Data Deduplication Method Based on CSP-DLP Asymmetric Homomorphic Encryption Algorithm for High-Density Scenarios

Qiongfeng Mo*, Wei Liao, Hengjian Liao, Jiarong Guo, Xiaodong Feng and Ruyan Guo

Qingyuan Power Supply Bureau of Guangdong Power Grid Co. Ltd., Qingyuan 511500, Guangdong, China
E-mail: Qiongfeng_MoCSG@outlook.com
**Corresponding Author*

Received 11 March 2026; Accepted 14 April 2026

Abstract

With the continuous growth of data scale in high-density scenarios such as the Internet of Things and the Internet of Vehicles, the repeated storage and frequent transmission of massive data not only causes waste of computing and storage resources, but also significantly increases the risk of sensitive information leakage. Therefore, this study innovatively proposes a data deduplication method that integrates asymmetric homomorphic encryption and federated learning. First, a novel asymmetric homomorphic encryption algorithm is designed using the conjugate search problem and the discrete logarithm problem. This algorithm ensures the indistinguishability of ciphertext while providing a cryptographic foundation for data comparability in the ciphertext state, resolving the inherent conflict between privacy protection and data deduplication. Based on this, the proposed encryption algorithm is combined with a federated learning framework to construct an efficient data

Journal of Cyber Security and Mobility, Vol. 15_3, 747–776.
doi: 10.13052/jcsm2245-1439.1539

processing flow that supports ciphertext deduplication, achieving secure identification and filtering of redundant privacy data. The experimental findings reveal that the encryption cost of the introduced encryption algorithm under 128-bit security strength is only 62.5% of the traditional Paillier scheme, and the ciphertext size is reduced by about 42.4%. When conducting deduplication testing in Internet of Vehicles scenarios, the proposed method achieves a duplicate detection rate of 97.4% on a million-level dataset. Moreover, under the condition of maintaining full encrypted processing, the storage requirements are reduced by an average of 38.6%, and the cross-node communication overhead is reduced by about 29.4%. In summary, the proposed method combines high security, high detection rate, and low overhead in high-density scenarios, achieving a balance between privacy protection and data deduplication efficiency. This research provides a scalable, deployable and practical engineering value technology path for privacy data management in the Internet of Things, industrial Internet, smart cities and other fields.

Keywords: Internet of vehicles, conjugate search problem, discrete logarithm problem, homomorphic encryption, federated learning, data deduplication.

1 Introduction

In recent times, with the swift advancement of technologies like cloud computing and AI, information resources have experienced explosive growth. In addition to traditional structured data with obvious structure, there is also a large amount of unstructured information [1, 2]. Therefore, high-density scenarios (Internet of Things (IoT), vehicle-to-everything (V2X)) also place higher demands on the performance and reliability of storage systems. In V2X, data privacy leakage and data redundancy are particularly prominent. How to effectively lower the overhead of data transmission and storage in the system while protecting user data privacy is the key to improving the overall performance and dependability of the system [3, 4]. V2X systems consist of vehicle nodes, roadside units (RSU), and cloud servers. Vehicles continuously collect sensitive data such as location, speed, and status during high-speed movement and upload them to the RSU via open wireless channels. However, the openness of wireless communication makes it easy for privacy data such as vehicle trajectories and identity information to be intercepted or illegally used. On the other hand, V2X generates massive amounts of duplicate data during peak hours (such as similar location information uploaded by multiple

vehicles on the same road segment), leading to wasted storage resources and communication congestion. Therefore, designing a method for deduplicating redundant privacy data for V2X scenarios has significant theoretical and practical value.

