Scientific Research Publishing

Journal of Computer and Communications, 2026, 14(4), 56-79
https://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

Dimension-Scalable Privacy-Preserving Data Aggregation in Edge Computing Systems

Xiao Wei

School of Mathematics and Statistics, Guilin University of Technology, Guilin, China
Email: jmx2070906206@163.com

How to cite this paper: Wei, X. (2026) Dimension-Scalable Privacy-Preserving Data Aggregation in Edge Computing Systems. *Journal of Computer and Communications*, **14**, 56-79.
https://doi.org/10.4236/jcc.2026.144003

Received: March 13, 2026
Accepted: April 18, 2026
Published: April 21, 2026

Abstract

With the rapid increase of terminal devices in the Internet of Things (IoT), it has become a significant challenge to achieve real-time and privacy-preserving data aggregation. To address this challenge, edge computing has emerged as an effective paradigm to reduce latency, where a privacy-preserving data aggregation scheme is exploited to preserve data privacy. However, most existing privacy-preserving data aggregation schemes are limited by fixed data dimensions, low scalability, and high communication or computational overhead. To address these shortcomings, this paper proposes a multidimensional privacy-preserving data aggregation scheme that supports flexible dimension expansion and privacy protection in edge computing systems. The scheme integrates the Chinese Remainder Theorem (CRT) with an elastic modulus set to efficiently pack multidimensional data. This design enables terminal devices to add new data dimensions without interrupting current operations or modifying historical data. Furthermore, by exploiting Bulletproofs-based zero-knowledge proofs and Bellare-Neven (BN) signatures with half-aggregation, the proposed scheme enables lightweight and scalable batch verification of data integrity and authenticity. These mechanisms effectively reduce the verification workload and communication bandwidth in large-scale deployments. In addition, an optimized Paillier homomorphic encryption algorithm is used to enable efficient aggregation of encrypted multidimensional data. Experimental results and theoretical analysis show that the proposed scheme significantly reduces computational and communication costs compared with existing methods.

Keywords

Data Aggregation, Edge Computing, Chinese Remainder Theorem, Dynamic Dimension Expansion

1. Introduction

As a fundamental component of the digital economy, the Internet of Things (IoT) has entered a period of rapid development. In fields such as smart healthcare [1] [2], smart grids [3] [4], and industrial automation [5], real-time data collection is enabled by the utilization of multidimensional sensors in IoT. However, conventional cloud computing models are often inadequate for real-time applications, mainly because of high transmission latency, limited bandwidth, and inherent privacy risks [6]. These challenges can be effectively addressed by the edge computing framework, which is composed of a three-layer architecture consisting of terminal devices (TDs), edge nodes (ENs), and a control center (CC) [7]. The main advantages of edge computing are low-latency response, improved bandwidth utilization, and enhanced privacy protection [8] [9].

However, due to the openness and distributed deployment of edge nodes, the risks of privacy leakage and data tampering have increased [10] [11]. For example, sensitive data may be extracted by semi-honest edge nodes through collusion, and adversaries with forged identities can inject malicious data. To achieve a balance between data availability and privacy protection, researchers have proposed single-dimension privacy-preserving data aggregation (PDA) schemes [12]-[14] and multidimensional privacy-preserving data aggregation (MPDA) schemes [15]-[19]. Most existing schemes utilize homomorphic encryption [20] or the Chinese Remainder Theorem (CRT) [21]-[23] to achieve ciphertext aggregation. However, several challenges remain, such as limited dynamic scalability, low computational efficiency, and high communication overhead [24]. These schemes are generally designed for fixed data dimensions. As a result, the addition of a new dimension typically requires the reconstruction of global parameters, such as the modulus set, which results in considerable computational and communication overhead.

To address these challenges, this paper proposes a multidimensional privacy-preserving data aggregation scheme with dynamic dimension scalability. Specifically, the proposed scheme exploits CRT-based packing and an elastic modulus set to achieve flexible dimension expansion. To ensure lightweight verification and efficient aggregation, the proposed scheme integrates Bulletproofs-based zero-knowledge proofs, Bellare-Neven (BN) signatures with half-aggregation, and an optimized Paillier encryption algorithm. Based on these mechanisms, the main contributions of this work are summarized as follows.

1. A dynamic multidimensional data extension mechanism is proposed by introducing an elastic modulus set for CRT. In the proposed scheme, when the data dimension needs to be extended, ongoing operations are not interrupted because redundant coprime primes are preallocated. Therefore, when a new dimension needs to be added, appropriate primes can be directly selected from the modulus set, which eliminates the requirement to reconstruct historical data packings. Furthermore, the coprime property of CRT ensures unique decoding and constrains the data size, thereby preventing overflow.

2. To ensure data integrity and authenticity, Bulletproofs-based zero-knowledge

proofs are integrated with BN signatures to provide multi-layered verification. In the proposed scheme, half-aggregation is exploited to reduce communication and computational overhead at the edge. Furthermore, the BN signature mechanism naturally binds the public key into the challenge hashing. As a result, robustness against false data injection, rogue-key attacks, and insider privilege abuse is ensured, even under a stronger adversarial model.

3. Efficient and scalable system-level optimizations are achieved through a hierarchical batch processing mechanism and a fault rollback strategy. In the event of batch verification failure, a grouped rollback procedure, such as dividing data into groups of ten, is triggered to quickly identify and eliminate invalid data, thereby ensuring processing efficiency and system stability.

4. Experimental results and theoretical analysis demonstrate that the proposed scheme significantly reduces both computational and communication costs compared with existing methods, thereby improving scalability and practicality in large-scale IoT deployments.

The remainder of this paper is organized as follows. Section II reviews related work. Section III describes the system model, adversary model, and design objectives. Section IV outlines the necessary preliminaries. Section V provides a detailed description of the proposed scheme. Sections VI and VII present the security analysis and performance evaluation, respectively. Finally, Section VIII concludes the paper.

2. Related Works

Privacy-preserving data aggregation plays a vital role in IoT, since it enables efficient data collection while safeguarding user privacy. Extensive research has been devoted to both single-dimensional and multidimensional aggregation schemes, with the goal of achieving a balance among security, efficiency, and flexibility. This section provides a concise overview of several representative privacy-preserving data aggregation approaches.

In the field of single-dimensional data aggregation, several core schemes have been proposed. Fan *et al.* [13] proposed a scheme in which an offline trusted third party distributes blinding factors to each smart meter. The scheme combines BLS short signatures with batch verification to ensure that the aggregator cannot access individual data. However, this scheme lacks fault tolerance and depends on the trusted third party to manage the blinding factors. To overcome these limitations, Liu *et al.* [25] proposed the (k,n)-PDA scheme, which uses Shamir's (k,n) secret sharing to distribute the decryption key without requiring an online trusted authority. This approach, however, introduces high computational overhead and relies on the cloud server. To achieve real-time and low-latency aggregation, Guo *et al.* [26] proposed a fog-based edge architecture, in which efficient symmetric homomorphic encryption replaces public-key operations at the smart meter side. This design enables a lightweight scheme that supports ciphertext-domain aggregation in real-time without relying on any online trusted party. Nevertheless, these approaches are limited to single-dimensional aggregation, which restricts fine-grained

data analysis and high-frequency reporting in IoT applications.

To overcome the limitations of single-dimensional aggregation, researchers have proposed multidimensional privacy-preserving data aggregation (MPDA) schemes. Early schemes, such as the EPPA proposed by Lu *et al.* [15], exploited Paillier homomorphic encryption and super-increasing sequences to achieve multidimensional aggregation. However, these schemes generally suffered from cross-dimensional interference, data overflow risks, high computational costs, and limited support for non-linear aggregation functions. To address these issues, Chen *et al.* [27] improved the encoding strategy, which allowed a single ciphertext to represent multiple data types and support variance and one-way ANOVA analysis. Liu *et al.* [28] introduced a dual-trapdoor encryption mechanism in a fog computing environment to balance data controllability and functionality, although the system complexity remains high. Boudia *et al.* [19] proposed a lightweight aggregation scheme based on elliptic curve cryptography, avoiding expensive bilinear pairing operations. Subsequently, developed under a fog computing architecture, the ESMA scheme [29] further enhanced multidimensional data encoding and fault tolerance. More recently, Peng *et al.* [30] proposed a CRT-based packing method with a counter to prevent data overflow and support complex functions. Xu *et al.* [31] combined CRT with symmetric encryption in an edge computing environment, improving scalability and fault tolerance. Moreover, their schemes leveraged the distributed computing capabilities of edge nodes to achieve low-latency and real-time data aggregation.

In the field of PDA, studies have focused on enhancing data integrity and verifiability. To this end, digital signatures or lightweight verification mechanisms have been integrated into aggregation frameworks to enable result verification while preserving data privacy. Boschini *et al.* [32] proposed a progressive and efficient verification framework that supports partial verification even under interruptions. Similarly, Zhang *et al.* [33] designed a post-quantum identity-based signature scheme for IoT networks, which enhances both signature security and authentication performance. To extend these algorithmic improvements to system-level applications, Shrivastava *et al.* [34] combined quantum-resistant signatures with blockchain to ensure data integrity in cloud and IoE systems. However, most of these methods still rely on static signature models and lack flexible coordination among multiple terminals. Consequently, balancing verification efficiency and system robustness under strong adversarial conditions remains challenging for these methods.

