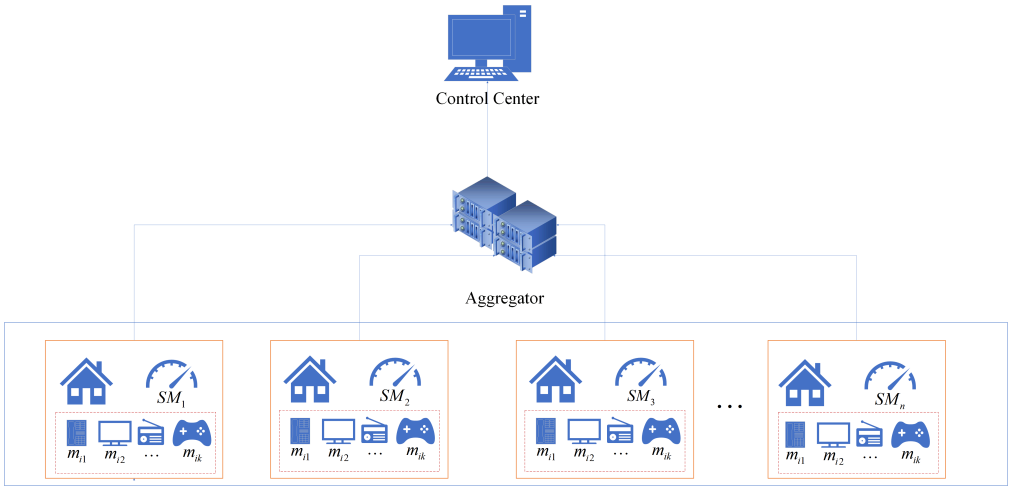

Fig. 1. System model.

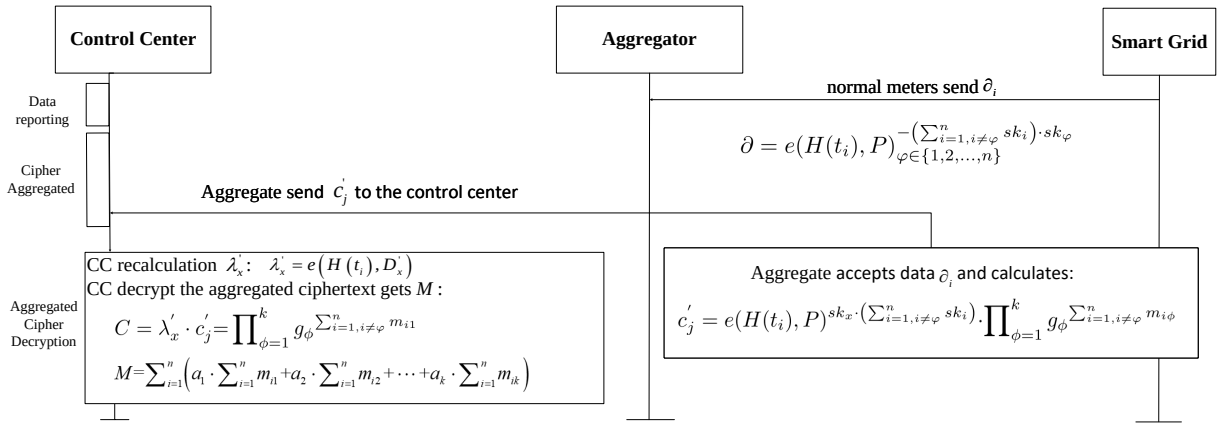

Fig. 2. Fault tolerance mechanism.

3) AG selects a random number $sk_j \in Z_q^*$ as the private key and calculates the public key as $pk_j = sk_j \cdot P$. When AG receives the public key of all smart meters, it will calculate $L_i = (\sum_{\beta=1}^{i} pk_\beta - \sum_{\beta=i+1}^{n} pk_\beta - pk_x)$ and send L_i to SM_i (i = 1,..., n). After receiving L_i, SM_i calculates the encryption key $S_i = sk_i \cdot L_i$ for encrypting real-time power data.

4) AG calculates the sum $pk_\alpha = \sum_{i=1}^{n} pk_i$ of the public keys of all SM_i and sends pk_α to CC. CC computes the decryption key $D_x = sk_x \cdot pk_\alpha = sk_x \cdot \sum_{i=1}^{n} pk_i$ to decrypt the total power data.

C. SM Data Reporting

At the beginning of each data reporting cycle, SM_i collect multiple types of data $m_{i1}, m_{i2}, m_{i3}, ..., m_{ik}$, where 1, 2, 3,..., k represent the dimension of the data type and are encrypted using the encryption key S_i. The steps are as follows.

1) Smart meter SM_i extracts data $m_{i1}, m_{i2}, ..., m_{ik}$.

2) Smart meter SM_i gets timestamp t_i.

3) Smart meter SM_i calculates $\lambda_i = e(H(t_i), S_i)$.