Federated learning (FL) provides a new way for collaborative optimization and data sharing of high-density storage systems. FL is a distributed machine learning method that can achieve multi-party joint modeling through encrypted parameter exchange, ensuring that data can be trained without leaving the local device or institution [5]. Currently, homomorphic encryption (HE) has been proven to be applicable to privacy protection in machine learning, so it can also be applied to the FL framework [6]. For example, Cai et al. introduced an FL scheme grounded in multi-key HE. This scheme can protect gradient privacy without losing model accuracy. Experimental results showed that, compared with the FL scheme based on HE, the performance of this scheme was improved by 2 times [7]. Mantey et al. introduced an FL healthcare recommendation system grounded in HE. The results showed that when the encrypted gradient was calculated on the global server side, it had almost no impact on the recommendation results and, at the same time, it could provide an additional secure channel for the transmission of user gradients between servers [8].

Although HE-based FL methods can perform calculations directly without decrypting data, thus providing convenience for clients with insufficient computing power, in high-density scenarios, the repeated storage and frequent transmission of massive amounts of data not only reduces the training efficiency of HE-based FL methods but also affects their privacy protection effect [9, 10]. To meet the urgent need for fast data storage and access, some researchers have proposed data deduplication technology [11]. Data deduplication technology is a data reduction technology that reduces capacity occupation by eliminating duplicate data blocks in the storage system, which can effectively extend the life of storage media and improve system performance [12]. For example, Song et al. proposed a fog-assisted cloud storage encrypted data deduplication scheme. The results showed that the scheme could achieve deduplication within the same data owner and deduplication between different data owners [13]. Zhao et al. proposed a selective local data deduplication method based on content similarity. This method achieved storage space saving of encrypted data through incremental compression and local compression [14]. In addition, Qi et al. proposed a lightweight key-aware encryption deduplication system. The results showed that the system achieved dynamic access control for secure data deduplication [15].

In summary, while existing research on redundant privacy data deduplication has made progress, problems still exist. For example, commonly-used HE methods generally employ noise injection mechanisms to probabilistically obfuscate plaintext data. However, within the FL framework, the introduction of noise can lead to cumulative error. Specifically, noise injected independently by each client accumulates during homomorphic aggregation on the server side. When the number of participants is large and the data density is high, the accumulated noise may exceed the effective range of the gradient signal itself, causing a deviation in the update direction of the aggregated model, ultimately leading to difficulty in model convergence or convergence to a suboptimal solution. Furthermore, encryption mechanisms require ciphertext to be indistinguishable, while data deduplication requires comparability judgment of ciphertext, creating a fundamental conflict. Specifically, encryption algorithms introduce randomization factors to ensure that the same plaintext generates different ciphertexts under different encryption operations, thus resisting chosen-plaintext attacks. Data deduplication requires the ability to accurately determine whether two ciphertexts correspond to the same plaintext, necessitating a definite correspondence between ciphertexts. Traditional encryption schemes achieve security by breaking the comparability between ciphertexts, which fundamentally contradicts the deterministic matching mechanism required for deduplication. To address these issues, this research focuses on V2X networks in high-density scenarios and innovatively proposes the following design. First, the privacy protection requirements of V2X networks are analyzed, and a model for a V2X information retrieval service system that protects both location and identity privacy is designed. The Discrete Logarithm Problem (DLP) is a fundamental problem in public-key encryption algorithms. The Conjugacy Search Problem (CSP) is a major difficult problem used in group theory-based cryptography. Therefore, addressing the data redundancy problem in high-density scenarios, this research proposes an asymmetric HE algorithm based on CSP-DLP, combining CSP and DLP. Finally, by combining the proposed asymmetric HE algorithm with an FL framework, a data deduplication method based on CSP-DLP and FL is proposed. Through this design, the research aims to effectively reduce system storage and communication overhead, alleviate computational and storage pressure in high-density scenarios, and improve the overall efficiency and scalability of FL model training, while ensuring data privacy and system security.

2 Method

2.1 V2X LBS Information Retrieval Service System

The vehicle network system mainly consists of vehicle nodes, RSU and cloud servers [16], as shown in Figure 1.

