QSDA: Quality-Aware Secure Multi-Dimensional Data Aggregation with Location Privacy for HIoT

Yijun Wang, Lei Wu*, *Member, IEEE*, Ye Su, Hao Wang, *Member, IEEE*,
Weizhi Meng, *Senior Member, IEEE* and Zhiquan Liu, *Senior Member, IEEE*

***Abstract*—Data aggregation, as a data processing technique, facilitates accurate diagnosis in the Healthcare Internet of Things (HIoT) by integrating multi-source heterogeneous health data. However, achieving efficient and secure aggregation of multi-dimensional medical data remains challenging, particularly when simultaneously preserving location privacy and providing fair, quality-driven incentives. To address these issues, this paper proposes a Quality-Aware Secure Multi-Dimensional Data Aggregation scheme with Location Privacy for HIoT (QSDA). First, the scheme employs inner product encryption to support aggregation task matching without revealing users' actual coordinates, and further integrates symmetric homomorphic encryption with super-increasing sequences to enable one-stop compressed aggregation of multi-dimensional data, thereby effectively supporting common statistical operations such as mean and variance. Second, it introduces a data quality incentive mechanism based on offset metrics, while leveraging blockchain auditing to ensure the traceability of the aggregation process and the verifiability of the aggregation results. Finally, security analysis and performance evaluation demonstrate the scheme's effectiveness and efficiency.**

***Index Terms*—HIoT, symmetric homomorphic encryption, multi-dimensional data aggregation, dynamic location privacy, incentive mechanism.**

I. Introduction

DRIVEN by the rapid advances in mobile healthcare and remote diagnostic technologies, the Healthcare Internet of Things (HIoT) is reshaping the landscape of traditional medical services. Wearable devices and intelligent sensors can continuously collect users' physiological and behavioral data, and securely aggregating multi-source health data provides crucial support for health monitoring and personalized treatment [1]–[3]. To enhance the real-time performance and scalability of medical data processing, researchers have adopted an edge–cloud collaborative architecture, where edge nodes perform distributed preprocessing and local aggregation while the "cloud server" [4] conducts global analysis and decision optimization, effectively reducing the computational burden on centralized servers and improving system responsiveness. However, medical data are typically multi-dimensional, heterogeneous, and highly sensitive. Any leakage during transmission, aggregation, or storage may expose personal privacy or lead to medical data misuse. Therefore, achieving low-latency and high-accuracy encrypted aggregation of multi-dimensional medical data while preserving confidentiality and privacy has become a crucial challenge [5].

To address the above issues, researchers have proposed various privacy-preserving data aggregation mechanisms. Some studies [6] employ the Paillier homomorphic encryption scheme, which enables summation operations on encrypted data based on its additive homomorphism. Although this method can perform aggregation in the encrypted state, its functionality is limited to additive homomorphism, and its computational efficiency is low, making it unsuitable for resource-constrained devices. In contrast, symmetric homomorphic encryption (SHE) [7] supports both additive and multiplicative homomorphisms, maintaining strong cryptographic security while enabling richer ciphertext-domain computations for complex encrypted statistics and aggregation tasks. However, when dealing with multi-dimensional heterogeneous medical data such as blood pressure and heart rate [8], SHE still needs to encrypt and aggregate each dimension separately, leading to exponentially increased computation and communication costs that hinder real-time and lightweight deployment. Therefore, although some studies [9] have attempted to integrate super-increasing sequence encoding with improved homomorphic encryption mechanisms to enhance the privacy of multi-dimensional data aggregation, achieving scalable and high-precision secure aggregation under lightweight constraints remains a significant challenge.

Although existing schemes have made notable progress in data privacy protection and efficiency optimization, they generally overlook the issue of location privacy. It is worth noting that medical task allocation often relies on geographic location matching, where securely performing matching and verification without exposing users' real coordinates is crucial for maintaining system trustworthiness. To address this issue, several studies [10] have proposed privacy-preserving location matching mechanisms based on homomorphic encryption or secure multi-party computation, enabling secure associations between users and tasks. However, in the absence of effective incentive mechanisms, users are often reluctant to participate in location verification and data sharing. Therefore, integrating lightweight SHE with extended super-increasing sequence

This work was supported by the National Natural Science Foundation of China under Grant 62472266, 62472265, 62302280 and the Natural Science Foundation of Shandong Province under Grant ZR2025MS1027.

Yijun Wang, Lei Wu, Ye Su, Hao Wang are with the School of Information Science and Engineering, Shandong Normal University, Jinan 250358, China (E-mail:wangyij0619@163.com, wulei@sdnu.edu.cn, suye@sdnu.edu.cn, and wanghao@sdnu.edu.cn).

Weizhi Meng is with the School of Computing and Communications, Lancaster University, United Kingdom (email: weizhi.meng@ieee.org).

Zhiquan Liu is with the College of Cyber Security, Jinan University, Guangzhou 510632, China. (e-mail: zqliu@vip.qq.com).

* Corresponding author.

encoding, alongside location privacy protection, is a necessary direction for constructing efficient and secure medical data aggregation schemes.

To enhance user participation and data trustworthiness, researchers have further introduced incentive mechanisms into privacy-preserving data aggregation processes. Incentive mechanisms are considered an effective means to encourage active user engagement and improve data quality by guiding users to contribute high-precision data through appropriate reward strategies. Existing studies [11] have explored various approaches such as prepaid compensation or reputation-based credits to attract user participation. Unfortunately, most of these schemes neglect the evaluation of the authenticity and validity of the submitted data. In fact, data quality and incentive distribution are positively correlated, and a well-designed incentive mechanism can not only enhance data credibility but also improve the long-term sustainability of the system. However, existing schemes remain insufficient in terms of incentive and auditing mechanisms, lacking the ability to ensure traceability of the aggregation process and verifiability of the results.

In summary, the current approach still cannot meet the data aggregation requirements in HIoT. The main contributions of this paper are summarized as follows:

1) This paper proposes a Quality-Aware Secure Multi-Dimensional Data Aggregation with Location Privacy for HIoT. The scheme jointly preserves medical data privacy, location privacy, and incentive fairness, achieving efficient and secure aggregation under resource-constrained edge environments. Compared with existing schemes, QSDA attains a better balance among privacy, efficiency, and scalability.
2) For aggregation and privacy, the scheme employs vector inner product operations to enable users to participate in task selection without revealing their real coordinates. By integrating lightweight SHE with extended super-increasing sequence encoding, it achieves one-stop compressed aggregation of multi-dimensional data, supports common statistical operations, and reduces computational and communication overhead.
3) For incentive and auditing, the scheme adopts an offset-based data quality evaluation and reward allocation mechanism to encourage high-quality data contributions. Moreover, a blockchain-based auditing module is introduced at the incentive layer to record aggregation indexes and signature information on-chain, ensuring result verifiability and process traceability.
4) A comprehensive security analysis is conducted to demonstrate that the proposed scheme satisfies requirements for data confidentiality, identity authenticity, and resistance to replay attacks. In addition, experimental results show that QSDA incurs low computational and communication overhead. These findings validate both the security and efficiency of the proposed scheme.

II. RELATED WORK

In HIoT systems, achieving secure, trustworthy, and efficient multi-dimensional data aggregation in resource-constrained and privacy-sensitive environments has long been a research focus [12], [13]. Early studies primarily concentrated on encrypted aggregation and lightweight uploading mechanisms for single-dimensional data. Shen et al. [14] proposed a verifiable encrypted statistical analysis method on e-commerce platforms. Yan et al. [15] presented a privacy-preserving, efficient multitask data aggregation scheme based on MCS fog computing, aggregating multiple concurrent tasks from various requesters. However, these solutions are unable to accommodate the complex multi-dimensional structures of real medical data.

To address the challenge of multi-dimensional heterogeneous medical data, researchers have proposed solutions capable of integrating multiple data types while providing fault tolerance. For example, Liu et al. [16] employed the Chinese Remainder Theorem (CRT) for multi-dimensional data processing. Building on this, Chen et al. [17] further developed a fine-grained linear homomorphic encryption scheme that encodes multi-dimensional data using CRT and assigns different weights to each data dimension for users. Notably, some studies have used CRT [18] to compress and encode multi-dimensional data. However, CRT imposes strict requirements on the coprimality of moduli and dimensional consistency, making it difficult to directly perform aggregation and squared analysis within the encrypted domain. To overcome this, Zhao et al. [19] combined Paillier encryption with super-increasing sequence encoding to achieve efficient multi-dimensional data compression. Nevertheless, these schemes have not effectively integrated the multiplicative homomorphic property with data compression mechanisms.