The above-mentioned existing studies have advanced along three primary directions: functional extension, efficiency optimization, and architectural evolution. Although significant progress has been made in security and efficiency, three common limitations remain: i) the data dimensions are fixed, making dynamic expansion difficult; ii) static signature schemes cannot withstand attacks under strong adversarial conditions; and iii) the lack of system-level fault recovery and rollback mechanisms hinders reliability.

3. System Design

3.1. System Model

Our system model consists of three types of entities: CC, ENs, and TDs, as illustrated in Figure 1. In this architecture, TDs collect and encrypt data, and then transmit the resulting ciphertexts to ENs. ENs aggregate the ciphertexts, and the CC performs the decryption. The detailed responsibilities of each entity are described as follows.

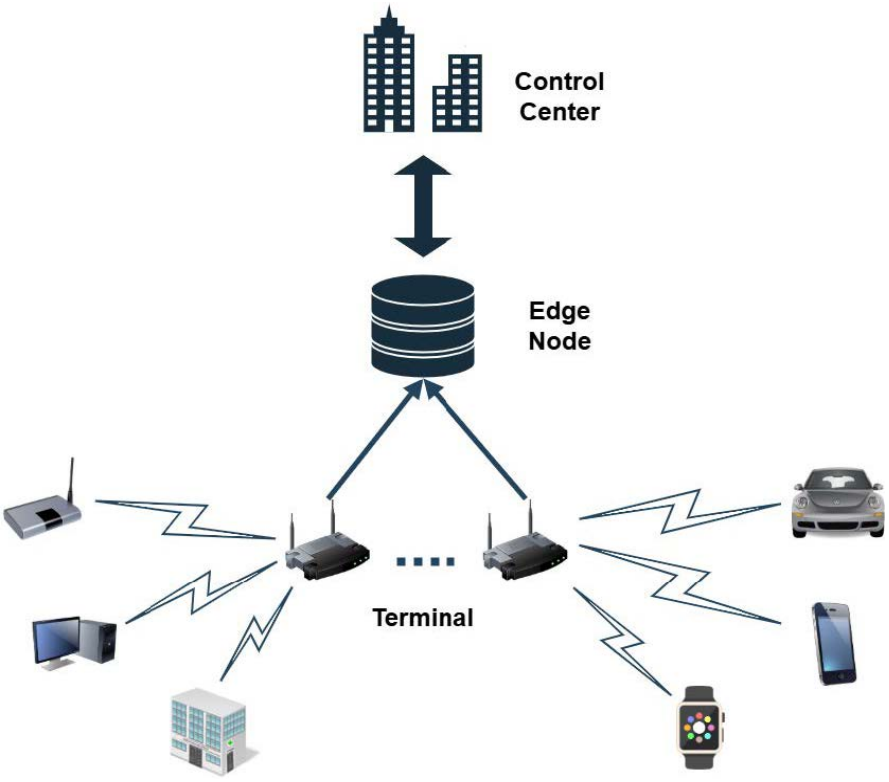

Figure 1. System model.

- **Control Center (CC):** The CC performs system initialization, including the generation of cryptographic key pairs and the construction of the modulus set. It decrypts the aggregated ciphertexts and manages the process of dynamic data dimension extension. Furthermore, the CC oversees secure key distribution and conducts regular key updates to ensure system security and scalability.
- **Terminal Devices (TDs):** TDs collect raw data using multidimensional sensors and pack the collected data. They utilize the Paillier homomorphic encryption algorithm to ensure data privacy. Furthermore, TDs generate zero-knowledge proofs and digital signatures over the encrypted data to ensure data authenticity and integrity.
- **Edge Nodes (ENs):** ENs aggregate the received ciphertexts and verify both the digital signatures and zero-knowledge proofs. Upon successful verification, they forward the aggregated results to the CC, ensuring the confidentiality and integrity of the data processing chain.

To ensure the deterministic execution of the non-interactive BN signature and

batch verification, we assume a time-slotted synchronization mechanism for the aggregation epochs. At the beginning of each epoch, the serving EN determines an active terminal list L according to the currently registered and reachable TDs in its coverage domain, and broadcasts L together with the epoch/version information and an authentication tag (or digital signature) to all participating TDs. Before computing the BN challenge, each TD verifies the authenticity and freshness of the received list. Consequently, all TDs in the same epoch compute their cryptographic challenges based on the same trusted and globally synchronized L. We assume the network topology remains quasi-static within the short aggregation window. If any terminal disconnects or fails to upload its ciphertext, the EN detects the state mismatch during batch verification and deterministically triggers the recursive group rollback mechanism to isolate the missing or invalid proofs. Severe synchronization failures within an epoch may lead to challenge inconsistency and packet rejection. Likewise, if a large number of TDs drop out during the same epoch, the aggregation process may experience increased verification overhead and reduced service availability.

3.2. Threat Model

In our security model, the CC and ENs are regarded as semi-honest (*i.e.*, honest-but-curious) entities. Specifically, they adhere to the prescribed protocol, where ENs perform batch verification and ciphertext aggregation, and the CC decrypts the results. However, they may attempt to infer the private multidimensional data of individual TDs from the processed ciphertexts. We assume that communication between the CC and other entities occurs over secure channels and that no collusion occurs between the CC and ENs, or between ENs and TDs. Therefore, collusion between the CC and the serving EN is not considered in the present threat model.

While the majority of TDs are benign, we assume the existence of a Probabilistic Polynomial-time (PPT) adversary $\mathcal{A}$. $\mathcal{A}$ may act as an external attacker or compromise a fraction of the TDs. Specifically, $\mathcal{A}$ is capable of launching the following attacks.

- **Privacy Violation:** $\mathcal{A}$ may eavesdrop on public communication channels to intercept ciphertexts, attempting to distinguish or recover the underlying multidimensional plaintext data without the master private key.
- **Data Forgery and Impersonation:** $\mathcal{A}$ may attempt to forge digital signatures to impersonate legitimate TDs or tamper with the aggregated ciphertexts and signatures during transmission.
- **Data Pollution:** Through compromised TDs, $\mathcal{A}$ may attempt to forge zero-knowledge proofs to inject invalid, out-of-range data into the aggregation pool, bypassing range verification and distorting the final statistical results at the CC.

3.3. Design Goals

Guided by the system and threat models, we design the proposed scheme to satisfy the following core requirements.

- **Dynamic Dimension Scalability.** The scheme must support the on-demand addition of data dimensions, ensuring uninterrupted system services through flexible modulus set management.
- **Data Authenticity and Integrity.** Lightweight BN signatures must be incorporated to enable the real-time verification of data integrity, preventing data tampering and impersonation attacks.
- **Privacy-Preserving Aggregation.** The scheme must prevent the disclosure of raw data during aggregation by exploiting Paillier homomorphic encryption, complemented by Bulletproofs to verify data compliance.
- **Efficient Data Processing.** The scheme must achieve high computational and communication efficiency. Specifically, CRT-based packing reduces the encryption burden on TDs, while BN half-aggregation and batch verification minimize communication overhead at the ENs, supporting concurrent access from numerous TDs.
- **Robustness against Malicious Entities.** The scheme must resist inference attacks from semi-honest ENs and active attacks from compromised TDs. The integration of BN signatures prevents rogue-key attacks during aggregation, while Bulletproofs block the injection of invalid data, ensuring robust security under strong adversarial conditions.

4. Preliminaries

In this section, we outline the cryptographic assumptions and primitives underlying our proposed scheme.

4.1. Cryptographic Hardness Assumptions

The formal security proofs of our scheme rely on two standard cryptographic hardness assumptions.

1) Decisional Composite Residuosity (DCR) Assumption: Let $N = pq$ be a Paillier modulus, where p and q are two large prime numbers of equal bit-length. The DCR problem states that given a random element $z \in \mathbb{Z}_{N^2}^*$, it is computationally intractable for any probabilistic polynomial-time (PPT) algorithm to determine whether z is an N-th residue modulo N^2. Specifically, it is computationally hard to distinguish whether there exists an $x \in \mathbb{Z}_{N^2}^*$ such that $z \equiv x^N \left(\mathrm{mod}\, N^2\right)$ or if z is uniformly distributed in $\mathbb{Z}_{N^2}^*$.

2) Elliptic Curve Discrete Logarithm Problem (ECDLP): Let $\mathbb{G}$ be a cyclic elliptic curve group of prime order l, and let G be a generator of $\mathbb{G}$. Given a tuple (G, Y), where $Y = xG$ for a randomly chosen secret scalar $x \in \mathbb{Z}_l^*$, the ECDLP asserts that extracting x is computationally infeasible for any PPT adversary.

4.2. Paillier Cryptosystem and CRT Packing

To enable privacy-preserving data aggregation at the edge, our scheme utilizes the Paillier cryptosystem integrated with the Chinese Remainder Theorem (CRT).

1) Paillier Cryptosystem: Paillier [35] is an asymmetric encryption scheme featuring an additive homomorphic property. To optimize computational efficiency, our scheme adopts the variant where the generator is fixed as $g = N+1$.