4) Smart meter SM_i encrypts $m_{i1}, m_{i2}, ..., m_{ik}$:

$$c_i = \lambda_i \cdot g_1^{m_{i1}} \cdot g_2^{m_{i2}} \cdot \cdots \cdot g_k^{m_{ik}} \quad (1)$$

D. AG Data Aggregation

At this stage, AG sets a counter to troubleshoot the meter. If the meter is damaged, AG will perform a fault recovery operation. Otherwise, AG aggregates the ciphertext c_i. The steps are as follows.

1) AG sets counter $count = count + 1$ to record whether there is a meter failure. If $count$=n, it represents the normal operation of all meters.

- AG calculates:

$$c_j = \prod_{i=1}^{n} c_i = \prod_{i=1}^{n} \lambda_i \cdot \prod_{\phi=1}^{k} g_\phi^{\sum_{i=1}^{n} m_{i\phi}} \quad (2)$$

- AG sends $\{c_j\}$ to CC.

2) If $count < n$, it means that the meter fails and AG performs the recovery work:

- The fault meter public key is compiled into the set $F = \{pk_\varphi\}_{\varphi\in\{1,2,...,n\}}$.
- AG informs the rest of the normal meter $\{SM_i\}_{i\neq\varphi}$ to calculate the missing data:

$$\partial_i = e(-sk_i \cdot H(t_i), pk_\varphi)_{\varphi\in\{1,2,...,n\}} \tag{3}$$

- AG receives data $\{\partial_i\}$ for data aggregation:

$$\begin{aligned}\partial &= \prod_{i=1,i\neq\varphi}^{n} \partial_i = e(\sum_{i=1,i\neq\varphi}^{n} sk_i \cdot H(t_i), -pk_\varphi) \\ &= e(H(t_i), P)_{\varphi\in\{1,2,...,n\}}^{-\left(\sum_{i=1,i\neq\varphi}^{n} sk_i\right)\cdot sk_\varphi}\end{aligned} \tag{4}$$

$$\begin{aligned}c_j^{'} &= \partial \cdot e(H(t_i), \sum_{i=1,i\neq\varphi}^{n} S_i) \cdot \prod_{\phi=1}^{k} g_\phi^{\sum_{i=1,i\neq\varphi}^{n} m_{i\phi}} \\ &= e(H(t_i), P)^{sk_x\cdot\left(\sum_{i=1,i\neq\varphi}^{n} sk_i\right)} \cdot \prod_{\phi=1}^{k} g_\phi^{\sum_{i=1,i\neq\varphi}^{n} m_{i\phi}}\end{aligned} \tag{5}$$

- Then AG sends $\left\{c_j^{'}, F\right\}$ to CC and inform the occurrence of meter damage.

E. CC Decryption Ciphertext

At this stage, CC obtains power plaintext by decrypting aggregated ciphertext. If the meter damage information sent by AG is received, CC will recalculate the decryption key $D_x^{'}$ to decrypt the aggregate ciphertext $c_j^{'}$.

1) CC computes $\lambda_x = e\left(H\left(t_i\right), S_i\right)$ under normal condition. Then the decryption operation is performed:

$$C = \lambda_x \cdot c_j = g_1^{\sum_{i=1}^{n} m_{i1}} \cdot \cdots \cdot g_k^{\sum_{i=1}^{n} m_{ik}} \tag{6}$$

The final control center gets M:

$$\begin{aligned}M &= \sum_{i=1}^{n}\left(a_1 \cdot \sum_{i=1}^{n} m_{i1} + \cdots + a_k \cdot \sum_{i=1}^{n} m_{ik}\right) \\ &= \log_g C\end{aligned} \tag{7}$$

2) If the meter damage information is received, CC recalculates the decryption key $D_x^{'}$:

$$D_x^{'} = sk_x \cdot pk_\alpha = sk_x \cdot \sum_{i=1,i\neq\varphi}^{n} pk_i \tag{8}$$

$$\lambda_x^{'} = e\left(H\left(t_i\right), S_x^{'}\right) \tag{9}$$

Then the decryption operation is performed:

$$C = \lambda_x^{'} \cdot c_j^{'} = \prod_{\phi=1}^{k} g_\phi^{\sum_{i=1,i\neq\varphi}^{n} m_{i\phi}} \tag{10}$$

Through Algorithm 1, the regional total power data of each data type is obtained: $\eta_\phi = \sum_{i=1}^{n} m_{i\phi}, \phi = 1, 2, 3, ..., k$. In addition, fault meter number is less than $n/2$, fault-tolerant mechanism can run normally.

The Algorithm 1 execution process is as follows:

Algorithm 1 Multidimensional data extraction.