Furthermore, considering the issues of task location matching verification and aggregation result correctness verification, in terms of location verification, Zhang et al. [20] proposed a location-aware verifiable outsourced data aggregation scheme based on the Internet of Vehicles, which achieves verifiability of task area matching. Regarding aggregation result verification, Li et al. [21] designed a designated verifier aggregation signature scheme based on permissioned blockchain. However, the above schemes fail to simultaneously achieve task location privacy protection and integrity verification of aggregation results, which makes it a new challenge to design incentive mechanisms that can take both privacy preservation and result trustworthiness into account.

As user participation increasingly impacts system performance, user incentive mechanisms under privacy protection have become a research focus. For example, Tang et al. [22] employed signature techniques to maintain fair incentives for patients and combined the BGN cryptosystem with Shamir's secret sharing to ensure data fault tolerance. Further, to quantify actual data quality, Yu et al. [23] proposed a privacy-preserving data aggregation and quality assessment protocol based on smart contracts. However, the above methods do not take into account differentiated incentives based on user data quality and also lack effective supervision of the incentive process and aggregation results, making it difficult to ensure both fairness and system trustworthiness.

To enhance the overall auditability and data trustworthiness of the system, blockchain technology has gradually become the foundational framework for privacy-preserving aggregation systems. Wang et al. [24] designed multiple smart contracts to achieve fair transactions and public verification. In blockchain-assisted aggregation schemes tailored for healthcare scenarios, Lee et al. [25] utilized blockchain to ensure data integrity and maintain audit records, while also presenting a secure and flexible approach to data access. Although blockchain features operational transparency and immutability, it inherently lacks capabilities for data quality assessment and verification of computational correctness.

In summary, although current research has achieved significant progress in areas such as privacy encryption, multiple data types, and incentive mechanisms, there remains a lack of a comprehensive and multifunctional unified architecture for data aggregation.

III. Preliminaries

A. Symmetric Homomorphic Encryption

Symmetric Homomorphic Encryption (SHE) is a lightweight symmetric homomorphic encryption scheme that supports homomorphic addition and multiplication. Specifically, SHE consists of three algorithms: (1) key generation *KeyGen()*, (2) encryption *Enc()*, and (3) decryption *Dec()*, as detailed below:

1) KeyGen(): Three security parameters $\{k_0, k_1, k_2\}$, satisfying the relation $k_1 \ll k_2 < k_0$. Select two large prime numbers p and q, each of bit-length k_0, and compute the modulus $\mathcal{N} = p \cdot q$. Construct the private key $sk = (p, \mathcal{L})$,where $|\mathcal{L}| = k_2$. The message space is defined as $\mathcal{M} = \{0,1\}^{k_1}$.

2) Enc(): According to the symmetric encryption formula $Enc(m) = (r \cdot \mathcal{L} + m) \cdot (1 + r' \cdot p) \bmod \mathcal{N}$ where $r \in \{0,1\}^{k_2}$, $r' \in \{0,1\}^{k_0}$. Two encryptions of 0, denoted as $Enc(0)_1$ and $Enc(0)_2$, are generated using the private key sk as part of the public key. Public key $pk = (Enc(0)_1, Enc(0)_2, \mathcal{N}, k_0, k_1, k_2)$, and plaintext $m \in \mathcal{M}$.

$$Enc(m) = m + r_1 \cdot Enc(0)_1 + r_2 \cdot Enc(0)_2 \bmod \mathcal{N},$$

where $r_1, r_2 \in \{0,1\}^{k_2}$.

3) Dec():

$$m = (Enc(m) \bmod p) \bmod \mathcal{L} = (r \cdot \mathcal{L} + m) \bmod \mathcal{L}.$$

Additive and Multiplicative Homomorphic Properties:

$$c_1 + c_2 = Enc(m_1 + m_2), \quad c_1 + m_2 = Enc(m_1 + m_2).$$

$$c_1 \cdot c_2 = Enc(m_1 \cdot m_2), \quad c_1 \cdot m_2 = Enc(m_1 \cdot m_2).$$

It is noteworthy that the SHE scheme adopted in this paper follows the construction in [7], which has been theoretically proven to achieve IND-CPA semantic security.

B. Extended Super-Increasing Sequence

A super-increasing sequence is a key technique for handling multi-type data, known for its excellent decodability and uniqueness. Given a sequence of positive real numbers $A = \{a_1, a_2, \ldots, a_n\}$, if for any $i \in \{2, 3, \ldots, n\}$, the condition

$$a_i > \sum_{j=1}^{i-1} a_j$$

holds, then the sequence is called a super-increasing sequence. The sequence ensures that each component value can be uniquely recovered during decryption and is widely applied in encrypted aggregation scenarios.

To meet the encoding requirements of multi-dimensional medical data, this paper extends the traditional super-increasing sequence and proposes an extended sequence

$$\vec{\mu} = \{\mu_0 = 1, \mu_1, \mu_2, \ldots, \mu_j, \mu_{j+1}, \ldots, \mu_{2j}\},$$

which satisfies the condition

$$\sum_{j=1}^{i-1} u_j \cdot 2^{\max} \cdot n < u_i, \quad i \in [2, n],$$

where $2^{\max}$ denotes the maximum bit length across all dimensions of the medical data. By introducing squared term weights, this approach enables both the original value and its square in each dimension to be mapped into the same ciphertext. The extended super-increasing sequence supports complex statistical analyses, such as mean and variance, in the encrypted domain while maintaining a constant ciphertext length.

Let the user's secret be $s \in \mathbb{Z}_p$. Multiple secrets can be encoded into a single value using a super-increasing sequence. Suppose a participant has l secrets $s_1, s_2, \ldots, s_l$ to share. First, generate a sequence $a_1, a_2, \ldots, a_l$, where $a_1 = 1$, and $a_j > \sum_{i=1}^{j-1} a_i \cdot s_i$. Then, the l secrets can be encoded as:

$$S = \sum_{i=1}^{l} a_i \cdot s_i.$$

$$s_i = \frac{S - (S \bmod a_i)}{a_i}, \quad S = S - s_i \cdot a_i, \quad i = l, l-1, \ldots, 1$$

IV. System Architecture

This section first defines the system model of the proposed scheme, then analyzes the main security threats it may face in practical applications, and further clarifies the corresponding design goals.

A. System Model

The system model is illustrated in Fig. 1. It consists of six distinct entities:

1) Trusted Authority (TA): TA initializes the system and registers entities by generating system parameters and securely distributing keys.

2) Data Owners (DOs): DOs collect and encrypt healthcare data, then upload ciphertexts to fog nodes for local aggregation.

3) Fog Nodes (FNs): FNs assign tasks, perform local aggregation, and support data quality evaluation.

4) Cloud Server (CS): CS verifies signatures, executes global aggregation, and returns aggregated results to requesters.

5) Requesters: Requesters publish tasks to the cloud and receive aggregated results, typically healthcare institutions or researchers.

6) Blockchain (BC): BC functions as an auditing layer, recording key interactions and enabling incentive distribution and malicious behavior detection.

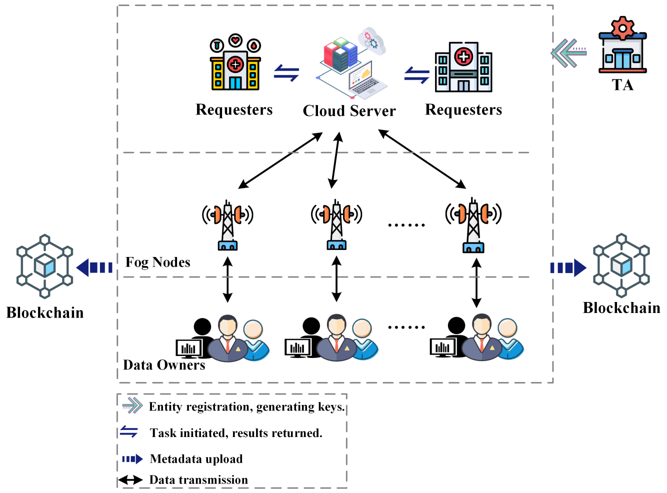


Fig. 1. System Model.

B. Threat Model

We mainly address internal attackers and their associated security threats. We assume that data transmission channels within the system are secure. The trusted authority is regarded as fully trustworthy.