- *Key Generation:* Compute the modulus $N = pq$ and the Carmichael function $\lambda = \mathrm{lcm}(p-1, q-1)$. The decryption parameter is $\mu = \lambda^{-1} \pmod N$. The public key is N, and the private key is (λ, μ).
- *Encryption:* To encrypt a message $m \in \mathbb{Z}_N$, choose a random $r \in \mathbb{Z}_N^*$ and compute the ciphertext as $c = (1 + m \cdot N) \cdot r^N \pmod{N^2}$.
- *Decryption:* Given c, the plaintext is recovered as $m = L\left(c^\lambda \pmod{N^2}\right) \cdot \mu \pmod N$, where $L(x) = \dfrac{x-1}{N}$.
- *Additive Homomorphism:* For ciphertexts $c_1 = \mathrm{Enc}(m_1)$ and $c_2 = \mathrm{Enc}(m_2)$, the product of ciphertexts decrypts to the sum of their plaintexts: $c_1 \cdot c_2 \pmod{N^2} = \mathrm{Enc}(m_1 + m_2 \pmod N)$.

2) CRT Packing Mechanism: The CRT provides an isomorphic mapping to pack multidimensional small integers into a single large integer. Let $\Psi = \{\psi_1, \cdots, \psi_k\}$ be a set of pairwise coprime moduli, and let $M = \prod_{i=1}^k \psi_i$. A multidimensional data vector $(d_1, \cdots, d_k)$, where each $d_i < \psi_i$, can be uniquely packed into a single integer $D \in \mathbb{Z}_M$ as

$$D = \sum_{i=1}^{k} d_i \cdot M_i \cdot M_i' \pmod M, \tag{1}$$

where $M_i = M/\psi_i$, and M_i' is the modular inverse of M_i modulo ψ_i. After homomorphically aggregating the packed integers, the aggregated sum for each dimension i can be extracted using the modulo operation $D_{agg} \pmod{\psi_i}$, provided that the aggregated plaintext does not exceed the Paillier modulus N.

4.3. Bellare-Neven Signatures and Half-Aggregation

To ensure data authenticity while minimizing communication overhead, we employ the BN signature mechanism. Unlike standard Schnorr multi-signatures that require interactive rounds, the BN scheme allows a verifier to non-interactively compress multiple independent signatures into a single compact representation.

Let H be a cryptographic hash function. A terminal device generates a secret key $sk \in \mathbb{Z}_l^*$ and a public key $pk = sk \cdot G$. To sign a message M, the device chooses a random nonce $r \in \mathbb{Z}_l^*$, computes $R = r \cdot G$, and derives the challenge $c = H(L \,\|\, pk \,\|\, R \,\|\, M)$, where L is the list of participating public keys. The signature scalar is $s = r + c \cdot sk \pmod l$, yielding the signature $\sigma = (R, s)$.

Given n signatures $\sigma_i = (R_i, s_i)$ from n distinct devices, the half-aggregation technique compresses the scalars into $s_{agg} = \sum_{i=1}^n s_i \pmod l$. The aggregate signature becomes $(R_1, \cdots, R_n, s_{agg})$, reducing bandwidth requirements by approximately half. Because the distinct public key pk_i is hashed into the challenge c_i, this mechanism prevents rogue-key attacks, in which an adversary crafts

public keys to cancel out legitimate signatures.

4.4. Pedersen Commitments and Bulletproofs

To prevent data pollution attacks without compromising privacy, our scheme integrates Pedersen commitments with Bulletproofs.

1) Pedersen Commitment: The Pedersen commitment is a homomorphic commitment scheme providing perfectly hiding and computationally binding properties under the ECDLP. For a data value d and a random blinding factor $r \in \mathbb{Z}_l^*$, the commitment is defined as $com = d \cdot G + r \cdot H$, where G and H are independent generators of $\mathbb{G}$. It satisfies additive homomorphism: $com_1 + com_2 = (d_1 + d_2)G + (r_1 + r_2)H$.

2) Bulletproofs Range Proof: Bulletproofs is a non-interactive zero-knowledge proof protocol that requires no trusted setup [36]. In our scheme, for the multidimensional data vector of a terminal device TD_i, we use com_i to denote the corresponding commitment object, which may be instantiated as a vector commitment or an equivalent set of per-dimension commitments. Based on com_i, TD_i generates a proof π_i to convince the ENs that each committed dimension lies within its valid range $[0, \Gamma_j]$ without revealing the plaintext values. The verifier checks π_i with respect to the corresponding com_i, so that the multidimensional range proof is explicitly bound to the same committed data instance.

5. Our Proposed Scheme

Our scheme includes four phases: system initialization, data collection, data aggregation, and data decryption. The symbols used in our scheme are defined in Table 1.

Table 1. Symbol definitions.

Symbol	Description
$N = pq$	Paillier modulus (p, q are large secure primes)
$g = N + 1$	Generator for the Paillier cryptosystem
λ, μ	Carmichael function $\lambda = \text{lcm}(p-1, q-1)$, and decryption parameter $\mu = \lambda^{-1} (\text{mod } N)$
l	Prime order of the elliptic curve group $\mathbb{G}$
Ψ, ψ_j	System modulus set, and j -th dimensional coprime modulus
Γ_j	Maximum legitimate boundary value for the j -th dimension
T_{cap}	Secure capacity threshold to prevent plaintext overflow
$n_{\max}$	Maximum physical capacity of TDs connected to a single EN
d_{ij}	The plaintext data of the j -th dimension for terminal TD_i
$S^{(k)}, M^{(k)}$	The k -th chunk of active moduli and its corresponding product
(sk_i, pk_i)	Digital signature private and public key pair of TD_i
L	List of public keys from all participating terminals
com_i, π_i	Commitment object for the multidimensional data of TD_i, and the corresponding Bulletproofs range proof
$C_i^{(k)}, C_{agg}^{(k)}$	Individual and aggregated ciphertext for the k -th chunk
σ_i, σ_{agg}	Bellare-Neven signature and aggregated half-signature

5.1. System Initialization

The CC initializes the system and publishes the public parameters as follows.

- *Step 1:* Let $\mathbb{G}$ be an elliptic curve group of prime order l with generators G and H. The CC randomly selects two large prime numbers $p,q \leftarrow \text{GenPrime}(\kappa)$, computes the Paillier modulus $N = pq$ and the Carmichael function $\lambda = \text{lcm}(p-1, q-1)$, and sets $g = N+1$ as the generator for the Paillier cryptosystem. Additionally, the CC computes the decryption parameter $\mu = \lambda^{-1} \pmod N$.
- *Step 2:* The CC initializes the modulus set by selecting $K_{\max} + 20\%$ extra prime numbers using a prime generation function, denoted as the set $\Psi = \{\psi_1, \cdots, \psi_K\}$. These primes will be used for CRT packing.
- *Step 3:* Each terminal device TD_i independently generates a key pair (sk_i, pk_i) for the digital signature algorithm to ensure data authenticity and non-repudiation. The secret key sk_i is kept local and is never shared.
- *Step 4:* To prevent plaintext overflow during homomorphic aggregation, the CC establishes a packing capacity boundary. Let $n_{\max}$ denote the maximum number of terminal devices connected to a single Edge Node (EN). The CC computes the secure capacity threshold T_{cap} as

$$T_{cap} = \left\lfloor \frac{N}{n_{\max}} \right\rfloor. \tag{2}$$

This threshold guarantees that the aggregated sum of any valid packed data chunk does not exceed the Paillier modulus N, thereby preventing modular truncation during decryption.

- *Step 5:* To guarantee the state machine consistency and forward compatibility of the dynamic aggregation protocol, the CC initializes the global system epoch with a genesis version tag $V_{id}^{(0)}$. Consequently, the CC broadcasts the complete public parameters to all ENs and TDs according to Equation (3).

$$PP = \left\{ N, g, \Psi, \mathbb{G}, l, G, H, T_{cap}, V_{id}^{(0)} \right\} \tag{3}$$

5.2. Data Collection

Each terminal device TD_i performs the following operations on its multidimensional data vector $d_i = (d_{i1}, \cdots, d_{im})$.