Input: superincreasing sequence $\overrightarrow{a}$ and M
Output: η_ϕ **for** $\phi = 1, 2, ..., k$
Begin:
1: **Set** $X = M$
2: **for** $\phi = k$ **to** 1 **do**
3: $\eta_\phi = \frac{X-(X \bmod a_\phi)}{a_\phi}$
4: **end**
5: **return** $(\eta_1, \eta_2, ..., \eta_k)$
end

Where:

$$\begin{aligned}X = M = a_1 \sum_{i=1}^{n} m_{i1} + a_2 \sum_{i=1}^{n} m_{i2} + \cdots \\ + a_{k-1} \sum_{i=1}^{n} m_{i(k-1)} + a_k \sum_{i=1}^{n} m_{ik}\end{aligned}$$

For any data type less than constant d, we can obtain the following results:

$$\begin{aligned}a_1 \sum_{i=1}^{n} m_{i1} + a_2 \sum_{i=1}^{n} m_{i2} + \cdots + a_{k-1} \sum_{i=1}^{n} m_{i(k-1)} \\ < a_1 \sum_{i=1}^{n} d + a_2 \sum_{i=1}^{n} d + \cdots + a_{k-1} \sum_{i=1}^{n} d \\ = \sum_{j=1}^{k-1} a_j \cdot n \cdot d < a_k\end{aligned}$$

So, gets:

$$\begin{aligned}X \bmod a_k = a_1 \sum_{i=1}^{n} m_{i1} + \cdots + a_{k-1} \sum_{i=1}^{n} m_{i(k-1)} \\ \eta_k = \frac{X - (X \bmod a_k)}{a_k} = \sum_{i=1}^{n} m_{ik}\end{aligned}$$

Therefore, we can use Algorithm 1 to obtain:

$$\eta_\phi = \sum_{i=1}^{n} m_{i\phi}, \phi = 1, 2, 3, ..., k \tag{11}$$

VI. Safety Analysis

In practical applications, privacy security is one of the most concerning issues for users. We discusses the security of the system from the following four aspects: Against External Attack, Internal (AG) Attack, Collusion (AG and CC) Attack, and Fault tolerance.

A. Against External Attack

Malicious attackers will use a series of attack methods to obtain information, among which they use communication channels to steal unauthorized information, which is referred to as external attack. In this system environment, the SM encrypts the power data by the encryption key $S_i = sk_i \cdot L_i$, where $L_i = (\sum_{\beta=1}^{i} pk_\beta - \sum_{\beta=i+1}^{n} pk_\beta - pk_x)$. If the external attacker obtains the encrypted information $c_i = \lambda_i \cdot g_1^{m_{i1}} \cdot g_2^{m_{i2}} \cdot \cdots \cdot g_k^{m_{ik}}$ (where $\lambda_i = e\left(H\left(t_i\right), S_i\right)$) of the smart meter SM_i. So as to decrypt the ciphertext c_i, external attackers needs to make $e(H\left(t_i\right), P)^{sk_i\cdot\left(\sum_{\beta=1}^{i} sk_\beta - \sum_{\beta=i+1}^{n} sk_\beta - sk_x\right)} = 1$. This moment, the external attacker first needs to get the sk_i of the

SM_i, then obtain the public key $pk_i, i \in \{1, 2, ..., n\}$ of each smart meter and the public key pk_x of CC. Finally, external attackers calculate $-S_i = -L_i \cdot sk_i$ to decrypt the encrypted information c_i. However, the private key sk_i of the smart meter SM_i is only known to the entity itself, and the public keys of SM/CC are sent through private and secure channels during the registration phase, which cannot be obtained by outsiders. Therefore, external attackers cannot calculate $-S_i = -L_i \cdot sk_i$ and cannot decrypt encrypted information.

If an external attacker obtains the aggregate ciphertext $c_j = e(H(t_i), P)^{-sk_x \cdot (\sum_{i=1}^{n} sk_i)} \cdot \prod_{\phi=1}^{k} g_\phi^{\sum_{i=1}^{n} m_{i1}}$ sent by AG to CC through eavesdropping on the communication channel, they want to decrypt the aggregate ciphertext. As known from the aggregate ciphertext, an attacker who wants to decrypt data needs to make $e(H(t_i), P)^{-sk_x \cdot (\sum_{i=1}^{n} sk_i)}=1$. At this time, the attacker needs to obtain sk_x of CC and $pk_i, i \in \{1, 2, ..., n\}$ of all smart meters. However, this part of the information is also not available to other entities and external personnel. In addition, if an external attacker obtains the sk_x of CC and the $pk_i, i \in \{1, 2, ..., n\}$ of all smart meters, he can only obtain information on total electricity consumption, and unable to get the real time power information of a user through aggregated data. Through the above discussion, MFATM can effectively resist external attack initiated by malicious attackers