In this threat model, we focus on the potential security risks associated with four system entities: Data Owners, Fog Nodes, Cloud Server, and Requesters. DOs are assumed to be "honest but curious," meaning they honestly determine whether to participate in tasks based on location and incentive agreements, but may attempt to infer other users' health information out of curiosity. The behavior of "honest-but-curious" entities is consistent with the semi-honest model, both referring to participants who follow the protocol while trying to deduce sensitive information from the process. Similarly, FNs are also modeled as "honest-but-curious." They correctly perform local aggregation and securely store patient data, but may collude with the cloud server to obtain private information from other FNs. The cloud server is assumed to honestly execute data storage and task distribution functions, but may also be interested in analyzing the encrypted health data from FNs for sensitive content. Additionally, requesters are considered to honestly issue tasks and receive results, but may still attempt to infer individual privacy from the aggregated medical data they receive.

C. Design Goals

To ensure the security and reliability of multi-dimensional medical data throughout the processes of collection, encryption, aggregation, and decryption, the QSDA scheme is designed around the following five core security objectives:

1) Data Confidentiality: The scheme ensures that user privacy is preserved during data transmission and aggregation, preventing adversaries from inferring individual information from ciphertexts or aggregated results.

2) Data Integrity and Verifiability: Aggregated results can be verified and traced, guaranteeing that data is not tampered with or forged during transmission and storage.

TABLE I
TABLE OF NOTATIONS

Notation	Definition
k_0, k_1, k_2	Security parameters
p, q	Two large prime numbers
N	Secure modulus such that $N = p \times q$
$\vec{\mu}$	A sequence of positive integers
μ_k	The k-th value in the sequence
T	A data aggregation request
$\vec{v}_k$	Vector corresponding to the k-th circular area
$\vec{u}_i$	Vector corresponding to the i-th user's location
$\mathbb{S}$	Task's spatial region constraint
m_i	Medical data of the i-th user
$\hat{m}_i$	Preprocessed value of m_i
C_i	The ciphertext of $\hat{m}_i$
θ_i	Offset value of the i-th user's data
Θ_l'	List of all θ_i under fog node l
C	Aggregated ciphertext of all DOs' data
$\mathcal{L}_{\Theta'}$	List of all Θ_l'
M	Aggregated plaintext of original data
$\hat{M}_{j+k}$	Aggregated squared value of the j-th dimension in M
$\hat{M}_k$	Aggregated value of the j-th dimension in M
AVE_k	Average value of aggregated data
VAR_k	Variance of aggregated data
η_i	Accuracy evaluation value of $\mathbf{m}_i$
ρ_i	Reward allocated to participant i

3) Location Privacy: Users can complete task selection without disclosing their real coordinates, thereby avoiding location privacy risks.

4) Replay Resistance and Unforgeability: The system automatically rejects outdated data and effectively resists forgery and replay attacks, ensuring the authenticity and timeliness of both data and aggregation results.

5) Incentive Fairness: The scheme ensures fairness and impartiality in the incentive process. Meanwhile, an auditing mechanism is integrated to enable traceability and verifiability of aggregation results and thereby enhancing the overall transparency and sustainability of the system.

V. Detailed Scheme

In this section, the structure of the proposed scheme is presented in detail, including five key algorithms: 1) System Initialization; 2) Task Publication; 3) Data Processing; 4) Data Aggregation; 5) Verification and Decryption.

We first introduce the basic notation used throughout the scheme and provide their descriptions in Table I to enhance understanding.

A. Overview of QSDA

The proposed QSDA is a privacy-preserving multi-dimensional data aggregation scheme designed for mobile crowd-sensing scenarios, with a focus on ensuring data privacy, location privacy, and secure location verification in HIoT applications. In data collection, users often generate sensitive information such as location, identity, and health indicators, which necessitates strong privacy guarantees alongside efficient aggregation. For ease of illustration, we take an MCS application in medical services as an example.

As shown in Fig. 2, the overall workflow consists of five stages: system initialization, task publication, data processing, data aggregation, and decryption. In the system initialization stage, a trusted authority generates system parameters and registers all entities. During task publication, the requester submits task details and constraints to the cloud server, which distributes them to appropriate fog nodes and users. In the data processing stage, users perform location verification according to regional constraints. Eligible users collect local multi-dimensional health data and encrypt it. To ensure traceability and tamper resistance during data transmission and storage, the system generates a unique hash code for each encrypted data packet and its associated metadata, which is recorded in the auditing layer. Each fog node performs local aggregation on the received data and computes offset values. In the aggregation phase, the cloud server aggregates results from all FNs to form the complete aggregated dataset. Finally, in the decryption and feedback phase, the requester decrypts the results and evaluates user data quality based on offset information to complete reward allocation. In summary, QSDA enables efficient aggregation of multi-dimensional medical data and quality-driven incentive allocation while preserving user privacy. By incorporating blockchain auditing, it strengthens transparency and provides sustainable trust in HIoT environments.

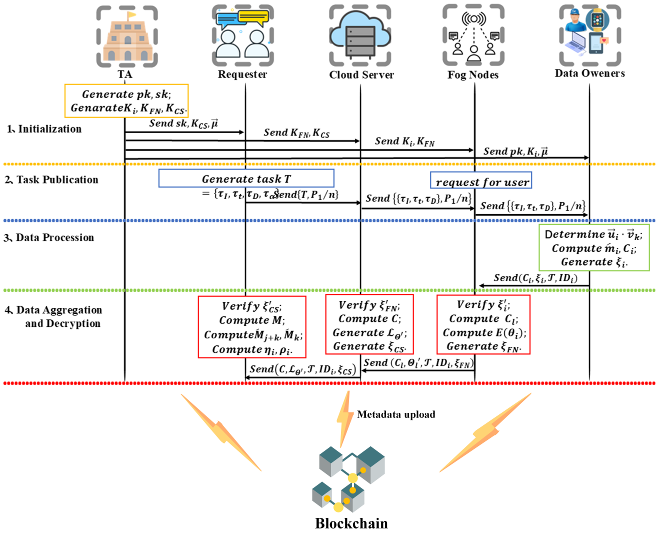

TA
Requester
Cloud Server
Fog Nodes
Data Oweners
Generate pk, sk;
GenarateK_i, K_{FN}, K_{CS}.
1、Initialization
Send $sk, K_{CS}, \vec{\mu}$
Send K_{FN}, K_{CS}
Send K_i, K_{FN}
Send $pk, K_i, \vec{\mu}$
2、Task Publication
Generate task T
$= \{\tau_I, \tau_t, \tau_D, \tau_\alpha\}$
Send$\{T, P_1/n\}$
Send $\{\{\tau_I, \tau_t, \tau_D\}, P_1/n\}$
request for user
Send $\{\{\tau_I, \tau_t, \tau_D\}, P_1/n\}$
3、Data Procession
Determine $\vec{u}_i \cdot \vec{v}_k$;
Compute $\acute{m}_i, C_i$;
Generate ξ_i.
Send$(C_i, \xi_i, \mathcal{T}, ID_i)$
4、Data Aggregation and Decryption
Verify ξ'_{CS};
Compute M;
Compute$\acute{M}_{j+k}, \acute{M}_k$;
Compute η_i, ρ_i.
Verify ξ'_{FN};
Compute C;
Generate $\mathcal{L}_{\Theta'}$;
Generate ξ_{CS}.
Verify ξ'_i;
Compute C_l;
Compute $E(\theta_i)$;
Generate ξ_{FN}.
Send $(C_l, \Theta_l', \mathcal{T}, ID_i, \xi_{FN})$
Send$(C, \mathcal{L}_{\Theta'}, \mathcal{T}, ID_i, \xi_{CS})$
Metadata upload
Blockchain

Fig. 2. Overview of our QSDA scheme.

B. Detailed Construction of QSDA

1) System Initialization

Step 1: The Trusted Authority (TA) selects security parameters (k_0, k_1, k_2) such that $k_1 \ll k_2 < k_0$; chooses two large primes p and q (satisfying $|p| = |q| = k_0$), computes $N = p \times q$, and ensures that $g \in \mathbb{Z}_N^*$ is a generator with order $\phi(N) = (p-1)(q-1)$; generates a random blinding factor $|\mathcal{L}| = k_2$ with $\mathcal{L} \in \mathbb{Z}_N^*$, the private key is $sk = (p, \mathcal{L})$. Two encryptions of zero $Enc(0)$ are generated as part of the public key: $Enc(0)_1$ and $Enc(0)_2$. The public key is constructed as

$$pk = \big(Enc(0)_1, Enc(0)_2, N, k_0, k_1, k_2\big),$$

and is publicly distributed to FNs and users. The public key pk enables all users to encrypt data, while only the requester holding the private key sk can decrypt the data.