- *Step 1:* TD_i selects m prime numbers from the modulus set Ψ to form the active set corresponding to its current data dimensions, denoted by $S_{active} = \{\psi_{s_1}, \cdots, \psi_{s_m}\}$. To prevent plaintext overflow caused by a single large modulus product, TD_i performs an adaptive capacity check using T_{cap}. TD_i partitions S_{active} into B chunks, denoted by $\{S^{(1)}, \cdots, S^{(B)}\}$, using a greedy algorithm. For any chunk $k \in \{1, \cdots, B\}$, the product of its moduli must satisfy the capacity threshold:

$$M^{(k)} = \prod_{\psi \in S^{(k)}} \psi \le T_{cap}. \tag{4}$$

- *Step 2:* For each chunk k, TD_i computes the specific CRT packing coefficients. For a given prime $\psi_{s_j} \in S^{(k)}$, the coefficients are calculated as

$$M_j^{(k)} = \frac{M^{(k)}}{\psi_{s_j}}, \quad \left(M_j^{(k)}\right)' \equiv \left(M_j^{(k)}\right)^{-1} \left(\mathrm{mod}\, \psi_{s_j}\right). \tag{5}$$

- *Step 3:* TD_i generates the data commitment $com_i = \mathrm{commit}\left(d_{i1}, \cdots, d_{im}\right)$, where $d_{ij} \le \Gamma_j < \psi_{s_j}/2$. Then, it uses the Bulletproofs protocol to generate a zero-knowledge range proof covering all dimensions, computed as

$$\pi_i = \mathrm{Bulletproofs.Prove}\left(d_{i1}, \cdots, d_{im}, \Gamma_1, \cdots, \Gamma_m, com_i\right). \tag{6}$$

- *Step 4:* Rather than computing a single integer, TD_i constructs a set of packed values. For each chunk $k \in \{1, \cdots, B\}$, the packed value is computed as

$$D_i^{(k)} = \sum_{\psi_{s_j} \in S^{(k)}} d_{ij} \cdot M_j^{(k)} \cdot \left(M_j^{(k)}\right)' \left(\mathrm{mod}\, M^{(k)}\right). \tag{7}$$

- *Step 5:* TD_i encrypts each packed chunk independently using the optimized Paillier encryption algorithm. For each $k \in \{1, \cdots, B\}$, the ciphertext is generated as

$$C_i^{(k)} = \left(1 + D_i^{(k)} \cdot N\right) \cdot \left(r_i^{(k)}\right)^N \left(\mathrm{mod}\, N^2\right), \tag{8}$$

where $r_i^{(k)} \leftarrow \mathbb{Z}_N^*$. Consequently, the terminal generates a ciphertext vector $\boldsymbol{C}_i = \left\{C_i^{(1)}, \cdots, C_i^{(B)}\right\}$.

- *Step 6:* To ensure data authenticity and prevent forgery without requiring online interaction, TD_i applies the BN signature mechanism. It generates a random nonce $r_i \leftarrow \mathbb{Z}_l^*$, computes $R_i = r_i G$, and obtains the authenticated public key list $L = \{pk_1, \cdots, pk_n\}$ broadcast by the serving EN for the current epoch. After verifying the validity and freshness of L, TD_i binds the active version tag $V_{id}^{(0)}$ into the cryptographic challenge c_i, which is computed as

$$c_i = \mathscr{H}\left(L \,\|\, pk_i \,\|\, R_i \,\|\, C_i \,\|\, \pi_i \,\|\, com_i \,\|\, V_{id}^{(0)}\right) \tag{9}$$

The terminal computes $s_i = r_i + c_i \cdot sk_i \left(\mathrm{mod}\, l\right)$. The resulting BN signature is $\sigma_i = \left(R_i, s_i\right)$.

- *Step 7:* TD_i transmits the data packet $\left\{\boldsymbol{C}_i, \pi_i, \sigma_i, com_i\right\}$ to the EN.

5.3. Data Aggregation

ENs aggregate the messages received from terminal devices (TD_i) and forward the aggregated results to the CC.

- *Step 1:* The EN collects data packets from n active devices:

$$\left\{\left\{V_{id}^{(0)}, C_1, \pi_1, \sigma_1, com_1\right\}, \cdots, \left\{V_{id}^{(0)}, C_n, \pi_n, \sigma_n, com_n\right\}\right\}. \tag{10}$$

Upon receiving the data packets from the terminals, the EN first checks the plaintext version tag of each packet. It verifies whether the tag matches the current

system version (initially $V_{id}^{(0)}$). If a packet contains a mismatched version tag, the EN discards it to prevent cross-version interference. For the remaining valid packets, the EN proceeds with batch verification and ciphertext aggregation.

- *Step 2:* The EN performs batch verification of the individual BN signatures to authenticate device identities and data integrity:

$$Valid_{sig} = \text{BatchVerify}_{\text{BN}}\left(pk_i, C_i, \sigma_i\right). \tag{11}$$

Subsequently, the EN performs batch verification of the Bulletproofs range proofs. This step ensures that all hidden multidimensional data fall within legitimate ranges without requiring decryption according to Equation (12).

$$Valid_{proof} = \text{BatchVerify}_{\text{bulletproofs}}\left(\pi_i, com_i, \Gamma_j\right). \tag{12}$$

Here, each proof π_i is verified with respect to its corresponding commitment object com_i and the prescribed range bounds $\left\{\Gamma_j\right\}$, thereby ensuring that the verified range statement is bound to the same multidimensional data instance committed by TD_i.

- *Step 3:* If batch verification fails, the EN applies a group rollback mechanism to efficiently locate the invalid data. The failed batch is divided into smaller groups for independent re-verification. This process is recursively subdivided until malicious or faulty devices are accurately identified and excluded.
- *Step 4:* Upon successful verification, the EN aggregates the ciphertexts from all valid terminals on a chunk-by-chunk basis. For each chunk index $k \in \left\{1, \cdots, B\right\}$, the aggregated ciphertext is computed utilizing the homomorphic property of the Paillier cryptosystem:

$$C_{agg}^{(k)} = \prod_{i=1}^{n} C_i^{(k)} \pmod{N^2}. \tag{13}$$

This yields an aggregated ciphertext vector $C_{agg} = \left\{C_{agg}^{(1)}, \cdots, C_{agg}^{(B)}\right\}$.

- *Step 5:* To compress the signatures, the EN applies the non-interactive half-aggregation technique of the BN scheme. After validating the individual signatures, the EN aggregates the scalar components into a single compact scalar as

$$s_{agg} = \sum_{i=1}^{n} s_i \pmod{l}. \tag{14}$$

The resulting aggregate signature is $\sigma_{agg} = \left(R_1, \cdots, R_n, s_{agg}\right)$. This process compresses the n individual signatures by approximately half, thereby optimizing the communication overhead to the CC.

5.4. Data Decryption

The CC verifies and decrypts the aggregated data to extract the multidimensional statistical results.

- *Step 1:* Upon receiving the aggregated packet, the CC verifies the validity of the BN aggregate signature. The CC recalculates the independent challenges

c_i for each terminal and executes the batch verification equation as

$$s_{agg}G = \sum_{i=1}^{n} R_i + \sum_{i=1}^{n} c_i \cdot pk_i. \tag{15}$$

If the equation holds, it cryptographically guarantees that no rogue key attacks have occurred and that the integrity of the aggregated ciphertext C_{agg} is intact.

- *Step 2:* If the verification passes, the CC processes the aggregated data from each Edge Node independently to prevent homomorphic plaintext overflow. Let E denote the total number of active ENs, and let $C_{agg,e}^{(k)}$ denote the k-th aggregated ciphertext chunk received from the e-th EN ($1 \le e \le E$). For each EN e and each chunk $k \in \{1, \cdots, B\}$, the CC performs Paillier decryption as follows:

$$L_e^{(k)} = \frac{\left(C_{agg,e}^{(k)}\right)^{\lambda} \left(\mathrm{mod}\, N^2\right) - 1}{N}, \tag{16}$$

$$D_{agg,e}^{(k)} = \left(L_e^{(k)} \cdot \mu\right) \left(\mathrm{mod}\, N\right). \tag{17}$$

Because the adaptive chunking mechanism enforces $M^{(k)} \le T_{cap}$ during the data collection phase, and the number of devices aggregated by a single EN is bounded by $n_e \le n_{\max}$, the arithmetic sum for each EN satisfies:

$$D_{agg_{real},e}^{(k)} = \sum_{i=1}^{n_e} D_i^{(k)} \le n_{\max} \cdot T_{cap} < N. \tag{18}$$

Therefore, no modular truncation occurs during the decryption of any individual EN's packet, ensuring data integrity at the edge level.

- *Step 3:* To obtain the global statistical results across the entire system, the CC aggregates the decrypted chunks from all ENs in the plaintext domain as

$$D_{global_{agg}}^{(k)} = \sum_{e=1}^{E} D_{agg,e}^{(k)}. \tag{19}$$

- *Step 4*: Finally, the CC performs CRT decoding to recover the aggregated value for each specific dimension. For a given dimension j, the CC identifies the specific chunk k containing the modulus ψ_{s_j} (*i.e.*, $\psi_{s_j} \in S^{(k)}$) and extracts the value as

$$d_{total,j} = D_{global_{agg}}^{(k)} \left(\mathrm{mod}\, \psi_{s_j}\right). \tag{20}$$

For multi-EN deployment, Eq. (20) is directly applicable only when the modulus ψ_{s_j} is sufficient to bound the system-wide sum of dimension j. Otherwise, each EN should first decode its local per-dimension aggregate, after which the CC computes the final global sum by adding the decoded results from all ENs.

5.5. Dynamic Dimension Update

1. Trigger Conditions

- Expanding data dimensions may be necessary to accommodate new business requirements, such as adding new sensor types or statistical indicators.
- Unused redundant prime numbers must remain available in the existing mod-

ulus set Ψ (*i.e.*, the total number of dimensions has not exceeded $K_{\max}+20\%$).

2. Dimension Update Process

Once the trigger conditions are met, the dimension update process is executed as follows, integrating with the adaptive chunking mechanism.