B. Internal (AG) Attack

Internal attackers will search for suitable devices (such as lost legitimate AG) to steal unauthorized power consumption data, a process known as internal (AG) attack. In the aggregation stage, the legitimate AG collects the ciphertext information of all smart meters. Although AG can get the power ciphertext of a user at this stage, the user's electricity usage information m_i cannot be recovered from the ciphertext $c_i = \lambda_i \cdot g_1^{m_{i1}} \cdot g_2^{m_{i2}} \cdot \cdots \cdot g_k^{m_{ik}}$, $\lambda_i = e\left(H\left(t_i\right), S_i\right)$. As in ciphertext c_i, the attackers and those who know the public key information of all entities in the system. So AG only demand to obtain the sk_i of SM_i to construct the bilinear pairing $e\left(H\left(t_i\right), \left(\sum_{\beta=1}^{i} pk_\beta - \sum_{\beta=i+1}^{n} pk_\beta - pk_x\right)\right)^{-sk_i}$. However, the sk_i is only known to the entity itself. Therefore, internal attack through aggregators cannot decrypt ciphertext data. In addition, AG aggregates the ciphertexts of n smart grid devices into a new total power data ciphertext $c_j = \prod_{i=1}^{n} c_i = e(H\left(t_i\right), (\sum_{i=1}^{n} pk_i))^{sk_x} \cdot \prod_{\phi=1}^{k} g_\phi^{\sum_{i=1}^{n} m_{i\phi}}$ in the ciphertext aggregation phase. At this point, the attacker launches an attack on the aggregated ciphertext, hoping to steal the user's electricity usage information. The attacker needs to obtain the sk_x of CC. However, the sk_x is only known to the entity itself, and other entities and outsiders cannot be obtained. Therefore, the internal attack performed by AG in aggregation phase cannot decrypt the total ciphertext data.

C. Collusion (AG and CC) Attack

Suppose AG and CC collude and share a single user's ciphertext $c_i = \lambda_i \cdot g_1^{m_i}$. However, SM uses the encryption key $S_i = sk_i \cdot L_i$ for encryption, even if CC gets L_i calculated by AG, it cannot be decrypted. Because CC cannot get the private key sk_i of the smart meter SM_i. Furthermore, CC will try to use the decryption key $D_x = sk_x \cdot pk_\alpha = sk_x \cdot \sum_{i=1}^{n} pk_i$ to decrypt the ciphertext of a single SM. However, this situation is not feasible, because the decryption key D_x is designed based on all SMs. Therefore, the decryption key D_x does not have the ability to decrypt the ciphertext of a single SM.

D. Fault Tolerance

The fault tolerance is realized, and the fault-tolerant mechanisms will not leak any useful electric data about the cooperative users. It is worth noting that, the fault means that the SM device cannot send data normally. If the CC and the AG faults occur, the SM intentionally sends false data, etc., they are not within the scope of the faults discussed in this article.

Fault-tolerance mechanisms typically include fault detection, troubleshooting, and aggregate recovery operations. We implement these functions through the following three steps:

1) AG uses a counter to detect faults;
2) When AG gets the missing data calculated by the normal SM, it cannot use the data to decrypt the user information;
3) When smart meter users in the system cannot upload power data normally due to equipment failure and other reasons, the recovery operation of the aggregator can enable normal smart meter users to calculate an aggregate value. In this case, CC can get an aggregated ciphertext, and after CC successfully decrypts it can get the power consumption data of other normal smart meter users. The maximum utilization rate of effective power data has been achieved.

In other words, even if some SM cannot work, the proposed scheme MAFTM can still restore the normal aggregation process through AG's fault tolerance mechanism, so that the power data information of other users is not affected. At the same time, because AG cannot infer the encryption key of the smart meter from the data calculated by the normal meter, it cannot be used to decrypt the user's personal information through these data, which protects the user's privacy. In fact, while implementing fault-tolerant mechanisms, we also need some additional computational cost. However, the additional computational cost itself is low, and the possibility of executing fault-tolerant mechanisms is also low (although there is a need for fault-tolerant mechanisms to exist). Considering the actual situation, we have also considered additional computational cost while designing a fault tolerance mechanism. In other words, the execution of fault-tolerant mechanisms only consumes relatively small computational resources.

VII. Performance

We will compare MAFTM scheme with some existing schemes in three aspects: Feature Comparison, Computational Cost, and Fault Folerance.

A. Feature Comparisons

Firstly, we will compare MAFTM scheme with the other eight schemes [6], [14], [15], [17], [18], [19], [21], [27] for a feature comparison (the comparison results are shown in the Table I). Chen *et al.* [6] constructed a system without a TTP based on elliptic curve cryptography, and the computational

TABLE I. FUNCTION COMPARISON OF RELATED SCHEMES

Schemes	Against External Attack	Against Internal Attack	Against Collusion Attack	Fault Tolerance	Multidi Mensional	TTP Required
Chen *et al.* [6]	Yes	Yes	Yes	No	Yes	No
Lu *et al.* [14]	Yes	Yes	No	No	Yes	Yes
Boudia *et al.* [15]	Yes	Yes	No	No	Yes	Yes
Xue *et al.* [17]	Yes	Yes	No	Yes	No	No
Wang *et al.* [18]	Yes	Yes	Yes	Yes	Yes	No
Xue *et al.* [19]	Yes	Yes	Yes	Yes	No	No
Lu *et al.* [21]	Yes	Yes	Yes	No	Yes	Yes
Wang *et al.* [27]	Yes	Yes	No	Yes	Yes	No
MAFTM	Yes	Yes	Yes	Yes	Yes	No