Step 2: Each user generates a vector from their location coordinates (x_i, y_i) as $\vec{u}_i = \big(x_i^2 + y_i^2, -2x_i, -2y_i, 1\big)$. To support encryption and aggregation of multi-dimensional medical data, a superincreasing vector of $2j+1$ positive integers is generated: $\{\mu_0 = 1, \mu_1, \mu_2, \ldots, \mu_j, \mu_{j+1}, \cdots, \mu_{2j}\}$ with the constraint:

$$\sum_{j=1}^{i-1} \mu_j \cdot 2^{2v} \cdot n < \mu_i < N,$$

where j is the maximum dimension of the medical data, and $2v$ is the maximum bit-length per dimension.

Step 3: The TA generates unique HMAC symmetric keys for each entity in the system. For each user DO_i, a key K_i is generated and securely distributed to the user and their associated FNs. For each FN FN_l, a key K_{FN} is generated and distributed to the FN and CS. For the CS, a key K_{CS} is generated and distributed to the CS and the requester. The hash values of all keys are recorded on the blockchain to ensure auditability and tamper resistance during key distribution.

Step 4: The blockchain runs mechanisms to complete the initialization process.The TA records the public key information of all entities into the blockchain system for public query by other entities. Subsequently, all key metadata generated in the system, including user multi-dimensional data, timestamps, local aggregated data, digital signatures, unique identification codes, offset index lists, and encrypted region constraint hashes, are hashed and stored on-chain.

2) Task Publication and User Recruitment

After system initialization, each requester re sets a task $T = \{\tau_I, \tau_t, \tau_D, \tau_\alpha\}$, where τ_I, τ_t, τ_D, and τ_α represent the unique task identifier, task deadline, detailed task description , and geographic location constraints, respectively.In this phase, the requester specifies the task region τ_α, modeling the irregular polygon area as a union of multiple circular regions. We define the set $\mathbb{S} = \{\vec{v}_1, \vec{v}_2, \ldots, \vec{v}_n\}$ as the n circular region constraints for the task domain, where a user needs to be located within any one circle to participate. We define the set $\mathbb{S} = \{\vec{v}_1, \vec{v}_2, \ldots, \vec{v}_n\}$ as the n circular region constraints for the task domain, where a user needs to be located within any one circle to participate. To prevent replay attacks, the requester assigns each task a unique timestamp $\mathcal{T}$.

When publishing the task, the requester pre-pays rewards $\mathbb{P} = (P_1, P_2)$, where P_1 is a fixed prepaid reward, and P_2 is a variable reward to incentivize users to provide high-value data. The requester then sends the aggregation request $\{T, \frac{P_1}{n}\}$ to the CS, with P_1 distributed as a fixed prepaid reward to all potential users. Upon receiving the task, the CS selects a set of FNs located within τ_α and assigns aggregation tasks $\{\tau_I, \tau_t, \tau_D\}$ to the corresponding FNs. These FNs query their users to determine whether they intend to share their health data for aggregation. The CS distributes rewards to the FNs, which identify contributing users and distribute rewards accordingly. Each user receiving the task first confirms whether their cost co_i satisfies $co_i \leq \frac{P_1}{n}$, where $\frac{P_1}{n}$ is the minimum reward offered by the requester and co_i represents the user's cost for completing task T, including power consumption, computation, and network transmission costs. The calculation of this cost is beyond the scope of this work, and thus the computational overhead is not further considered in this paper. If the condition is met, the user accepts the task T.

Circular Range Determination Method: Each circular region $\mathcal{C}_k$ is defined by center (x_k, y_k) and radius R_k. Using vector operations, the standard equation of a circle, $(x-x_{\mathcal{C}})^2+(y-y_{\mathcal{C}})^2-r^2=0$, can be expressed as the inner product of two vectors: $(x^2+y^2, -2x, -2y, 1)\cdot(1, x_{\mathcal{C}}, y_{\mathcal{C}}, x_{\mathcal{C}}^2+y_{\mathcal{C}}^2-r^2)=0$. Thus, the circle equation can be represented succinctly as $\vec{u}\cdot\vec{v}=0$. Traditionally, task region determination combines Euclidean distance with homomorphic encryption for privacy-preserving location verification, which introduces considerable computational and communication overhead. In this work, we employ the inner product method to determine the positional relationship between points and circles. The requester generates the task range vector $\vec{v}_k = (1, x_k, y_k, x_k^2+y_k^2-R_k^2)$, defining $\vec{v}_k$ as the regional constraint for each circle $\mathcal{C}_k$. A user's location (x, y) corresponds to vector $\vec{u}_i = (x^2+y^2, -2x, -2y, 1)$. If $\vec{v}_k\cdot\vec{u}_i = x^2+y^2-2x_kx-2y_ky+(x_k^2+y_k^2-R_k^2)\leq 0$, the user is inside the circle; otherwise, (x, y) lies outside the circular region $\vec{v}_k$.

3) Data Processing

Each user DO_i decides whether to contribute their data for aggregation. If a user wishes to participate and upload their local data m_i, they execute the following steps:

Step 1: The user performs local location verification by computing the inner product $\vec{u}_i\cdot\vec{v}_k$ between the task vector $\vec{v}_k$ and their own vector $\vec{u}_i$. If $\vec{u}_i\cdot\vec{v}_k\leq 0$, this indicates that the user is located within region C_k, meaning there exists at least one k such that $\vec{u}_i\cdot\vec{v}_k\leq 0$ without revealing the specific k. Users U_i who meet the condition will accept the task.

Step 2: U_i generates a j-dimensional dataset $m_i = \{m_{i,0}, m_{i,1}, \dots, m_{i,j}\}$. Then, the user aggregates these $j+1$ medical data dimensions using a super-increasing vector

$$\vec{\mu} = \{\mu_0 = 1, \mu_1, \mu_2, \dots, \mu_j, \mu_{j+1}, \dots, \mu_{2j}\}$$

to obtain a single aggregated value $\hat{m}_i$. The aggregation is given by

$$\begin{aligned}\hat{m}_i &= \mu_0 m_{i,0} + \mu_1 m_{i,1} + \mu_{j+1} m_{i,1}^2 + \cdots + \mu_j m_{i,j} + \mu_{2j} m_{i,j}^2\\ &= \mu_0 m_{i,0} + \sum_{k=1}^{j}\left(\mu_k m_{i,k} + \mu_{j+k} m_{i,k}^2\right).\end{aligned}$$

Here, $m_{i,0}$ denotes the identity information of U_i; $m_{i,k}$ $(1\leq k\leq j)$ is the k-th dimension of medical data collected by U_i; and μ_k, μ_{j+k} are elements of the super-increasing sequence corresponding to the k-th data dimension and its squared term, respectively.

Step 3: U_i selects random values $r_1, r_2 \in \{0,1\}^{k_2}$ and encrypts $\hat{m}_i$ using the SHE public key encryption scheme, generating ciphertext:

$$C_i = Enc(\hat{m}_i) = \hat{m}_i + r_1\cdot Enc(0)_1 + r_2\cdot Enc(0)_2 \pmod N.$$

Step 4: U_i generates a data signature using the HMAC key K_i, $\xi_i = \mathrm{HMAC}\big(K_i, C_i\|\mathcal{T}\|ID_i\big)$, and uploads the data packet $(C_i, \xi_i, \mathcal{T}, ID_i)$ to the FNs.

4) Data Aggregation

a. Fog Node Aggregation

Step 1: Upon receiving data, the FN recalculates the HMAC using key K_i, $\xi_i' = \mathrm{HMAC}\big(K_i, C_i\|\mathcal{T}\|ID_i\big)$, and verifies if ξ_i matches ξ_i'. Invalid data packets are discarded, and the valid user list is recorded to ensure only legitimate data is aggregated.

Step 2: The FN performs homomorphic addition of all ciphertexts within its jurisdiction to obtain the local ciphertext aggregate. It first counts the number of valid data points n, then computes C_l and the local mean ciphertext $\bar{C}_l$:

$$C_l = \sum_{i=1}^{n} C_i \pmod N, \bar{C}_l = \frac{1}{n}\sum_{i=1}^{n} C_i \pmod N.$$

The offset for each user is computed as the ciphertext distance:

$$Enc(\theta_i) = dist(C_i, \bar{C}_l) = (C_i - \bar{C}_l)^2 = C_i^2 - 2C_i\bar{C}_l + \bar{C}_l^2.$$

Step 3: The FN generates a positive random mask r_i for each offset ciphertext and computes the masked offset as $\Theta_i = Enc(\theta_i + \epsilon + r_i)$.