- *Step 1:* The CC selects a prime ψ_{new} from the unused modulus set Ψ that satisfies $\psi_{\text{new}} > n_{\max} \cdot \Gamma_{\text{new}}$, where Γ_{new} is the maximum boundary for the new dimension and $n_{\max}$ is the maximum number of terminal devices. The CC increments the global system version to V_{id}^{new} and broadcasts $\left\{V_{id}^{\text{new}}, \psi_{\text{new}}, \Gamma_{\text{new}}\right\}$ to all ENs and TDs.
- *Step 2:* Upon receiving the update, TD_i updates its active version to V_{id}^{new} and adds ψ_{new} to its active dimension set: $S_{\text{active}}^{\text{new}} = S_{\text{active}} \cup \left\{\psi_{\text{new}}\right\}$. To prevent overflow, TD_i re-runs the greedy capacity check against T_{cap}. If the addition of ψ_{new} causes a chunk to exceed T_{cap}, TD_i automatically allocates ψ_{new} to a new chunk $B+1$, ensuring the system remains overflow-free.
- *Step 3:* TD_i updates its local CRT packing parameters for the specific chunk k containing ψ_{new}. When uploading the updated data vector $d_i = \left(d_{i1}, \cdots, d_{im}, d_{i\text{new}}\right)$, it computes the new packed values $\left\{D_i^{(k)}\right\}$ and encrypts them into a new ciphertext vector C_i^{new}.
- *Step 4:* To prove the validity of the new dimension, TD_i generates an updated Bulletproofs range proof π_{new} covering all dimensions (including $d_{i\text{new}} \le \Gamma_{\text{new}}$) and an updated commitment com_i^{new}. To ensure consistent security, TD_i maintains the BN signature mechanism for the updated packet. It generates a random nonce $r_i \leftarrow \mathbb{Z}_l^*$, computes $R_i = r_i G$, and calculates the cryptographic challenge as

$$c_i = \mathscr{H}\left(L \,\|\, pk_i \,\|\, R_i \,\|\, C_i^{\text{new}} \,\|\, \pi_{\text{new}} \,\|\, com_i^{\text{new}} \,\|\, V_{id}^{\text{new}}\right). \tag{21}$$

The terminal uses its private key sk_i to compute $s_{i,\text{new}} = r_i + c_i \cdot sk_i \left(\text{mod}\, l\right)$, resulting in the updated signature $\sigma_{i,\text{new}} = \left(R_i, s_{i,\text{new}}\right)$.

- *Step 5:* The ENs filter packets by the new version tag V_{id}^{new}, perform the vectorized aggregation, and execute the BN half-aggregation as defined in Section V.C. Subsequently, the CC verifies the aggregate signature, decrypts the chunks, and decodes the new dimension using the CRT as

$$d_{\text{new}}^{\text{total}} = D_{agg}^{(k)} \left(\text{mod}\, \psi_{\text{new}}\right), \tag{22}$$

where k is the index of the chunk containing ψ_{new}.

6. Security Analysis

In this section, we analyze the security of the proposed dimension-scalable data aggregation scheme under standard cryptographic assumptions. Rather than relying solely on informal attack scenarios, we provide theorem-based security arguments and proof sketches for data privacy, authenticity, and integrity. The following arguments are developed under the system and threat assumptions defined

in Section III.

6.1. Data Confidentiality (IND-CPA Security)

Theorem 1. Under the decisional composite residuosity (DCR) assumption, the Paillier-based encryption component of the proposed scheme achieves IND-CPA security for the packed multidimensional data.

Proof sketch. The argument follows the standard IND-CPA security intuition of Paillier encryption under the DCR assumption. In the proposed scheme, each multidimensional data vector is first deterministically packed into a plaintext integer by the public CRT-based encoding rule and is then encrypted by the Paillier cryptosystem. Therefore, any PPT adversary that can distinguish between encryptions of two packed multidimensional plaintexts with non-negligible advantage can be used to distinguish Paillier encryptions under the same public key, contradicting the IND-CPA security of Paillier under the DCR assumption. Hence, the confidentiality of the packed multidimensional data reduces to the IND-CPA security of the Paillier encryption component. In addition, the adaptive chunking mechanism keeps each packed plaintext within the admissible encoding bound, thereby avoiding ambiguity caused by modular overflow during aggregation and decoding.

6.2. Data Authenticity and Integrity (EUF-CMA Security)

Theorem 2. In the random oracle model, the BN half-aggregation mechanism of the proposed scheme is existentially unforgeable against chosen-message attacks under the hardness of the discrete logarithm (DL) problem in $\mathbb{G}$.

Proof sketch. The argument follows the standard Bellare–Neven-style security intuition in the random oracle model. Suppose that a PPT adversary can forge a valid BN aggregate signature for the target public-key list with non-negligible probability. Then, by replaying the adversary under the same randomness but with a different oracle response, one can obtain two valid aggregate signatures corresponding to the same nonce values and different challenge values. Their difference yields a non-trivial linear relation over the involved public keys, from which the secret scalar associated with the target public key can be extracted using the standard forking-based argument. Because a successful forgery with non-negligible probability would imply an efficient solver for the DL problem in $\mathbb{G}$, this contradicts the stated hardness assumption. Hence, under the stated assumptions, the BN half-aggregation mechanism provides authenticity and integrity protection against chosen-message forgery. Moreover, because all terminals in the same epoch compute their challenges over the same authenticated public-key list, the attack surface for rogue-key-style manipulation is reduced during EN-side batch verification.

6.3. Consistency of Commitments and Range Proofs

In the proposed scheme, com_i denotes the commitment object associated with the multidimensional data of TD_i, and π_i denotes the corresponding Bulletproofs range proof. Owing to the hiding and binding properties of the underlying

commitment mechanism, together with the soundness of Bulletproofs, the EN can verify that each committed dimension satisfies its prescribed range constraint without learning the plaintext values. Since the verifier checks π_i with respect to the corresponding com_i, the range proof is bound to the same multidimensional data instance contained in the packet generated by TD_i.

7. Performance Evaluation

7.1. Experimental Setup

This section evaluates the performance of the proposed scheme regarding computational and communication costs, followed by a comparison with four existing schemes: ESMA [29], MMSDA [30], EPPA [15], and LPPDA [37].

To ensure fairness, the security level is set to 112 bits across all evaluated schemes. The Paillier cryptosystem adopts 1024-bit primes (p, q) and a 2048-bit modulus N. For the elliptic curve operations, the NIST P-224 curve is utilized to provide a 224-bit prime field, where group elements are 448 bits in length (uncompressed). For schemes involving bilinear pairings (e.g., ESMA), an equivalent Barreto-Naehrig (BN-254) curve is employed to maintain a consistent security strength. Identifier and timestamp sizes are fixed at 32 bits and 64 bits, respectively.

Key cryptographic operations are simulated using the PBC and OpenSSL libraries. The experiments are conducted on a PC running a 64-bit Windows 11 operating system, equipped with an Intel Core i5-1035G1 CPU (1.00 GHz) and 16 GB RAM. The execution times of the basic operations, averaged over 1000 trials, are summarized in Table 2.

Table 2. Basic operation execution time (milliseconds).

Symbol	Definition	Time (ms)
$T_{Pail_{En}}$	Standard Paillier encryption time	164.826
T_M	Modular multiplication in $\mathbb{Z}_{N^2}$	0.085
T_E	Modular exponentiation in $\mathbb{Z}_{N^2}$	1.821
T_{bp}	Bilinear pairing operation time	3.851
T_{PM}	Elliptic curve point multiplication time	0.281
T_{Pa}	Elliptic curve point addition time	0.004
T_h	Generic hash function computation time	0.001
T_H	Hash-to-point operation time	0.096
T_P	Local point multiplication verification time	0.050
T_{enc}	Optimized Paillier encryption time	5.200
T_{sig}	BN signature generation time	0.300
T_{crt}	Single CRT packing time	0.010

7.2. Computational Costs

It is typically assumed that the CC possesses sufficient computational and com-

munication resources. Therefore, the analysis of computational complexity is primarily focused on TDs and ENs.

In our scheme, TDs first pack multidimensional data into a single integer using the CRT, which is equivalent to data representation in a composite modulus system. Each terminal then encrypts the packed data using an optimized Paillier homomorphic encryption algorithm and generates a BN signature alongside a Bulletproofs range proof to verify data validity and authenticity. After receiving the ciphertexts, ENs perform batch verification of the signatures and range proofs, followed by ciphertext aggregation.

Therefore, the total computational costs for TDs and ENs are $mT_{crt}+T_{enc}+T_{sig}+\left(2m+2\log_2 m\right)T_{PM}$ and $\left(n-1\right)T_M+nT_h+\left(n+1\right)T_{PM}+n\left(2\log_2 m\right)T_{PM}$, respectively.