In Figure 1, in the vehicle network system, vehicle nodes are responsible for collecting various traffic data in real time and transmitting the data to the RSU via wireless communication. The RSU is responsible for further data collection and processing (e.g., HE, deduplication of redundant privacy data) and uploading the processed data to the cloud server. The cloud server then performs aggregation calculation and deep analysis to provide intelligent services. Among them, Location-Based Services (LBS) are one of the most basic intelligent services in the vehicle network [17]. The LBS server stores Point of Interest (POI) data indexed by geographic location [18]. However, wireless communication is open, and POI data is easily intercepted by malicious eavesdroppers during transmission. At the same time, in the absence of trusted constraints, the server may be at risk of illegally using or selling user location information. Therefore, how to ensure that communication data and location privacy are not leaked is the main privacy requirement in the vehicle network LBS service. In response to this privacy requirement, a vehicle network LBS information retrieval service system was designed, as shown in Figure 2.

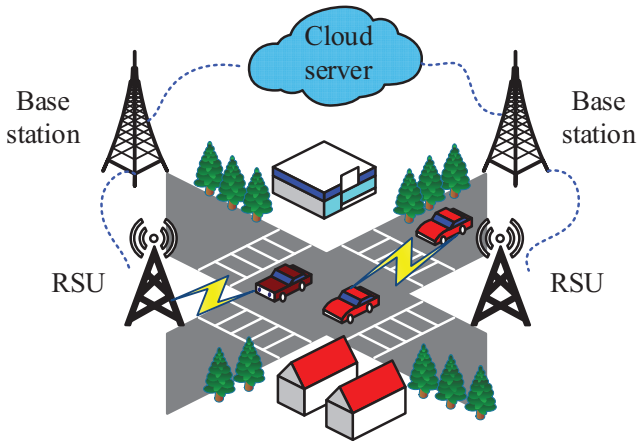

Figure 1 Internet of vehicles system architecture.

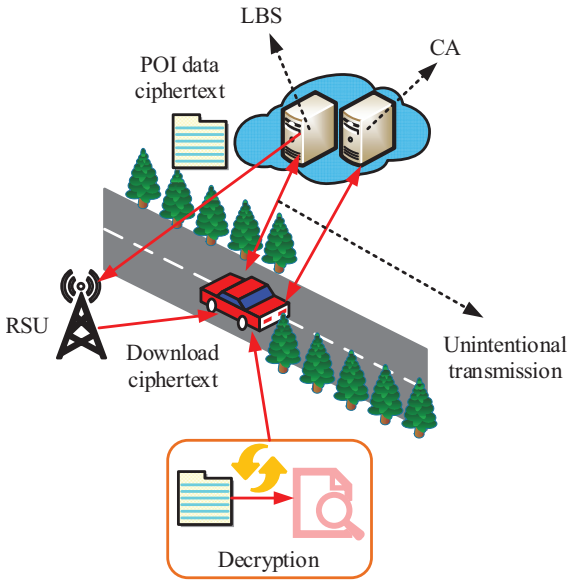

Figure 2 Internet of vehicles LBS information retrieval service system architecture.

In Figure 2, the system comprises an LBS server, RSUs, and vehicle nodes. The LBS server divides the map area into multiple blocks and stores the data at the edge through RSUs within each block. The POI data for each block is encrypted using an HE algorithm and stored in the corresponding RSU. For location privacy protection, the system employs an "inadvertent transmission" protocol to build the LBS information retrieval protocol. When a vehicle user needs to retrieve a POI, the system first encrypts the index of the target location and sends it to the LBS server. The LBS server then processes the location index in encrypted form, interacting with the corresponding RSU using the encryption key to retrieve the requested POI data. For identity privacy protection, the system implements conditional privacy protection through a private key/pseudonym binding mechanism and a digital signature authentication mechanism, as shown in Figure 3.