These offsets are arranged into a list Θ_l' indexed by users under the l-th FN. The FN also uploads the hash of the aggregated ciphertext and the identifier of the aggregated user set to the blockchain, which generates a unique record ID for subsequent verification and auditing.

Step 4: The FN signs the data using HMAC key K_{FN}, $\xi_{FN} = \mathrm{HMAC}\big(K_{FN}, C_l\|\Theta_l'\|\mathcal{T}\|ID_i\big)$, uploads the masked random values $E(r)$ to the blockchain, and sends the package $(C_l, \Theta_l', \mathcal{T}, ID_i, \xi_{FN})$ to the CS.

b. Cloud Server Aggregation

Step 1: After receiving messages from all FNs, the CS verifies each FN's signature by recomputing, $\xi_{FN}' = \mathrm{HMAC}\big(K_{FN}, C_l\|\Theta_l'\|\mathcal{T}\|ID_i\big)$, and compares with the received ξ_{FN}. Packets failing verification are discarded.

Step 2: The CS aggregates all locally aggregated ciphertexts by homomorphic addition $C = \sum_{l=1}^{m} C_l \pmod N$, where C_l is the l-th FN's local aggregate ciphertext, and m is the number of FNs.

Step 3: The CS merges all user offset ciphertext lists from the FNs into a global offset list $\mathcal{L}_{\Theta'} = \bigcup_{l=1}^{m}\Theta_l'$.

Step 4: Using HMAC key K_{CS}, the CS generates a signature, $\xi_{CS} = \mathrm{HMAC}\big(K_{CS}, C\|\mathcal{L}_{\Theta'}\|\mathcal{T}\|ID_i\big)$, and returns $(C, \mathcal{L}_{\Theta'}, \mathcal{T}, ID_i, \xi_{CS})$ to the requester.

c. Blockchain-based Auditing

To support traceability and verifiability, the system records critical metadata on the blockchain during this stage. The cloud server generates a hash digest of the aggregated ciphertexts, the merged offset list, and the corresponding signatures, and submits these hash records to the blockchain as immutable audit evidence.

5) Data Decryption

The requester can obtain correct aggregated medical data only if the returned aggregated data passes validity verification.

Step 1: Upon receiving the aggregated data package from the CS, the requester recomputes the HMAC using the shared key K_{CS} and verifies whether $\xi'_{CS} = \mathrm{HMAC}\big(K_{CS},\, C\|\mathcal{L}_{\Theta'}\|\mathcal{T}\|ID_i\big)$ holds. If $\xi'_{CS} \neq \xi_{CS}$, represents verification fails, the system can trigger a smart contract to halt incentives or mark malicious nodes, ensuring a trustworthy and auditable data aggregation service.

Step 2: The requester decrypts using the SHE private key $sk = (p, \mathcal{L})$. Suppose each user's ciphertext has the form $C_i = \hat{m}_i + p \cdot r_i \pmod N$, then the aggregated ciphertext C is represented as $C = M + p \cdot R$. The requester computes $M = D(C) = C \bmod p = (M + p \cdot R) \bmod p = M \bmod p$, recovering the aggregated plaintext M.

Step 3: Using a recursive algorithm, the requester calculates

$$M_{j+k} = \frac{M - (M \bmod \mu_{j+k})}{\mu_{j+k}}, \quad M_k = \frac{M - (M \bmod \mu_k)}{\mu_k},$$

where $M_k = \sum_{i=1}^{n} m_{i,k}, \quad M_{j+k} = \sum_{i=1}^{n} m_{i,k}^2, \quad k = 1, 2, \ldots, j$.

Thus, the requester recursively obtains each dimension's aggregated medical data and corresponding squared sums: $\{M_1, \ldots, M_j, M_{j+1}, \ldots, M_{2j}\}$. Algorithm 1 details the recovery process.

Next, the requester computes the average and variance for each dimension $k = 1, 2, \ldots, j$ using

$$AVE_k = \frac{M_k}{n}, \quad VAR_k = \frac{M_{j+k}}{n} - (AVE_k)^2.$$

Algorithm 1 Aggregated Medical Data Recovery at Requester

```
Input: Aggregated plaintext $M$ and super-increasing vector $\vec{\mu} = \{\mu_0 = 1, \mu_1, \ldots, \mu_j, \mu_{j+1}, \ldots, \mu_{2j}\}$
Output: $\{M_1, \ldots, M_j, M_{j+1}, \ldots, M_{2j}\}$
for $k = j$ down to $1$ do
    $M_{j+k} \leftarrow \frac{M - (M \bmod \mu_{j+k})}{\mu_{j+k}}$
    $M \leftarrow M - \mu_{j+k} \cdot M_{j+k}$
end for
for $k = j$ down to $1$ do
    $M_k \leftarrow \frac{M - (M \bmod \mu_k)}{\mu_k}$
    $M \leftarrow M - \mu_k \cdot M_k$
end for
return $\{M_1, \ldots, M_j, M_{j+1}, \ldots, M_{2j}\}$
```

Step 4: Reward Calculation. Using the offset ciphertext list and masks retrieved from the blockchain, compute

$$Enc(\theta_i + \epsilon) = Enc(\theta_i + \epsilon + r_i) - E(r_i),$$

where ϵ is a small positive number preventing division by zero when $\theta_i = 0$. A smaller θ_i indicates higher reliability of sensed data m_i.

After decrypting $\theta_i + \epsilon$, compute $\frac{1}{\theta_i + \epsilon}$ and sum over all users: $\sum_{i=1}^{n} \frac{1}{\theta_i + \epsilon}$. Each user's data quality [26] is then calculated as:

$$\eta_i = \frac{\frac{1}{\theta_i + \epsilon}}{\sum_{i=1}^{n} \frac{1}{\theta_i + \epsilon}}.$$

The reward for each user is computed as $\rho_i = \frac{P_1}{n} + P_2 \cdot \eta_i$, and securely distributed according to user IDs in the offset list. This incentivizes high-quality data submission. Algorithm 2 presents the reward allocation process.

Algorithm 2 Reward Allocation

```
Input: Requester inputs $\{\mathcal{L}_{\Theta'}, (P_1, P_2)\}$ and $\epsilon$; CS input $\frac{P_1}{n}$
Output: User rewards $\{\rho_i\}$
for $i = 1$ to $n$ do
    FN computes offset ciphertext $Enc(\theta_i) \leftarrow dist(C_i, \bar{C}_l)$ and $\frac{1}{\theta_i + \epsilon}$
end for
for $i = 1$ to $n$ do
    Requester calculates data quality:
        $\eta_i = \frac{\frac{1}{\theta_i + \epsilon}}{\sum_{i=1}^{n} \frac{1}{\theta_i + \epsilon}}$
    Requester calculates variable reward:
        $\rho_i = \frac{P_1}{n} + P_2 \cdot \eta_i$
    Requester sends $\{i, \rho_i\}_{i=1}^{n}$ to CS
end for
return $\{\rho_i\}_{i=1}^{n}$
```

VI. Security Analysis

To evaluate the effectiveness of the QSDA scheme in privacy protection and security assurance, this section theoretically analyzes and proves the scheme's security properties from perspectives including data encryption, data integrity, identity authentication, and resistance to replay attacks.

Theorem 1. If SHE is secure, our scheme guarantees data confidentiality. Under the proposed threat model, even in the presence of malicious FNs or collusion between the CS and FNs, adversaries cannot recover the original plaintext of any individual user.

Proof: Each user's data is first encrypted using SHE and then encoded into a multi-dimensional structure with the extended super-increasing sequence. During encryption, the user randomly selects high-entropy factors r_1, r_2, and the ciphertext is expressed as:

$$C_i = Enc(\hat{m}_i) = \hat{m}_i + r_1 Enc(0)_1 + r_2 Enc(0)_2 \pmod N.$$

Due to the randomness introduced, adversaries cannot establish valid plaintext–ciphertext mappings nor perform chosen-plaintext attacks. Thus the ciphertext satisfies IND-CPA security. Even if the cloud colludes with FNs and attempts differential analysis by combining aggregated results with offset lists, the attack will fail: (i) offsets merely reflect data quality differences and are non-reversible; (ii) the super-increasing sequence couples multi-dimensional data, preventing independent decoding of a single dimension; (iii) the randomness of SHE masks correlations between ciphertexts, rendering differential analysis ineffective. Therefore, adversaries cannot recover users' original plaintext, and the scheme guarantees data confidentiality.

Theorem 2 . In the proposed scheme, even if the requester obtains the decrypted aggregation result together with the plaintext offsets of all users, it is still impossible to recover the original data of any individual user.