For the comparison schemes, in ESMA [29], the total computational costs for the terminal and aggregator are $T_M+T_{Pail_{En}}$ and $\left(n-1\right)T_M+\left(n+1\right)T_{bp}$, respectively. In MMSDA [30], the terminal performs $\left(m+\lceil m/31\rceil\right)$ exponentiations and m multiplications in $\mathbb{Z}_{N^2}$, for a total cost of $\left(m+\lceil m/31\rceil\right)T_E+mT_M$, while the aggregator cost is $\left(n-1\right)\lceil m/31\rceil T_M$. In the EPPA [15] scheme, the terminal's total cost is $\left(m+\lceil m/16\rceil\right)T_E$, and the aggregator's cost is $\left(n-1\right)\lceil m/16\rceil T_M$. In LPPDA [37], the total computational costs for the terminal and aggregator are $3mT_E+mT_h$ and $\left(n+3\right)T_E+\left(3n+1\right)T_h+5nT_M$, respectively.

The computational costs of each scheme are summarized in **Table 3**. As shown in **Figure 2**, the proposed scheme provides a significant advantage in terminal-side computational cost compared with MMSDA [30], EPPA [15], and LPPDA [37], which suffer from rapid overhead growth due to repeated modular exponentiations for each data dimension. In contrast, our scheme mitigates this overhead by executing only a single optimized Paillier encryption regardless of dimensionality. Furthermore, the growth in our scheme is predominantly driven by Bulletproofs generation, maintaining a superior efficiency profile over the constant-cost ESMA [29] across the evaluated dimension range ($m \le 200$).

In contrast, as illustrated in **Figure 3**, the computational cost for ENs in the proposed scheme is higher than that of MMSDA, EPPA, and LPPDA, and closely parallels ESMA. This increase is an inherent and justifiable trade-off for integrating Bulletproofs and BN signatures to achieve robust verifiability, a critical security feature absent in MMSDA and EPPA. Specifically, our EN performs rigorous cryptographic batch verification of zero-knowledge proofs and digital signatures, preventing data pollution and impersonation attacks. However, since the execution time for aggregating and verifying 200 terminals remains approximately 1000 ms, this overhead is highly practical for typical edge nodes possessing substantial computational resources. Especially when the number of terminals becomes large, the batch verification operations with logarithmic complexity $O\left(\log\left(nm\right)\right)$ keep the computational burden manageable compared to strictly linear verification approaches.

It is worth noting that the logarithmic computational advantage $O\left(\log\left(nm\right)\right)$

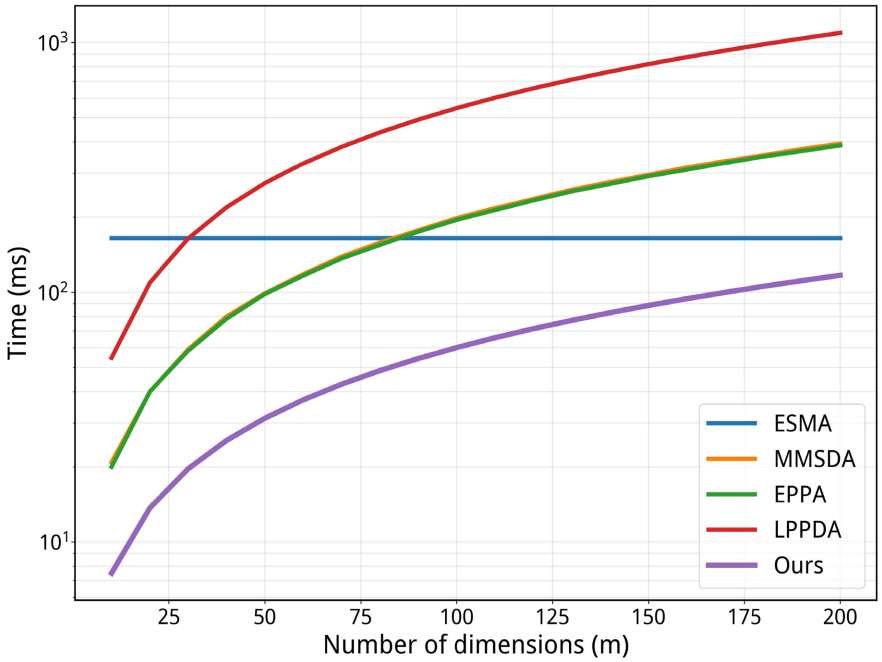

Figure 2. Terminal computational cost comparison.

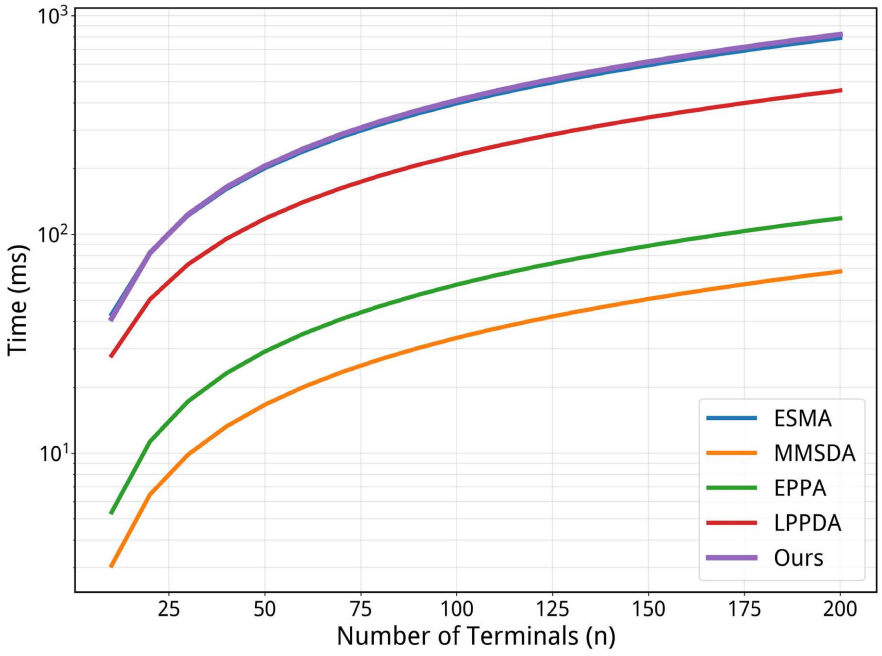

Figure 3. Edge node computational cost comparison.

of our batch verification is primarily achieved under an attack-free or low-fault environment. In adversarial scenarios where malicious terminals inject invalid proofs, the recursive group rollback mechanism is repeatedly triggered, temporarily degrading the verification efficiency towards a linear complexity $O(n)$ in the worst case. However, considering that the proportion of compromised devices in practical deployments is typically sparse, the amortized computational cost of

our scheme across long-term operations remains strictly bounded. Therefore, our scheme achieves a robust balance between rigorous security guarantees and high-dimensional aggregation efficiency.

Table 3. Computational cost comparison of each scheme.

Scheme	Terminal Cost	Edge Node (Aggregator) Cost
ESMA	$T_M + T_{Pail_{En}}$	$(n-1)T_M + (n+1)T_{bp}$
MMSDA	$\left(m + \lceil m/31 \rceil\right)T_E + mT_M$	$(n-1)\lceil m/31 \rceil T_M$
EPPA	$\left(m + \lceil m/16 \rceil\right)T_E$	$(n-1)\lceil m/16 \rceil T_M$
LPPDA	$3mT_E + mT_h$	$(n+3)T_E + (3n+1)T_h + 5nT_M$
Ours	$\underbrace{mT_{crt}}_{\text{CRT}} + \underbrace{T_{enc}}_{\text{Enc}} + \underbrace{T_{sig}}_{\text{BN\_Sig}} + \underbrace{2mT_{PM} + (2\log_2 m)T_{PM}}_{\text{BP\_Prove}}$	$\underbrace{(n-1)T_M}_{\text{Agg}} + \underbrace{nT_h + (n+1)T_{PM}}_{\text{BatchBN}} + \underbrace{n(2\log_2 m)T_{PM}}_{\text{BatchBP}}$

7.3. Communication Costs

The communication overhead is evaluated across two primary transmission stages: from TDs to ENs, and from ENs to the CC. A comprehensive comparison of the communication costs for each scheme is summarized in Table 4.

Table 4. Communication cost comparison of each scheme (bits).

Scheme	Terminal Cost	Edge Node Cost
ESMA	$(m+3)\times 224 + 320$	$(m+3)\times 224 + 320$
MMSDA	$\lceil m/31 \rceil \times 4096 + 512$	$\lceil m/31 \rceil \times 4096 + 512$
EPPA	$\lceil m/16 \rceil \times 4096 + 512$	$\lceil m/16 \rceil \times 4096 + 512$
LPPDA	8288	10,336
Ours	$448\lceil \log_2 m \rceil + 5664$	4544

1. From TDs to ENs: In the proposed scheme, each TD transmits a comprehensive data packet $\{C_i, \pi_i, \sigma_i, com_i\}$ to the EN. This packet inherently encapsulates the Paillier ciphertext, the Bulletproofs range proof, the BN signature, and the commitment, yielding a total communication cost of $2\times 2048 + \left(2\lceil \log_2 m \rceil + 4\right)\times 224 + 2 \times 224 + 224 = 448\lceil \log_2 m \rceil + 5664$ bits. For the baselines, the ESMA [29] scheme requires each terminal to send $\{C_{ij}, ID_{ij}, TS, \sigma_{ij}\}$, incurring an overhead of $(m+3)\times 224 + 320$ bits. MMSDA [30] and EPPA [15] transmit payloads of $\{C_i, \sigma_i\}$ and $\{C_i \| RA \| U_i \| TS \| \sigma_i\}$, which generate communication costs of $\lceil m/31 \rceil \times 4096 + 512$ bits and $\lceil m/16 \rceil \times 4096 + 512$ bits, respectively. Meanwhile, LPPDA [37] involves sending $\{CT_{i,j}, S_{i,j}, R_i, T_{i,j}, ID_i\}$, resulting in a fixed cost of 8288 bits.