As shown in Figure 3, after a vehicle user completes registration with the Certificate Authority (CA), the CA generates a unique private key/pseudonym pair for each legitimate vehicle user. After registration, the vehicle user uses the pseudonym as their identifier to interact with the RSU and LBS servers. Each time a message is sent, the vehicle user digitally signs the message using their private key to verify the message's legitimacy. Upon receiving a service request, the RSU and LBS servers first extract the corresponding

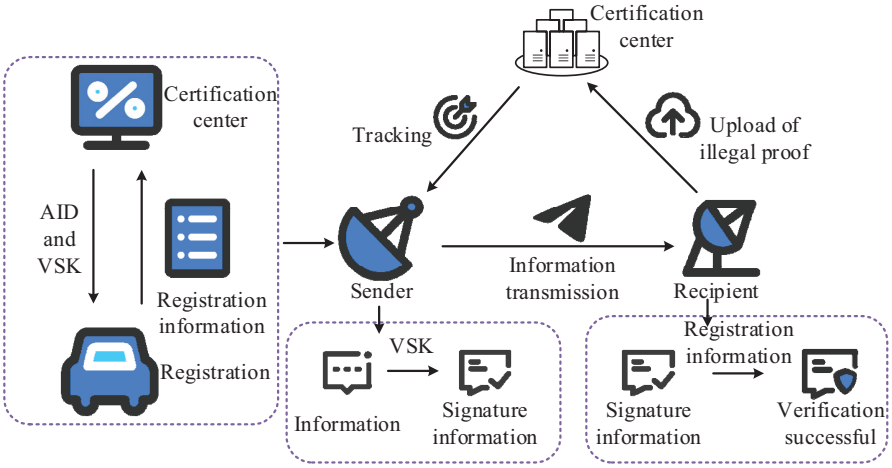

Figure 3 Conditional privacy protection authentication mechanism.

verification parameters based on the pseudonym and verify the message's digital signature. If verification is successful, the system proceeds to subsequent access control and data service processes; if verification fails, it indicates that the message has been tampered with or forged during transmission, and the request will be directly rejected and the abnormal behavior recorded. Through this design, the system can achieve identity authentication without revealing the vehicle's true identity and also supports identity tracing by the CA when necessary.

2.2 Asymmetric HE Algorithm

The privacy protection strategies designed in the V2X LBS information retrieval service system model mainly include HE algorithms, location privacy protection based on unintentional transmission, and conditional privacy protection authentication mechanisms. Traditional HE algorithms usually make security assumptions based on difficult mathematical problems in lattices and achieve plaintext obfuscation by introducing noise mechanisms [19, 20]. Nevertheless, this approach has a large computational overhead and is challenging to meet the real-time processing demands of the Internet of Vehicles. In order to solve the contradiction between data protection and availability, this study adopts CSP and DLP problems as security assumptions for the construction scheme and proposes an asymmetric HE algorithm grounded in CSP-DLP. In non-exchange cryptosystems, DLP ensures the

security of the encryption process, preventing attackers from obtaining plaintext data by reverse-engineering the private key. CSP provides additional security safeguards for the algorithm, effectively addressing potential attacks from quantum computing. In the parameter setting process, the security parameter is set to be K, and the non-commutative group G_{csp}, cyclic group G_{dlp} and their generator g are given through CSP and DLP. The plaintext space is $\Theta = \{0,1\}^n$ and the ciphertext space is $\mathrm{Z} = \{Q\Theta Q^{-1} \mid Q \in G_{csp}\}^n$. In the matrix encoding process, the arbitrary element k to be encoded is first encoded and mapped to the initial matrix E, and then transformed. Four independent random numbers x_1, x_2, x_3, and x_4 are randomly generated, which must satisfy the condition of Equation (1).

$$k = x_1 + x_2 + x_3 + x_4 \tag{1}$$

Next, the study constructs an intermediate matrix E' based on x_1, x_2, x_3, and x_4, as shown in Equation (2).

$$E' = \begin{pmatrix} x_1 & x_2 \\ x_3 & x_4 \end{pmatrix} \tag{2}$$

The constructed encoding matrix E of k based on E' is shown in Equation (3).