Proof: In QSDA, the FN uploads aggregated ciphertext summations of multiple users:

$$C_l = \sum_{i=1}^{n} C_i \pmod N \implies M = \mathrm{Dec}(C_l) = \sum_{i=1}^{n} \hat{m}_i.$$

Based on the security of the super-increasing sequence, each user's private value is indivisible in the dimensional space. Therefore, even if the requester possesses the aggregated sum, it cannot separate any individual $\hat{m}_i$. Furthermore, since user data are randomized before encryption, the ciphertexts satisfy IND-CPA security, which prevents differential or inference-based attacks. Hence, the scheme guarantees data integrity and verifiability.

Theorem 3. Under a semi-honest model, assuming the encryption mechanism satisfies semantic security (IND-CPA), no unauthorized entity can obtain users' real geographic locations during task filtering, thus achieving location privacy protection.

Proof: User location privacy is ensured through inner product computation. Specifically, the task region constraint is transformed into an inner product operation between the user vector and the task vector. Users only submit the result of the inner product without revealing their actual coordinates. Since only the inner product value is provided, adversaries cannot infer a user's location from a single computation. Therefore, the scheme effectively prevents geographic location privacy leakage.

Theorem 4. Attackers cannot impersonate other users to submit forged data packets. If an attacker tampers with ciphertexts or forges uploads, signature verification will detect and reject the packets.

Proof: Each user registers a unique key K_i, binding signature structure to identity ID_i and timestamp TS_i. Upon upload, users sign their encrypted data:

$$\mathrm{Sig}_i = \mathrm{HMAC}_{K_i}(ID_i \| TS_i \| C_i).$$

On receipt, the system verifies the signature; forged ciphertext C_i' fails verification:

$$\mathrm{Verify}(C_i', \mathrm{Sig}_i) = 0.$$

Since attackers do not know K_i, forged signatures cannot pass HMAC verification, ensuring data integrity and trustworthiness.

Theorem 5. Attackers cannot interfere with current aggregation by replaying valid ciphertexts from previous tasks.

Proof: Each upload embeds the current task timestamp TS_i in the signature. The system compares the received TS_i with the current timestamp TS_{current} and rejects if

$$TS_i < TS_{\mathrm{current}} \quad \text{or} \quad (ID_i, TS_i) \in T,$$

indicating replay. QSDA records offset hash summaries on the blockchain and generates unique task IDs for aggregation requests, preventing attackers from bypassing validation by altering timestamps. This defends against replay and timing pollution attacks.

In summary, the proposed scheme satisfies security requirements in data confidentiality, verifiability, incentive security, and location privacy.

VII. Experiments and Evaluation

A. Functional Comparison

To evaluate the comprehensiveness of the QSDA scheme, we compare it with the schemes proposed by Shen et al. [14], Yan et al. [15], Zhao et al. [19], and Tang et al. [22] based on six core functionalities. The results are shown in Table II.

Overall, QSDA demonstrates comprehensive advantages across six functional dimensions. Specifically, it ensures data confidentiality against collusion and differential analysis, supports verifiable aggregation results with transparent auditing, and preserves user location privacy through non-interactive verification. In addition, the scheme achieves fair incentive allocation by linking rewards to data quality, and enables multi-source data integration across devices and regions. Compared with existing approaches, QSDA not only satisfies the fundamental design objectives but also achieves superior practicality, completeness, and adaptability in HIoT scenarios.

B. Performance Analysis

Table III presents the computational overhead of each entity in the QSDA scheme. Let T_{Location}, T_{EnSHE}, and T_{Sign} denote the computation overhead for inner product location judgment, SHE encryption, and signature computation, respectively. The total computation cost for a user is given by:

$$T_{\text{User}} = T_{\text{Location}} + T_{\text{EnSHE}} + T_{\text{Sign}} = 6T_{\text{Mul}} + 5T_{\text{Add}} + T_{\text{Sign}}.$$

Assume that a FN receives data from n users. Let T_{Agg} and T_{Square} denote the overhead for ciphertext aggregation and squaring, respectively. The total computational overhead for an FN is:

$$\begin{aligned} T_{\text{FN}} &= (n+1)T_{\text{Sign}} + T_{\text{Agg}} + 4T_{\text{Mul}} + 4T_{\text{Add}} + 2T_{\text{Square}} \\ &= (n+1)T_{\text{Sign}} + 4T_{\text{Mul}} + (n+3)T_{\text{Add}} + 2T_{\text{Square}}. \end{aligned}$$

Assume the CS receives data from m FNs. The computation overhead for the CS is:

$$T_{\text{CS}} = (m+1)T_{\text{Sign}} + T_{\text{Agg}} = (m+1)T_{\text{Sign}} + (m-1)T_{\text{Add}}.$$

The task requester Re needs to generate the task request, the total computational overhead for the requester is:

$$T_{\text{Re}} = T_{\text{Sign}} + T_{\text{DeSHE}} + nT_{\text{Add}} = T_{\text{Sign}} + 2T_{\text{Mod}} + nT_{\text{Add}},$$

where T_{DeSHE} consists of two modular operations T_{Mod} due to the decryption process.

C. Experimental Evaluation

To intuitively quantify the computational burden of the QSDA scheme across different entities, we constructed a complete experimental environment. The implementation was carried out using Python 3.9 and the Pyfhel library. Experiments were conducted on a Windows 11 platform equipped with an Intel i5-13500H processor and 32 GB of RAM. The setup simulated a city-scale deployment environment, consisting of 100 to 500 users, 10 to 50 FNs, and a CS. The focus of the evaluation is on the computational consumption of three key roles: users, FNs, and the CS during the data aggregation process.

TABLE II
FUNCTIONAL COMPARISON WITH OTHER SCHEMES

Scheme	Confidentiality	Verifiable	Location Privacy	Auditing	Incentive	Multi-Source
Shen et al. [14]	✓	✓	×	×	×	×
Yan et al. [15]	✓	✓	×	×	×	✓
Zhao et al. [19]	✓	×	×	×	×	✓
Tang et al. [22]	✓	×	×	×	✓	×
Our Scheme	✓	✓	✓	✓	✓	✓

TABLE III
COMPUTATION OVERHEAD OF ENTITIES IN QSDA SCHEME

Entity	**Computation Overhead**
User	$6T_{\text{Mul}} + 5T_{\text{Add}} + T_{\text{Sign}}$
Fog Node (FN)	$(n{+}1)T_{\text{Sign}} + 4T_{\text{Mul}} + (n{+}3)T_{\text{Add}} + 2T_{\text{Sq}}$
Cloud Server (CS)	$(m{+}1)T_{\text{Sign}} + (m{-}1)T_{\text{Add}}$
Task Requester (Re)	$nT_{\text{Add}} + T_{\text{Sign}} + 2T_{\text{Mod}}$

Experimental Parameters: The number of users connected to each FN, denoted as n, was incremented from 100 to 500 in steps of 100. The number of FNs m was increased from 10 to 50 in steps of 10. The number of data dimensions j for each user during each aggregation round was varied from 2 to 10 in steps of 2. Additionally, the bit lengths of each entity's identifier, timestamp, task description, task region constraint, SHE key, and ciphertext were set to 32 bits, 64 bits, 1600 bits, 4096 bits, 32768 bits, and 4096 bits, respectively.

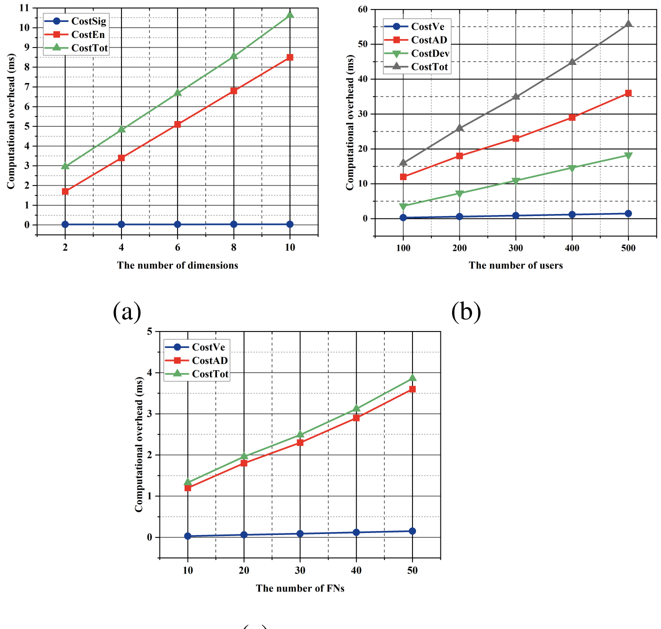
(1) Computational Overhead

The computational overhead of the requester includes signature verification, SHE decryption, and reverse extraction with addition operations on the aggregated data, resulting in a total cost of approximately 21.3 ms.