2. From ENs to the CC: In our scheme, the EN forwards only the aggregated result $\{C_{agg}, \sigma_{agg}\}$ to the CC. Due to the inherent properties of ciphertext aggregation and BN half-aggregation, this process incurs a strictly constant communication cost of

$2\times 2048+2\times 224=4544$ bits. In contrast, the ESMA [29] scheme forwards $\{C_{ij}, ID_{ij}, TS, \sigma_{ij}\}$ to the CC, requiring $(m+3)\times 224+320$ bits. The MMSDA [30] and EPPA [15] aggregators send $\{C,\sigma\}$ and $\{C \| RA \| GW \| TS \| \sigma_g\}$, demanding $\lceil m/31\rceil\times 4096+512$ bits and $\lceil m/16\rceil\times 4096+512$ bits, respectively. Finally, LPPDA [37] forwards $\{CT_{k,j}, S_{k,j}, R_k, U_{k,j}, T_{k,j}, ID_k\}$, maintaining a constant overhead of 10336 bits.

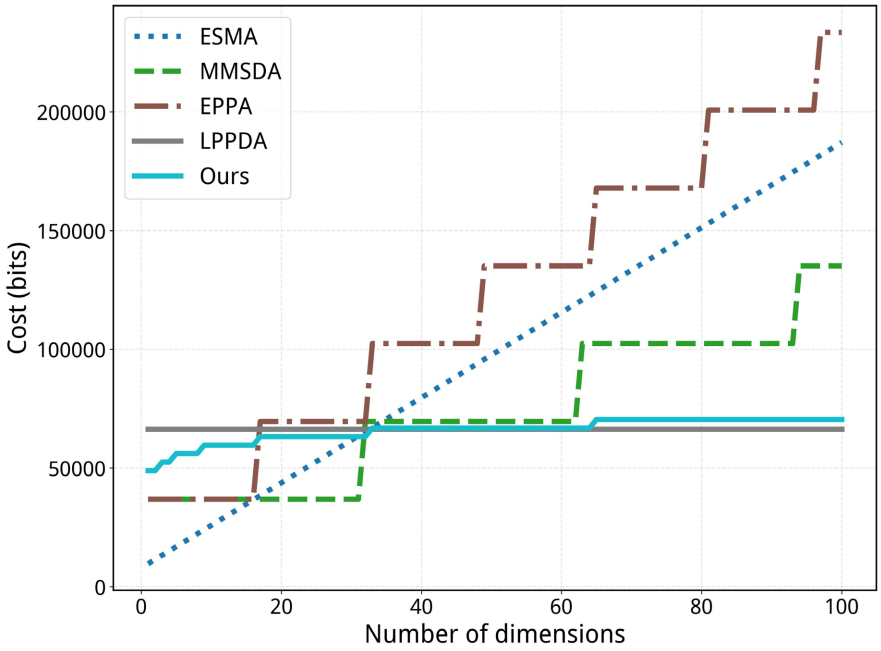

Figure 4. Terminal communication cost comparison.

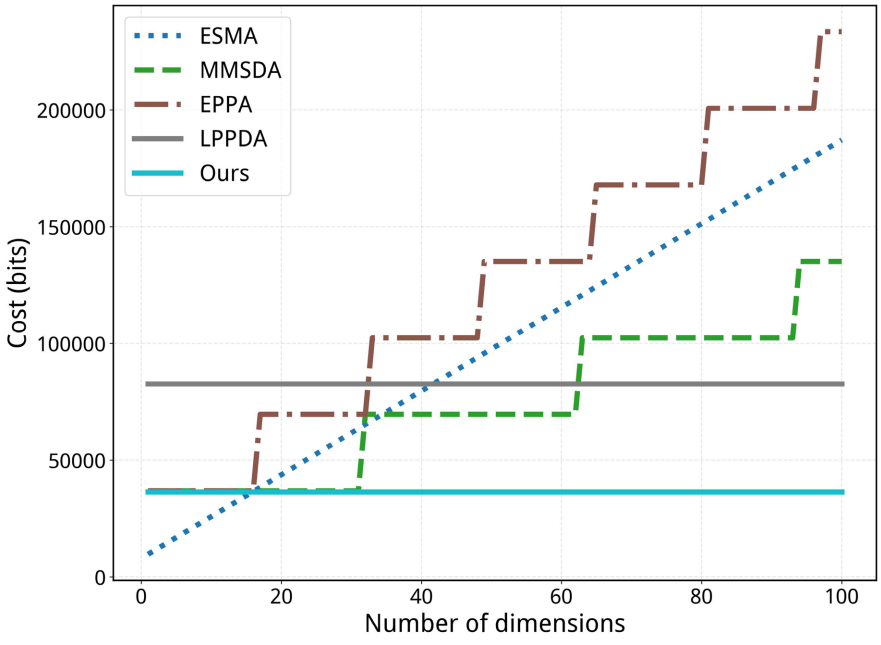

Figure 5. Edge node communication cost comparison.

As shown in Figure 4, the communication overhead from TDs to ENs in the proposed scheme exhibits a logarithmic growth pattern $O(\log m)$ with respect to the data dimension m. Although the baseline payload is marginally higher than MMSDA and EPPA in low-dimensional settings (e.g., $m < 30$) due to the inclusion of Bulletproofs and BN signatures, our scheme inherently suppresses the severe step-linear increases caused by massive ciphertext expansions in other schemes. Consequently, it establishes a decisive bandwidth advantage as m scales for high-dimensional IoT applications.

Furthermore, as shown in Figure 5, the communication cost from ENs to the CC in our scheme remains strictly constant, completely independent of the data dimension m. While ESMA, MMSDA, and EPPA suffer from dimension-dependent overheads, our scheme maintains the lowest cost across the entire evaluated range. This optimal bandwidth efficiency is achieved because the edge node forwards only a single aggregated Paillier ciphertext and one-half-aggregated BN signature, ensuring that the rigorous batch verification process introduces zero additional communication burden to the backbone network.

8. Conclusion

This paper proposes an efficient and verifiable multidimensional data aggregation scheme featuring dynamic dimension scalability and privacy protection in edge computing systems. In the proposed scheme, TDs apply CRT packing to combine multidimensional data into a single integer and encrypt it using an optimized Paillier homomorphic encryption algorithm. Consequently, ENs directly aggregate the ciphertexts without decryption, protecting data privacy. Furthermore, Bulletproofs-based zero-knowledge proofs and BN signatures with half-aggregation are integrated to ensure the integrity, authenticity, and range compliance of the encrypted data. This enables the CC to verify data authenticity and conduct an in-depth analysis of high-dimensional aggregated data. Overall, the proposed scheme achieves a balance among privacy protection, computational efficiency, and dynamic scalability. Finally, security proofs and performance evaluations demonstrate that the proposed scheme reduces computational and communication overhead while ensuring robust security.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

[1] Ahmadi, Z., Haghi Kashani, M., Nikravan, M. and Mahdipour, E. (2021) Fog-Based Healthcare Systems: A Systematic Review. *Multimedia Tools and Applications*, **80**, 36361-36400. https://doi.org/10.1007/s11042-021-11227-x

[2] Wang, B. (2025) Construction of an Intelligent Medical Monitoring System Network for Exercise Rehabilitation Based on the Internet of Things. *International Journal of High Speed Electronics and Systems*. https://doi.org/10.1142/s0129156425402013

[3] Lu, R. (2016) Privacy-Enhancing Aggregation Techniques for Smart Grid Commu-

nications. Springer.