$$E = \begin{pmatrix} E' & R_1 & R_2 \\ O & R_3 & R_4 \\ O & O & R_5 \end{pmatrix} \tag{3}$$

In Equation (3), O is a 2×2 zero matrix, and $R_1 \sim R_5$ are matrices randomly sampled from the 2×2 matrix. During the public-private key generation process, firstly, l elements are randomly selected from G_{dlp}, denoted as $k_1 \sim k_l$, and each element is encoded using a matrix encoding function to generate the encoded matrix elements $E_1 \sim E_l$. Secondly, random samples are taken from G_{csp} to construct a 6th-order matrix Q, as shown in Equation (4).

$$Q = \begin{pmatrix} Q_1 & Q_2 & Q_3 \\ Q_4 & Q_5 & Q_6 \\ Q_7 & Q_8 & Q_9 \end{pmatrix} \tag{4}$$

The other half of the public keys $\zeta_1 \sim \zeta_l$ are calculated according to Q as shown in Equation (5).

$$\zeta_1 = QE_1Q^{-1}, \ldots \zeta_l = QE_lQ^{-1} \tag{5}$$

Then, based on the auxiliary matrix T, homomorphic multiplication is performed, and the calculation of T is shown in Equation (6).

$$T = Q \begin{pmatrix} t & R_1 & R_2 \\ O & t & R_3 \\ O & O & t \end{pmatrix} Q^{-1} \tag{6}$$

In Equation (6), t is the matrix of $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$. During the encryption process, the plaintext m is processed by the public key to generate ciphertext C, and the calculation is shown in Equation (7).

$$\begin{cases} m = n_1 k_1^{r_1} + \cdots + n_l k_l^{r_l} \\ C = QMQ^{-1} \end{cases} \tag{7}$$

The matrix M is shown in Equation (8).

$$\begin{cases} M = \begin{pmatrix} M' & R_1 & R_2 \\ O & R_3 & R_4 \\ O & O & R_5 \end{pmatrix} \\ M' = \begin{pmatrix} m_1 & m_2 \\ m_3 & m_4 \end{pmatrix} \end{cases} \tag{8}$$

In the matrix M, the following conditions are met: $m = m_1 + m_2 + m_3 + m_4$. During the decryption process, the encoded matrix M of m is recovered using the private key, and the calculation formula is shown in Equation (9).

$$M = Q^{-1} C Q \tag{9}$$

Finally, the plaintext m is obtained through calculation $m = m_1 + m_2 + m_3 + m_4$. The security of the proposed algorithm can be reduced to the unsolvability of the CSP and DLP problems. At the chosen 128-bit security strength, the algorithm's security satisfies the following reduction relation:

(1) Ciphertext Indistinguishability: Assuming a probabilistic multinomial-time adversary A can distinguish the ciphertexts corresponding to two plaintexts of equal length with a non-negligible advantage, a simulator B can be constructed to solve the DLP using A's distinguishing ability. B receives DLP challenge instances and embeds them into the randomization component of the ciphertext. If A can correctly distinguish

the ciphertext, B can output the DLP solution using A's distinguishing result. This means that if an adversary A exists that can break the semantic security of this algorithm, then an algorithm B exists to solve the DLP, contradicting the difficulty assumption of DLP. Therefore, this algorithm satisfies the ciphertext indistinguishability under chosen-plaintext attacks.

(2) Private Key Irrecoverability: Given two elements in a non-commutative group, there exists a secret element between them such that one element can be obtained by performing a conjugate operation on the other element using this secret element. The process of recovering the private key from the public key is equivalent to solving for the secret element in the aforementioned conjugate equation, i.e., solving for a CSP instance. If a polynomial-time algorithm exists that can extract the private key from the public key with a non-negligible probability, it implies that a polynomial-time algorithm exists for solving CSP on the corresponding non-commutative group, which contradicts the hardness assumption of CSP.