cost is low. However, the system cannot operate normally when the smart meter fails, and the fault tolerance is low. Moreover, the multi-dimensional data reported in the scheme is encrypted multiple times in the same form, which increased computational burden on the system. Zuo *et al.* [22] shows that Lu *et al.*'s scheme [12] unable to defend collusion attack. Boudia *et al.*'s scheme [15] uses a relatively single public and private key pair to encrypt and decrypt plaintext when building the system. If CC accidentally obtains the power data of the meter, it will cause the problem of user privacy leakage, which means that the scheme [15] can only be applied to a three-tier system, and in the Chen *et al.*'s scheme [6]. Once again, it is pointed out that [15]. cannot resist collusion attack. Xue *et al.* [17] designed a fault-tolerant mechanism to improve fault tolerance, but Wang *et al.* [18] showed that the scheme [17] cannot resist collusion attack. Wang *et al.* [18] designed a fault-tolerant mechanism to improve fault tolerance and realized multi-subset data reporting. Xue *et al.* [19] used dynamic secret sharing to improve fault tolerance but could not achieve multi-dimensional data reporting. Lu *et al.* [21] used Paillier and introduced the blockchain into the edge layer to reduce the computational pressure on the edge layer. However, the scheme has low ability to resist faults and requires TTP for collaboration. Chen *et al.* [6] shows that Wang *et al.*'s scheme [27] cannot resist collusion attack. The proposed scheme MAFTM uses ECC to build the system. Smart meters use independent keys to encrypt power data. In the data reporting phase, smart meter users use the super increment sequence to report multidimensional data types. In addition deigns a fault-tolerant mechanism to solve the problem of meter failure. Performance analysis shows that while implementing fault-tolerant mechanisms, the proposed scheme MAFTM also has some improvement in computational cost compared to existing schemes.

After completing the feature comparison, we will compare the $Computational\ Cost$ and $Fault\ Toletance$. Firstly, we choose the schemes [17], [18] with fault-tolerant mechanism. Secondly, we selected some classic data aggregation schemes [15], [27]. Therefore, in the subsequent part of the paper, we compare MAFTM scheme with the schemes [15], [17], [18], [27].

B. Computational Cost

T_e is an element exponentiation in Z_N^*, T_{mul} is an element multiplication in Z_N^*, T_b is a bilinear map pairing, T_H is a hash to an element of Z_N^*, T_{H-G} is a hash to an element of G_1, GT_e is an element exponentiation in G_T, GT_{mul} is an element multiplication in G_T. G_{mul} is an element multiplication in G_1. G_{add} is an element addition in G_1. TD_e is time of Paillier encryption operation. Compared with exponential and pairing operations, G_{add} and T_H can be ignored and their values will not be calculated in the comparison. We use the java pairing-based cryptography (JPBC) [28] library to obtain the computational time of cryptographic operations, where N is 512 bits and G is 512 bits. The operating environment for this experiment is laptop with Intel Core i7-7700HQ (2.80GHz) processor, 8GB memory and 64-bit Window10 operating system. Finally, we use n to express the number of smart meters in the experimental simulation (see Table II for details).

TABLE II. COMPUTATIONAL TIME OF DIFFERENT OPERATIONS

Operation	time(ms)
T_e	0.88
T_{mul}	0.74
T_b	6.85
TD_e	5.33
T_{H-G}	1.31
GT_e	0.64
GT_{mul}	0.51
G_{mul}	9.7

1) User's Computational Cost: We assume that there are K data types. SM costs $T_{H-G} + T_b + K(GT_{mul} + GT_e) + G_{mul}$ in the MAFTM scheme. Scheme [15], scheme [17] and scheme [27] cost $2KG_{mul} + 2G_{mul}$, $3T_e + 2T_{mul} + TD_e$ and $2GT_e + GT_{mul} + G_{mul}$, respectively. Scheme [18] requires an additional cost for constructing blind factors, totaling $3T_e + 3T_{mul} + K(T_e + T_{mul}) + 4T_{H-G} + 3T_b$.

2) AG's Computational Cost: In the MAFTM scheme and compared schemes [17], [18], [28], AG needs to execute n multiplication operations. The total computational cost is nT_{mul}. AG executes n addition operation in the scheme [15], the total cost is nG_{add}.

TABLE III. COMPUTATIONAL COSTS: ACOMPARATIVE SUMMARY

Schemes	SM	AG
MAFTM	$T_{H-G} + T_b + K(GT_{mul} + GT_e) + G_{mul}$	nGT_{mul}
Boudia[15]	$2KG_{mul} + 2G_{mul}$	nG_{add}
Xue[17]	$3T_e + 2T_{mul} + TD_e$	nGT_{mul}
Wang[18]	$3T_e + 3T_{mul} + K(T_e + T_{mul}) + 4T_{H-G} + 3T_b$	nGT_{mul}
Wang[27]	$2GT_e + GT_{mul} + G_{mul}$	nGT_{mul}