[4] El-deep, S.E., Abohany, A.A., Sallam, K.M. and El-Mageed, A.A.A. (2025) A Comprehensive Survey on Impact of Applying Various Technologies on the Internet of Medical Things. *Artificial Intelligence Review*, **58**, Article No. 86. https://doi.org/10.1007/s10462-024-11063-z

[5] Kouari, O.E., Lazaar, S. and Achoughi, T. (2025) Fortifying Industrial Cybersecurity: A Novel Industrial Internet of Things Architecture Enhanced by Honeypot Integration. *International Journal of Electrical and Computer Engineering* (*IJECE*), **15**, 1089-1098. https://doi.org/10.11591/ijece.v15i1.pp1089-1098

[6] sun, G., Xing, X., Qian, Z. and Li, W. (2021) Edge Computing Assisted Privacy-Preserving Data Computation for IoT Devices. *Computer Communications*, **166**, 208-215. https://doi.org/10.1016/j.comcom.2020.11.018

[7] Dastjerdi, A.V. and Buyya, R. (2016) Fog Computing: Helping the Internet of Things Realize Its Potential. *Computer*, **49**, 112-116. https://doi.org/10.1109/mc.2016.245

[8] Dong, J., Cheng, J., Wu, J., Zhang, C., Zhao, S. and Tang, X. (2025) Real-Time AIoT for AAV Antenna Interference Detection via Edge-Cloud Collaboration. *IEEE Internet of Things Journal*, **12**, 10664-10680. https://doi.org/10.1109/jiot.2024.3512867

[9] Liu, Y., Peng, M., Shou, G., Chen, Y. and Chen, S. (2020) Toward Edge Intelligence: Multiaccess Edge Computing for 5G and Internet of Things. *IEEE Internet of Things Journal*, **7**, 6722-6747. https://doi.org/10.1109/jiot.2020.3004500

[10] Xie, D., Yang, J., Bian, W., Chen, F. and Wang, T. (2023) An Improved Identity-Based Anonymous Authentication Scheme Resistant to Semi-Trusted Server Attacks. *IEEE Internet of Things Journal*, **10**, 734-746. https://doi.org/10.1109/jiot.2022.3203991

[11] Roman, R., Zhou, J. and Lopez, J. (2013) On the Features and Challenges of Security and Privacy in Distributed Internet of Things. *Computer Networks*, **57**, 2266-2279. https://doi.org/10.1016/j.comnet.2012.12.018

[12] Liu, Y., Guo, W., Fan, C., Chang, L. and Cheng, C. (2019) A Practical Privacy-Preserving Data Aggregation (3PDA) Scheme for Smart Grid. *IEEE Transactions on Industrial Informatics*, **15**, 1767-1774. https://doi.org/10.1109/tii.2018.2809672

[13] Fan, C., Huang, S. and Lai, Y. (2014) Privacy-Enhanced Data Aggregation Scheme against Internal Attackers in Smart Grid. *IEEE Transactions on Industrial Informatics*, **10**, 666-675. https://doi.org/10.1109/tii.2013.2277938

[14] Ali, W., Din, I.U., Almogren, A. and Kim, B. (2022) A Novel Privacy Preserving Scheme for Smart Grid-Based Home Area Networks. *Sensors*, **22**, Article 2269. https://doi.org/10.3390/s22062269

[15] Lu, R.X., Liang, X.H., Li, X., Lin, X.D. and Shen, X.M. (2012) EPPA: An Efficient and Privacy-Preserving Aggregation Scheme for Secure Smart Grid Communications. *IEEE Transactions on Parallel and Distributed Systems*, **23**, 1621-1631. https://doi.org/10.1109/tpds.2012.86

[16] Li, S., Xue, K., Yang, Q. and Hong, P. (2018) PPMA: Privacy-Preserving Multisubset Data Aggregation in Smart Grid. *IEEE Transactions on Industrial Informatics*, **14**, 462-471. https://doi.org/10.1109/tii.2017.2721542

[17] Mohammadali, A. and Haghighi, M.S. (2021) A Privacy-Preserving Homomorphic Scheme with Multiple Dimensions and Fault Tolerance for Metering Data Aggregation in Smart Grid. *IEEE Transactions on Smart Grid*, **12**, 5212-5220. https://doi.org/10.1109/tsg.2021.3049222

[18] Zhang, X., You, L. and Hu, G. (2022) An Efficient and Robust Multidimensional Data Aggregation Scheme for Smart Grid Based on Blockchain. *IEEE Transactions on Net-*

work and Service Management, **19**, 3949-3959. https://doi.org/10.1109/tnsm.2022.3217312

[19] Merad Boudia, O.R., Senouci, S.M. and Feham, M. (2017) Elliptic Curve-Based Secure Multidimensional Aggregation for Smart Grid Communications. *IEEE Sensors Journal*, **17**, 7750-7757. https://doi.org/10.1109/jsen.2017.2720458

[20] Marcolla, C., Sucasas, V., Manzano, M., Bassoli, R., Fitzek, F.H.P. and Aaraj, N. (2022) Survey on Fully Homomorphic Encryption, Theory, and Applications. *Proceedings of the IEEE*, **110**, 1572-1609. https://doi.org/10.1109/jproc.2022.3205665

[21] Wolf, J.K. (1986) The Chinese Remainder Theorem and Applications. In: Blake, I.F. and Poor, H.V., Eds., *Communications and Networks*, Springer, 390-405. https://doi.org/10.1007/978-1-4612-4904-7_17

[22] Wu, Z., Liu, Y. and Jia, X. (2020) A Novel Hierarchical Secret Image Sharing Scheme with Multi-Group Joint Management. *Mathematics*, **8**, Article 448. https://doi.org/10.3390/math8030448

[23] Zhu, B., Li, Y., Hu, G. and Zhang, M. (2023) A Privacy-Preserving Data Aggregation Scheme Based on Chinese Remainder Theorem in Mobile Crowdsensing System. *IEEE Systems Journal*, **17**, 4257-4266. https://doi.org/10.1109/jsyst.2023.3262321

[24] Wu, L., Xu, M., Fu, S., Luo, Y. and Wei, Y. (2022) FPDA: Fault-Tolerant and Privacy-Enhanced Data Aggregation Scheme in Fog-Assisted Smart Grid. *IEEE Internet of Things Journal*, **9**, 5254-5265. https://doi.org/10.1109/jiot.2021.3109153

[25] Liu, H., Gu, T., Liu, Y., Song, J. and Zeng, Z. (2020) Fault-Tolerant Privacy-Preserving Data Aggregation for Smart Grid. *Wireless Communications and Mobile Computing*, **2020**, Article ID: 8810393. https://doi.org/10.1155/2020/8810393

[26] Guo, C., Tian, P. and Choo, K.R. (2021) Enabling Privacy-Assured Fog-Based Data Aggregation in E-Healthcare Systems. *IEEE Transactions on Industrial Informatics*, **17**, 1948-1957. https://doi.org/10.1109/tii.2020.2995228

[27] Chen, Y., Martinez-Ortega, J., Castillejo, P. and Lopez, L. (2019) A Homomorphic-Based Multiple Data Aggregation Scheme for Smart Grid. *IEEE Sensors Journal*, **19**, 3921-3929. https://doi.org/10.1109/jsen.2019.2895769

[28] Liu, J., Weng, J., Yang, A., Chen, Y. and Lin, X. (2020) Enabling Efficient and Privacy-Preserving Aggregation Communication and Function Query for Fog Computing-Based Smart Grid. *IEEE Transactions on Smart Grid*, **11**, 247-257. https://doi.org/10.1109/tsg.2019.2920836

[29] Merad-Boudia, O.R. and Senouci, S.M. (2021) An Efficient and Secure Multidimensional Data Aggregation for Fog-Computing-Based Smart Grid. *IEEE Internet of Things Journal*, **8**, 6143-6153. https://doi.org/10.1109/jiot.2020.3040982

[30] Peng, C., Luo, M., Vijayakumar, P., He, D., Said, O. and Tolba, A. (2022) Multifunctional and Multidimensional Secure Data Aggregation Scheme in WSNs. *IEEE Internet of Things Journal*, **9**, 2657-2668. https://doi.org/10.1109/jiot.2021.3077866

[31] Xu, S., Fan, J., Wei, X. and Zhang, Y. (2025) FGAMD: A Flexible-Group-Based Privacy-Preserving Aggregation Scheme for Multidimensional Data in Edge-Enhanced IoT. *IEEE Internet of Things Journal*, **12**, 13961-13971. https://doi.org/10.1109/jiot.2024.3524087

[32] Boschini, C., Fiore, D., Pagnin, E., Torresetti, L. and Visconti, A. (2024) Progressive and Efficient Verification for Digital Signatures: Extensions and Experimental Results. *Journal of Cryptographic Engineering*, **14**, 551-575. https://doi.org/10.1007/s13389-024-00358-0

[33] Zhang, Y., Tang, Y., Li, C., Zhang, H. and Ahmad, H. (2024) Post-Quantum Secure

Identity-Based Signature Scheme with Lattice Assumption for Internet of Things Networks. *Sensors*, **24**, Article 4188. https://doi.org/10.3390/s24134188

[34] Shrivastava, P., Alam, B. and Alam, M. (2024) Blockchain Based Quantum Resistant Signature Algorithm for Data Integrity Verification in Cloud and Internet of Everything. *ICST Transactions on Scalable Information Systems*, **11**, 1-6. https://doi.org/10.4108/eetsis.5488

[35] Paillier, P. (2019) Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. In: Stern, J., Ed., *Advances in Cryptology—EUROCRYPT'*99, Springer, 223-238. https://doi.org/10.1007/3-540-48910-x_16

[36] Zhou, L., Diro, A., Saini, A., Kaisar, S. and Hiep, P.C. (2024) Leveraging Zero Knowledge Proofs for Blockchain-Based Identity Sharing: A Survey of Advancements, Challenges and Opportunities. *Journal of Information Security and Applications*, **80**, Article ID: 103678. https://doi.org/10.1016/j.jisa.2023.103678

[37] Ding, Y., Wang, B., Wang, Y., Zhang, K. and Wang, H. (2020) Secure Metering Data Aggregation with Batch Verification in Industrial Smart Grid. *IEEE Transactions on Industrial Informatics*, **16**, 6607-6616. https://doi.org/10.1109/tii.2020.2965578