In summary, under the security assumption that both CSP and DLP are computationally infeasible, this algorithm can achieve semantically secure privacy protection under the selected security parameters and has provable resistance to classical cryptanalysis and potential quantum computing threats.

2.3 Data Deduplication Method Based on CSP-DLP and FL

To address the redundant data generated during the operation of the V2X LBS information retrieval service system, a data deduplication method based on CSP-DLP and FL was developed. The core of privacy protection in FL lies in controlling the risk of information leakage during gradient transmission [21, 22]. Under the horizontal FL framework, the relationship between the local model update volume of the client and the privacy protection strength can be expressed by formula (10).

$$\Delta w_k^{priv} = \Delta w_k + \mathrm{Lap}\left(\frac{\Delta f}{\varepsilon}\right) \tag{10}$$

In Equation (10), Δw_k^{priv} represents the gradient update amount after adding privacy protection; Δw_k represents the original gradient update amount; Lap(.) represents the Laplace noise mechanism; Δf represents the gradient sensitivity; and ε represents the privacy budget parameter.

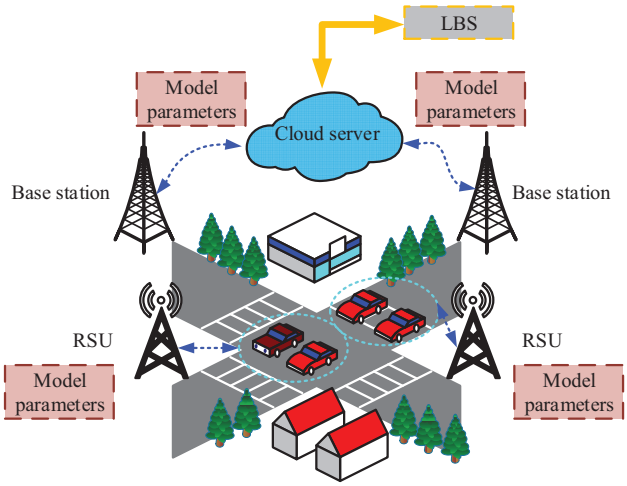

Figure 4 Operation flow of the internet of vehicles LBS information retrieval service system.

The operation flow of the V2X LBS information retrieval service system after incorporating the FL framework and the CSP-DLP asymmetric HE algorithm is shown in Figure 4.

As shown in Figure 4, vehicle nodes collect geographic location information in real time within their driving range and establish a secure communication connection with the RSU through a pseudonym registered with the CA center. After verifying the legitimacy of the vehicle's identity, the RSU uses an identity-based signature mechanism to confirm the identity and encrypts the received geographic location information using the CSP-DLP asymmetric HE algorithm. In the encrypted state, the RSU participates in the training of the local model, ensuring data privacy is not leaked. The cloud server performs aggregation calculations without accessing the original plaintext data, generates a new local model, and gradually feeds this model back to various RSUs. After receiving the model update, the RSU feeds the updated model back to the corresponding vehicle node, completing a full FL training and update cycle. In the CSP-DLP asymmetric encryption algorithm, identical data is hashed into the private key required by the algorithm. Since identical data corresponds to the same private key, the RSU can perform encrypted data comparison and deduplication without decrypting the data. The data deduplication process is shown in Figure 5.

As shown in Figure 5, during the initialization phase, the RSU runs G_{csp}, G_{dlp}, and g according to K to generate a public-private key pair and