Fig. 3(a) illustrates the average user-side encryption time (*CostEn*), signature computational overhead (*CostSig*), and total computational overhead per user (*CostTot*) as functions of the data dimension j. For this evaluation, the data dimension per user was set to $j \in \{2, 4, 6, 8, 10\}$. As j increases, the user's computational burden grows approximately linearly.

To better demonstrate the FN's computational overhead, we fixed the data dimension at $j = 10$ and incrementally increased the number of users $n \in \{100, 200, 300, 400, 500\}$. Fig. 3(b) shows the variation in computation time at the FN levels. At the FN side, the aggregation cost grows linearly with the number of users and includes the cost of signature verification (*CostVe*), data aggregation (*CostAD*), offset calculation (*CostDev*), and total cost (*CostTot*). When $n = 500$, the FN's signature verification cost was 1.5 ms, the offset computational overhead was 18.235 ms, the aggregation cost was 36 ms, and the total overhead amounted to 55.735 ms.

Lastly, we fixed the total number of users at 500 and the data dimension at $j = 10$, while varying the number of FNs $m \in \{10, 20, 30, 40, 50\}$ to evaluate the computational overhead at the CS. The results, depicted in Fig. 3(c), show that the CS's signature verification (*CostVe*) and aggregation cost (*CostAD*) increase linearly with m, with total computation time rising from 1.33 ms to 3.86 ms. These results verify that the scheme maintains good scalability and computational efficiency even under large-scale FN integration.



(a) (b)

(c)

Fig. 3. The variation of computational overhead with the number of dimensions, users, and FNs.

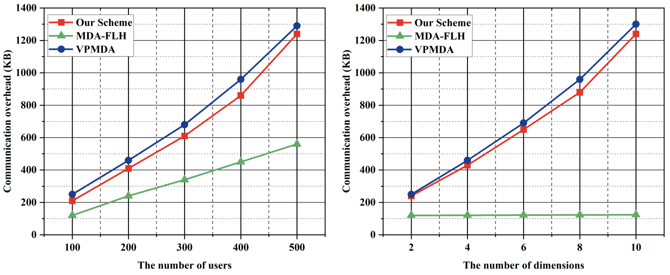
(2) Communication Overhead

According to the experimental parameter settings, the lengths of key fields are defined as follows: entity identifier 32 bits, timestamp 64 bits, task description 1600 bits, task region constraint 4096 bits, and single ciphertext 4096 bits. This subsection focuses on evaluating the communication overhead during the data aggregation phase. Assuming the data dimension $j = 10$, the number of users $n = 500$, and the number of FNs $m = 50$, in this representative large-scale HIoT scenario, each user uploads encrypted data along with a signature, resulting in a total communication volume of 1.27 MB. Each FN forwards and aggregates the data from its associated users and uploads the offset list and local aggregation results, incurring an overhead of 322.5 KB. The CS further collects the aggregated data from all FNs to complete global aggregation, with a communication cost of 59.0 KB. These results demonstrate that QSDA, by employing super-increasing sequence encoding and lightweight signature mechanisms, effectively controls the communication burden of each entity while supporting multi-dimensional data aggregation and secure verification, thereby achieving good scalability and practical performance.

(3) Scheme Comparison

Since both FNs and the CS are responsible for data aggregation, their computational overhead is primarily determined by the number of connected entities. Therefore, this comparison emphasizes the performance at the user side and the FN side. We compare the proposed QSDA scheme with two existing schemes: VPMDA [27] and MDA-FLH [17], which are designed for IoT-based healthcare data and smart grid applications, respectively.

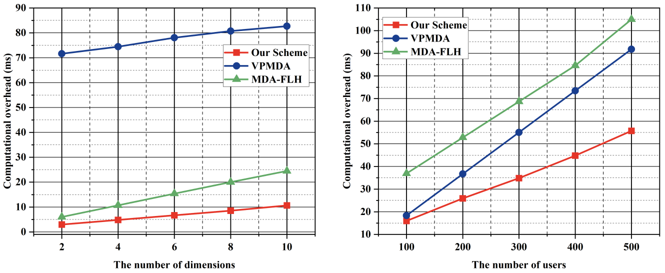
As shown in Fig. 4. In the communication overhead experiments, MDA-FLH employs CRT encoding to merge multi-dimensional data into a single ciphertext, thereby achieving lower communication cost compared with traditional per-dimension encryption schemes. VPMDA, on the other hand, supports multi-type data aggregation and verification, but incurs higher communication overhead. Fig. 4(a) illustrates the trend of communication overhead with respect to the number of users when the data dimension is fixed at $j = 10$. It can be observed that the overhead of all three schemes grows linearly with the number of users. QSDA consistently incurs lower overhead than VPMDA and slightly higher than MDA-FLH, demonstrating that it maintains a relatively low communication burden while supporting multi-dimensional statistical operations and secure verification. Fig. 4(b) shows the communication overhead as the data dimension increases with $n = 500$. As the dimension grows, the overhead of VPMDA increases rapidly, while QSDA also grows linearly but with a significantly smaller slope; MDA-FLH remains almost stable. Overall, QSDA achieves a balanced trade-off between functionality and communication efficiency, effectively controlling communication overhead while ensuring multi-dimensional statistical computation and privacy protection.



(a) (b)

Fig. 4. Comparison of the communication overhead.

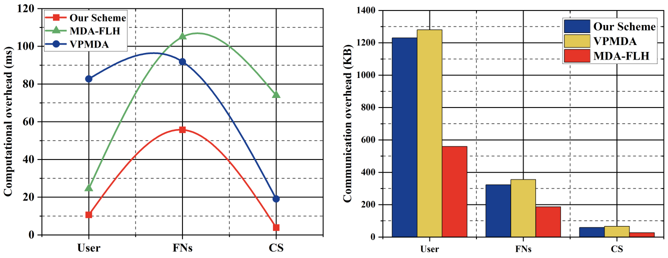
As shown in Fig. 5, QSDA adopts SHE, a homomorphic encryption scheme supporting addition, which demonstrates lower ciphertext expansion and faster ciphertext addition compared to the commonly used Paillier cryptosystem. Specifically, SHE enables homomorphic addition through modular arithmetic and noise management, requiring only a single modular addition to complete ciphertext aggregation. In contrast, Paillier encryption necessitates more complex modular exponentiation, resulting in significantly higher computational overhead at both the user and FN sides. In Paillier encryption, if the modulus length is 2048 bits, the ciphertexts of approximately 4096 bits. For higher security parameters, the ciphertext length expands correspondingly to 6144 or 8192 bits. This makes Paillier ciphertexts generally larger in size and more costly in transmission. In contrast, the ciphertext length of SHE remains fixed at 4096 bits, which provides additive homomorphism while significantly reducing communication overhead.



(a) (b)

Fig. 5. Comparison of the computational overhead of users, FNs.

Authentication Efficiency: QSDA employs a lightweight HMAC for data authentication. Compared to schemes such as VPMDA and MDA-FLH, which utilize BLS signatures based on bilinear pairings, HMAC offers superior computational efficiency.



(a) (b)

Fig. 6. Comparison of the computational and communication overhead.

Fig. 6 present the comparative computational and communication costs for each scheme when $j = 10$, $n = 500$, and $m = 50$. At the user side, all three schemes involve encryption and signing of multi-dimensional data. The MDA-FLH scheme employs Paillier encryption combined with CRT encoding, encrypting each dimension independently and embedding blind factors for security, resulting in an average encryption overhead of approximately 11.6 ms. The VPMDA scheme encrypts both the data and its squared values simultaneously, attaching BLS signatures and random masks, leading to the highest upload cost of 65.3 ms. In contrast, QSDA utilizes a super-increasing encoding to transform multi-dimensional data into a single structured ciphertext, combined with lightweight SHE encryption and HMAC signing, achieving a total overhead of only 10.627 ms. This significantly reduces computational pressure on the user side. At the FN side, MDA-FLH not only performs homomorphic aggregation but also needs to reconstruct missing user ciphertexts caused by network delays or failures, involving multiple rounds of blind factor verification and modular operations, resulting in an average computation overhead of about 67.2 ms. Although VPMDA adopts a more compact aggregation structure, it still incurs a high overhead of up to 91.8 ms due to multiple verifications,including signatures, time-series hashing, and variance estimation. By comparison, QSDA incorporates Euclidean distance encoding into its offset-based incentive mechanism, requiring FNs to execute only one homomorphic aggregation and one offset encryption, with computation overhead controlled at 55.74 ms. This maintains compatibility with multi-dimensional incentive mechanisms while achieving excellent execution efficiency.