In summary, we will compare the computational costs of scheme MAFTM and scheme [15], [17], [18], [27] on the SM side and AG side. Table III lists the computational cost of each scheme. Fig. 3 shows the computational cost of the SM in the data reporting phase, Fig. 4 shows the computational cost required for data aggregation process. Furthermore, the data

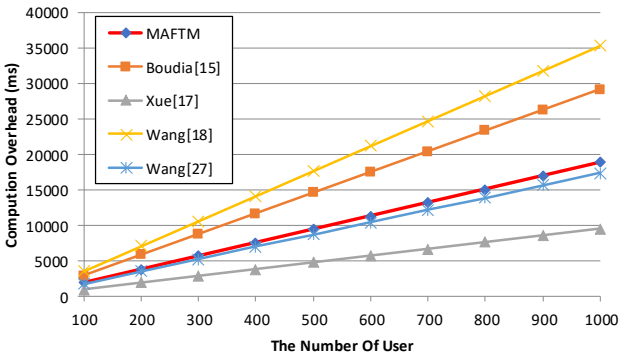

Fig. 3. Data reporting phase computational cost.

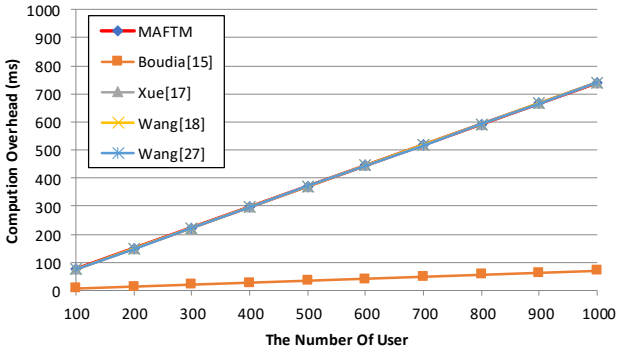

Fig. 4. The computational cost required for AG.

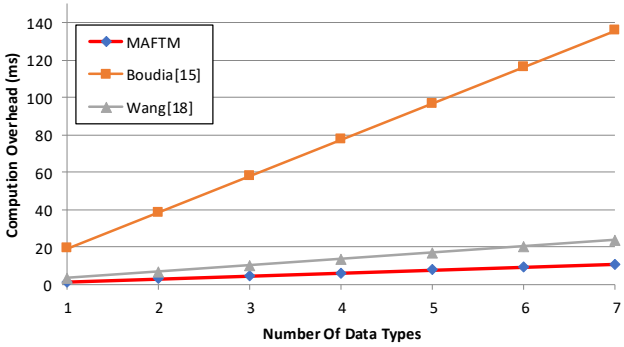

Fig. 5. Additional computational cost of data type K.

type K is 1 in the above computational cost. We compare the MAFTM scheme with schemes that implement multi-dimensional data aggregation, Fig. 5 shows the additional computational cost of data type K on the SM side.

C. Fault Toletance

If SM sends a power data ciphertext every 15 minutes, it sends an average of 17520 times a year. Assuming that every smart meter fails once in six years, we can know that 1/100000 = 0.00001 is the probability of failure through calculation. However, the failure probability of SM in daily life is far lower than 0.00001 [18]. Assuming that there are 1000 smart meters in a HAN area, it can be calculated that the minimum probability of a smart meter being damaged in six years is 0.01. Therefore, the number of times a fault-tolerant mechanism is executed is not high, but it is necessary to exist. In addition, the computational cost required to implement the fault-tolerant mechanism is affordable. When a fault-tolerant mechanism is executed, AG needs to collect computational information from a normal smart meter, where the fault computational cost for a single meter is $T_{H-G} + T_b + GT_e + (n-s)\,G_{add}$, where s is the number of SM that have failed. In addition, the formula $e(H(t_i), P)$ can be calculated in advance and each smart meter is the same, so it only needs to be calculated once. Therefore, the fault computational cost of a single meter is T_e. The additional computational cost required in the aggregation phase is $(n-s)T_{mul}$.

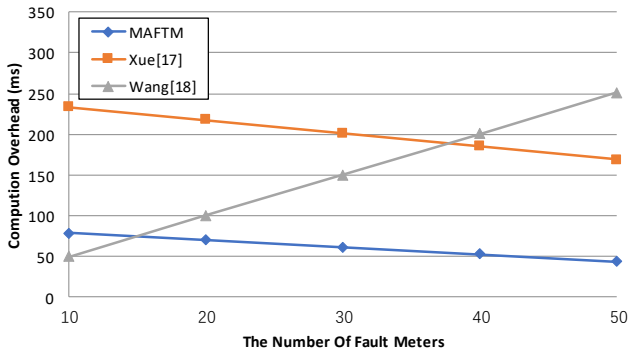

Fig. 6. Additional costs required for fault recovery(Number of faulty meters unchanged)

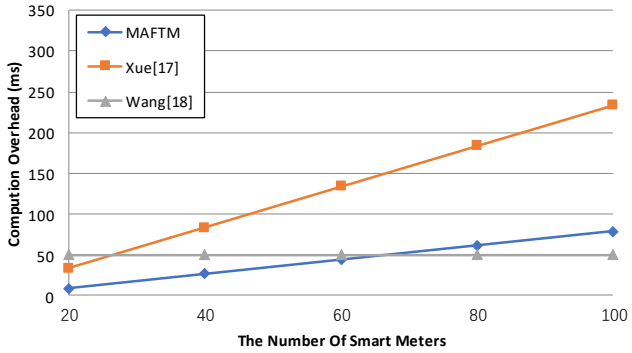

Fig. 7. Additional costs required for fault recovery(Constant number of users).