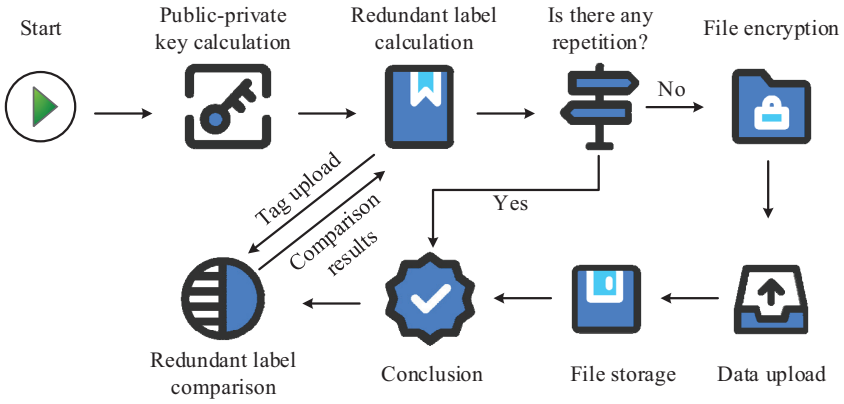

Figure 5 Data deduplication process.

sends the public key to the vehicle. The vehicle runs the encryption module to obtain the ciphertext of the key and sends it back to the RSU. The RSU obtains the vehicle's key by running the decryption module. During the vehicle data upload phase, the raw data collected by the vehicle is first processed by a one-way hash function to obtain the private key. Then, the corresponding public key is derived from the private key. Next, the vehicle uses the public key to perform HE on the raw data to generate ciphertext. Finally, the vehicle uses the ciphertext of the key to send the encrypted data along with its corresponding public key to the RSU. After receiving the data, the RSU first decrypts the symmetric key to obtain the ciphertext of the data and the corresponding public key. The RSU makes a judgment by comparing whether the public key of the current data already exists in the index list based on the data public key. If the public key already exists, it is determined to be duplicate data, and the ciphertext is directly discarded without storage or further processing. If the public key does not exist, it is determined to be new data, the public key is added to the index list, and the ciphertext of the data is stored in the local dataset for subsequent local model training in FL. The entire process is completed in encrypted form. RSU never touches the original plaintext data, achieving efficient data deduplication while protecting privacy.

Regarding the question of whether deterministic public-key comparison mechanisms might leak data distribution characteristics and trigger side-channel attacks, this paper addresses this from three aspects. First, the granularity of the leaked information is limited: The RSU can only determine whether two ciphertexts correspond to the same plaintext through

public-key indexing but cannot determine the specific value or meaning of the plaintext. In V2X LBS scenarios, after vehicle location, speed, and other data are homomorphically encrypted, the RSU cannot deduce the original coordinates or trajectory information from the public key. The RSU can only know "which data are duplicates," not the plaintext content itself. Second, the RSU's operating environment is controllable: RSUs in V2X systems are typically deployed in fixed locations by service providers or traffic management departments. Their behavior is constrained by physical protection, software integrity verification, and regular audits. Compared to a completely untrusted public cloud environment, the trust boundary of the RSU is more clearly defined. The conditions required for attackers to conduct side-channel analysis (such as continuous observation and precise timing measurements) are difficult to meet in actual deployments. Third, engineering hardening measures can be deployed. Even considering side-channel analysis under extreme conditions, the proposed scheme can still be further hardened through engineering means. For example, RSU can batch process deduplication requests and introduce random delays to obscure the comparison timing of individual data or adopt a periodic index update strategy for frequently repeated data to break long-term statistical correlations. These enhancements can be deployed without changing the core algorithm logic and can be flexibly configured as system parameters.

In summary, the deterministic public key comparison mechanism is reasonable and effective under the threat model set in this paper. RSU can obtain the repeated distribution characteristics of data, but this information itself does not constitute a substantial infringement on user privacy, nor does it violate the privacy protection goals of V2X LBS services.

3 Results

3.1 Algorithm Correctness Analysis

The study first verified the correctness of the homomorphic property of the asymmetric HE algorithm based on CSP-DLP through three operation methods: homomorphic addition, homomorphic subtraction, and homomorphic multiplication.