In summary, QSDA demonstrates lower computation overhead across all computational entities. Compared with MDA-FLH and VPMDA, QSDA achieves a superior trade-off between security and efficiency while fulfilling equivalent functional requirements, making it particularly suitable for deployment in resource-constrained applications such as medical IoT and remote health monitoring.

VIII. Conclusion

This paper proposes QSDA, a privacy-enhanced multidimensional data aggregation scheme for the Healthcare Internet of Things. By combining lightweight homomorphic encryption with extended super-increasing sequence encoding, the scheme enables efficient aggregation of multi-dimensional health data in the ciphertext domain while supporting common statistical operations such as mean and variance. For privacy protection, QSDA incorporates dynamic location verification via inner product, a quality-driven incentive mechanism, and a blockchain-based auditing process to ensure data privacy, location privacy, and fairness of incentives. Both theoretical analysis and experimental results demonstrate that QSDA achieves strong security, high efficiency, and good scalability, providing a practical solution for privacy-preserving data aggregation. Future work will further optimize blockchain auditing for scalable and efficient deployments.

References

[1] J. Shen, Z. Gui, X. Chen, J. Zhang, and Y. Xiang, "Lightweight and certificateless multi-receiver secure data transmission protocol for wireless body area networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 3, pp. 1464–1475, 2020.

[2] X. Zhang, L. Wu, L. Xu, Z. Liu, Y. Su, H. Wang, and W. Meng, "Privacy-preserving and verifiable multi-task data aggregation for iot-based healthcare," *Journal of Information Security and Applications*, vol. 89, p. 103977, 2025.

[3] J. Zhao, Y. Zheng, H. Huang, J. Wang, X. Zhang, and D. He, "Lightweight certificateless privacy-preserving integrity verification with conditional anonymity for cloud-assisted medical cyber–physical systems," *Journal of Systems Architecture*, vol. 138, p. 102860, 2023.

[4] X. Zhang, L. Wu, Z. Liu, H. Wang, L. Xu, S. Zhang, and R. Lu, "Towards auditable and privacy-preserving online medical diagnosis service over cloud," *IEEE Transactions on Services Computing*, vol. 17, no. 6, pp. 4397–4410, 2024.

[5] X. Zhang, C. Huang, D. Gu, J. Zhang, J. Xue, and H. Wang, "Privacy-preserving statistical analysis over multi-dimensional aggregated data in edge computing-based smart grid systems," *Journal of systems architecture*, vol. 127, p. 102508, 2022.

[6] Y. Cheng, J. Ma, Z. Liu, Z. Li, Y. Wu, C. Dong, and R. Li, "A privacy-preserving and reputation-based truth discovery framework in mobile crowdsensing," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 6, pp. 5293–5311, 2023.

[7] Y. Zheng, R. Lu, H. Zhu, S. Zhang, Y. Guan, J. Shao, F. Wang, and H. Li, "Setrknn: Efficient and privacy-preserving set reverse knn query in cloud," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 888–903, 2022.

[8] A. Alsharif, M. Nabil, A. Sherif, M. Mahmoud, and M. Song, "Mdms: Efficient and privacy-preserving multidimension and multisubset data collection for ami networks," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10 363–10 374, 2019.

[9] C. Regueiro, I. Seco, S. De Diego, O. Lage, and L. Etxebarria, "Privacy-enhancing distributed protocol for data aggregation based on blockchain and homomorphic encryption," *Information Processing & Management*, vol. 58, no. 6, p. 102745, 2021.

[10] Z. Li, W. Shi, Y.-J. Choi, H. Sekiya, and Q. Deng, "Location and reward privacy-preserving based secure task allocation in mobile crowdsensing," *IEEE Transactions on Mobile Computing*, vol. 24, no. 10, pp. 9951–9964, 2025.

[11] J. Hu, K. Yang, K. Wang, and K. Zhang, "A blockchain-based reward mechanism for mobile crowdsensing," *IEEE Transactions on Computational Social Systems*, vol. 7, no. 1, pp. 178–191, 2020.

[12] Y. Zhang, Q. Chen, and S. Zhong, "Privacy-preserving data aggregation in mobile phone sensing," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 5, pp. 980–992, 2016.

[13] A. Saleem, A. Khan, S. U. R. Malik, H. Pervaiz, H. Malik, M. Alam, and A. Jindal, "Fesda: Fog-enabled secure data aggregation in smart grid iot network," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6132–6142, 2019.

[14] H. Shen, G. Wu, Z. Xia, W. Susilo, and M. Zhang, "A privacy-preserving and verifiable statistical analysis scheme for an e-commerce platform," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 2637–2652, 2023.

[15] X. Yan, W. W. Ng, B. Zhao, Y. Liu, Y. Gao, and X. Wang, "Fog-enabled privacy-preserving multi-task data aggregation for mobile crowdsensing," *IEEE Transactions on Dependable and Secure Computing*, vol. 21, no. 3, pp. 1301–1316, 2023.

[16] J. Liu, H. Wang, J. Bao, R. Sun, X. Du, and M. Guizani, "Rpmda: Robust and privacy-enhanced multidimensional data aggregation scheme for fog-assisted smart grids," *IEEE Internet of Things Journal*, vol. 11, no. 9, pp. 16 021–16 032, 2024.

[17] D. Chen, T. Zhou, W. Liu, R. Li, L. Wu, and X. Yang, "Mda-flh: Multidimensional data aggregation scheme with fine-grained linear homomorphism for smart grid," *IEEE Internet of Things Journal*, vol. 11, no. 2, pp. 3524–3538, 2023.

[18] B. Zhu, Y. Li, G. Hu, and M. Zhang, "A privacy-preserving data aggregation scheme based on chinese remainder theorem in mobile crowdsensing system," *IEEE Systems Journal*, vol. 17, no. 3, pp. 4257–4266, 2023.

[19] J. Zhao, H. Huang, X. Zhang, D. He, K.-K. R. Choo, and Z. L. Jiang, "Vmemda: Verifiable multidimensional encrypted medical data aggregation scheme for cloud-based wireless body area networks," *IEEE Internet of Things Journal*, vol. 11, no. 10, pp. 18 647–18 662, 2024.

[20] J. Zhang, Y. Wang, Z. Ma, X. Yang, Z. Ying, and J. Ma, "A location-aware verifiable outsourcing data aggregation in multiblockchains," *IEEE Internet of Things Journal*, vol. 10, no. 6, pp. 4783–4798, 2022.

[21] T. Li, H. Wang, D. He, and J. Yu, "Designated-verifier aggregate signature scheme with sensitive data privacy protection for permissioned blockchain-assisted iiot," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 4640–4651, 2023.

[22] W. Tang, J. Ren, K. Deng, and Y. Zhang, "Secure data aggregation of lightweight e-healthcare iot devices with fair incentives," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8714–8726, 2019.

[23] R. Yu, A. M. Oguti, D. R. Ochora, and S. Li, "Towards a privacy-preserving smart contract-based data aggregation and quality-driven incentive mechanism for mobile crowdsensing," *Journal of Network and Computer Applications*, vol. 207, p. 103483, 2022.

[24] T. Wang, J. Wang, Q. Yang, B. Yang, H. Li, F. Xu, and Z. Qiao, "An efficient verifiable searchable encryption scheme with aggregating authorization for blockchain-enabled iot," *IEEE Internet of Things Journal*, vol. 9, no. 20, pp. 20 666–20 680, 2022.

[25] J. Lee, J. Oh, D. Kwon, M. Kim, K. Kim, and Y. Park, "Blockchain-enabled key aggregate searchable encryption scheme for personal health record sharing with multidelegation," *IEEE Internet of Things Journal*, vol. 11, no. 10, pp. 17 482–17 494, 2024.

[26] X. Yan, W. W. Ng, B. Zeng, B. Zhao, F. Luo, and Y. Gao, "P 2 sim: Privacy-preserving and source-reliable incentive mechanism for mobile crowdsensing," *IEEE Internet of Things Journal*, vol. 9, no. 24, pp. 25 424–25 437, 2022.

[27] X. Zhang, C. Huang, Y. Zhang, and S. Cao, "Enabling verifiable privacy-preserving multi-type data aggregation in smart grids," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 6, pp. 4225–4239, 2021.