We will compare the additional Fault Tolerance cost with the schemes [17], [18]. The additional computational cost requires to start the fault-tolerant mechanism is $(s+1)T_e$+ T_H in the scheme [17]. When the fault-tolerant mechanism is executed, the additional computational cost of each SM is $4T_e$ + $2T_{mul}$ + T_H in the scheme [18]. The extra cost of AG is $(n-1)T_{mul}$. Assume that the number of meters from 20 to 100, the number of faulty meters s = 10. The computational cost required to execute the fault-tolerant mechanism is shown in Fig. 6. On the contrary, we assume that the number of faulty smart meter is 0 to 50, while the number of normal smart meter is n=100. The additional cost of the fault-tolerant mechanism is shown in Fig. 7.

VIII. Conclusion

Aiming at the problems of relying on TTP to participate in collaboration, low fault tolerance, and unable to report multi-dimensional data in the current data aggregation scheme, this paper proposes the MAFTM scheme. MAFTM is based on ECC to construct multi-dimensional data aggregation without TTP participation. At the same time, we consider that SMs may fail in real life, causing CC to fail to decrypt normally. In order to prevent the sudden failure of the SM, we also designed a fault-tolerant mechanism. In this article, we demonstrate through comparative analysis that the MAFTM scheme is more functionally complete. Furthermore, performance analysis shows that the MAFTM scheme has a lower computational cost on the SM and AG sides. Finally, the additional cost generated by implementing fault-tolerant mechanism is also lower compared to other schemes. However, the disadvantage is that the fault-tolerant mechanism proposed in this article requires more than half of the SMs to participate in the collaboration. If a large range of SMs fail, the MAFTM scheme is likely to fail to perform the recovery function properly. In addition, the aggregation of more diverse data types remains a challenging issue, such as the collection of aggregated data under multi-subset structures. We will continue to study in future work to enhance the efficient utilization of power data by control centers.

Acknowledgment

This work is supported by Natural Science Foundation of Fujian Province of China (No. 2021J011066); Science and Technology Planning Project of Fujian Province of China (No. 2021L3032 and 2022G02003); Opening Foundation of Fujian Provincial Key Laboratory of Network Security and Cryptology Research Fund, Fujian Normal University (No. NSCL-KF2021-01); Scientific Research Staring Foundation of Fujian University of Technology (No. GY-Z20171).

References

[1] Li F, Qiao W, Sun H, et al. Smart transmission grid: Vision and framework[J]. IEEE Transactions on Smart Grid, 2010, 1(2): 168-177.

[2] Niyato D, Xiao L, Wang P. Machine-to-machine communications for home energy management system in smart grid[J]. IEEE Communications Magazine, 2011, 49(4): 53-59.

[3] Fadlullah Z M, Fouda M M, Kato N, et al. Toward intelligent machine-to-machine communications in smart grid[J]. IEEE Communications Magazine, 2011, 49(4): 60-65.

[4] Liang H, Choi B J, Zhuang W, et al. Towards optimal energy store-carry-and-deliver for PHEVs via V2G system[C]//2012 Proceedings IEEE INFOCOM. IEEE, 2012: 1674-1682.

[5] Li F, Luo B, Liu P. Secure information aggregation for smart grids using homomorphic encryption[C]//2010 first IEEE International Conference on Smart Grid Communications. IEEE, 2010: 327-332.

[6] Chen Y, Martínez-Ortega J F, Castillejo P, et al. An elliptic curve-based scalable data aggregation scheme for smart grid[J]. IEEE Systems Journal, 2019, 14(2): 2066-2077.

[7] Sui Z, Niedermeier M, de Meer H. RESA: A robust and efficient secure aggregation scheme in smart grids[C]//International Conference on Critical Information Infrastructures Security. Springer, Cham, 2016: 171-182.

[8] Shen H, Zhang M, Shen J. Efficient privacy-preserving cube-data aggregation scheme for smart grids[J]. IEEE Transactions on Information Forensics and Security, 2017, 12(6): 1369-1381.

[9] Lu R, Liang X, Li X, et al. EPPA: An efficient and privacy-preserving aggregation scheme for secure smart grid communications[J]. IEEE Transactions on Parallel and Distributed Systems, 2012, 23(9): 1621-1631.

[10] Paillier P. Public-key cryptosystems based on composite degree residuosity classes[C]//Advances in Cryptology—EUROCRYPT'99: International Conference on the Theory and Application of Cryptographic Techniques Prague, Czech Republic, May 2–6, 1999 Proceedings 18. Springer Berlin Heidelberg, 1999: 223-238.

[11] Boneh D, Lynn B, Shacham H. Short signatures from the Weil pairing[C]//Advances in Cryptology—ASIACRYPT 2001: 7th International Conference on the Theory and Application of Cryptology and Information Security Gold Coast, Australia, December 9–13, 2001 Proceedings 7. Springer Berlin Heidelberg, 2001: 514-532.

[12] Chen Y, Martínez-Ortega J F, Castillejo P, et al. A homomorphic-based multiple data aggregation scheme for smart grid[J]. IEEE Sensors Journal, 2019, 19(10): 3921-3929.

[13] Ming Y, Zhang X, Shen X. Efficient privacy-preserving multi-dimensional data aggregation scheme in smart grid[J]. IEEE Access, 2019, 7: 32907-32921.

[14] Lu R, Liang X, Li X, et al. EPPA: An efficient and privacy-preserving aggregation scheme for secure smart grid communications[J]. IEEE Transactions on Parallel and Distributed Systems, 2012, 23(9): 1621-1631.

[15] Boudia O R M, Senouci S M, Feham M. Elliptic curve-based secure multidimensional aggregation for smart grid communications[J]. IEEE Sensors Journal, 2017, 17(23): 7750-7757.

[16] Jia W, Zhu H, Cao Z, et al. Human-factor-aware privacy-preserving aggregation in smart grid[J]. IEEE Systems Journal, 2013, 8(2): 598-607.

[17] Xue K, Yang Q, Li S, et al. PPSO: A privacy-preserving service outsourcing scheme for real-time pricing demand response in smart grid[J]. IEEE Internet of Things Journal, 2018, 6(2): 2486-2496.

[18] Wang X, Liu Y, Choo K K R. Fault-tolerant multisubset aggregation scheme for smart grid[J]. IEEE Transactions on Industrial Informatics, 2020, 17(6): 4065-4072.

[19] Xue K, Zhu B, Yang Q, et al. An efficient and robust data aggregation scheme without a trusted authority for smart grid[J]. IEEE Internet of Things Journal, 2019, 7(3): 1949-1959.

[20] Wu L, Xu M, Fu S, et al. FPDA: fault-tolerant and privacy-enhanced data aggregation scheme in fog-assisted smart grid[J]. IEEE Internet of Things Journal, 2021, 9(7): 5254-5265.

[21] Lu W, Ren Z, Xu J, et al. Edge blockchain assisted lightweight privacy-preserving data aggregation for smart grid[J]. IEEE Transactions on Network and Service Management, 2021, 18(2): 1246-1259.

[22] Zuo X, Li L, Peng H, et al. Privacy-preserving multidimensional data aggregation scheme without trusted authority in smart grid[J]. IEEE Systems Journal, 2020, 15(1): 395-406.

[23] Li S, Xue K, Yang Q, et al. PPMA: Privacy-preserving multisubset data aggregation in smart grid[J]. IEEE Transactions on Industrial Informatics, 2017, 14(2): 462-471.

[24] Zhang X, Huang C, Gu D, et al. Privacy-preserving statistical analysis over multi-dimensional aggregated data in edge computing-based smart grid systems[J]. Journal of Systems Architecture, 2022, 127: 102508.

[25] Zhao S, Li F, Li H, et al. Smart and practical privacy-preserving data aggregation for fog-based smart grids[J]. IEEE Transactions on Information Forensics and Security, 2020, 16: 521-536.

[26] ElGamal T. A public key cryptosystem and a signature scheme based on discrete logarithms[J]. IEEE Transactions on Information Theory, 1985, 31(4): 469-472.

[27] Wang Z. An identity-based data aggregation protocol for the smart grid[J]. IEEE Transactions on Industrial Informatics, 2017, 13(5): 2428-2435.

[28] De Caro A, Iovino V. jPBC: Java pairing based cryptography[C]//2011 IEEE Symposium on Computers and Communications (ISCC). IEEE, 2011: 850-855.

[29] Das U, Namboodiri V. A quality-aware multi-level data aggregation approach to manage smart grid AMI traffic[J]. IEEE Transactions on Parallel and Distributed Systems, 2018, 30(2): 245-256.

[30] Zhang X, Tang W, Gu D, et al. Lightweight multidimensional encrypted data aggregation scheme with fault tolerance for fog-assisted smart grids[J]. IEEE Systems Journal, 2022, 16(4): 6647-6657.

[31] Zhang X, Huang C, Zhang Y, et al. Enabling verifiable privacy-preserving multi-type data aggregation in smart grids[J]. IEEE Transactions on Dependable and Secure Computing, 2021, 19(6): 4225-4239.

[32] Sengupta A, Singh A, Kumar P, et al. A secure and improved two factor authentication scheme using elliptic curve and bilinear pairing for cyber physical systems[J]. Multimedia Tools and Applications, 2022, 81(16): 22425-22448.