(1) Proof of homomorphic addition: Suppose there are two plaintexts m_1 and m_2, their corresponding ciphertexts are C_1 and C_2, respectively.

$$C_{add} = C_1 + C_2 = QM_1Q^{-1} + QM_2Q^{-1}$$

$$
\begin{aligned}
&= Q \begin{pmatrix} M_1' & R_{1,1} & R_{2,1} \\ O & R_{3,1} & R_{4,1} \\ O & O & R_{5,1} \end{pmatrix} Q^{-1} + Q \begin{pmatrix} M_2' & R_{1,2} & R_{2,2} \\ O & R_{3,2} & R_{4,2} \\ O & O & R_{5,2} \end{pmatrix} Q^{-1} \\
&= Q \begin{pmatrix} M_1' + M_2' & R_1^* & R_2^* \\ O & R_3^* & R_4^* \\ O & O & R_5^* \end{pmatrix} Q^{-1} \\
&= QM_{add}Q^{-1}
\end{aligned}
$$

C_{add} deciphering:

$$
M_{add} = Q^{-1}C_{add}G = \begin{pmatrix} M_1' + M_2' & R_1^* & R_2^* \\ O & R_3^* & R_4^* \\ O & O & R_5^* \end{pmatrix}
$$

$$
M_1' + M_2' = \begin{pmatrix} m_{1,1} + m_{1,2} & m_{2,1} + m_{2,2} \\ m_{3,1} + m_{3,2} & m_{4,1} + m_{4,2} \end{pmatrix}
$$

Plaintext m_{add} calculation:

$$
\begin{aligned}
m_{add} &= m_{1,1} + m_{1,2} + m_{2,1} + m_{2,2} + m_{3,1} + m_{3,2} + m_{4,1} + m_{4,2} \\
&= (m_{1,1} + m_{1,2} + m_{3,1} + m_{4,1}) + (m_{1,2} + m_{2,2} + m_{3,2} + m_{4,2}) \\
&= m_1 + m_2
\end{aligned}
$$

The result satisfies the properties of homomorphic addition, and the proof is valid.

(2) Proof of homomorphic subtraction:

$$
\begin{aligned}
C_{sub} = C_1 - C_2 &= QM_1Q^{-1} - QM_2Q^{-1} \\
&= Q \begin{pmatrix} M_1' & R_{1,1} & R_{2,1} \\ O & R_{3,1} & R_{4,1} \\ O & O & R_{5,1} \end{pmatrix} Q^{-1} \\
&\quad - Q \begin{pmatrix} M_2' & R_{1,2} & R_{2,2} \\ O & R_{3,2} & R_{4,2} \\ O & O & R_{5,2} \end{pmatrix} Q^{-1} \\
&= Q \begin{pmatrix} M_1' - M_2' & R_1^* & R_2^* \\ O & R_3^* & R_4^* \\ O & O & R_5^* \end{pmatrix} Q^{-1} \\
&= QM_{sub}Q^{-1}
\end{aligned}
$$

C_{sub} deciphering:

$$M_{sub} = Q^{-1}C_{sub}G = \begin{pmatrix} M'_1 - M'_2 & R^*_1 & R^*_2 \\ O & R^*_3 & R^*_4 \\ O & O & R^*_5 \end{pmatrix}$$

$$M'_1 - M'_2 = \begin{pmatrix} m_{1,1} - m_{1,2} & m_{2,1} - m_{2,2} \\ m_{3,1} - m_{3,2} & m_{4,1} - m_{4,2} \end{pmatrix}$$

Plaintext m_{sub} calculation:

$$\begin{aligned} m_{sub} &= m_{1,1} - m_{1,2} - m_{2,1} - m_{2,2} - m_{3,1} - m_{3,2} - m_{4,1} - m_{4,2} \\ &= (m_{1,1} + m_{1,2} + m_{3,1} + m_{4,1}) - (m_{1,2} + m_{2,2} + m_{3,2} + m_{4,2}) \\ &= m_1 - m_2 \end{aligned}$$

The result satisfies the property of homomorphic subtraction, and the proof is valid.

(3) Proof of homomorphic multiplication:

$$\begin{aligned} C_{mul} &= C_1 \times C_2 + C_1 \times T \times C_2 \\ &= QM_1M_2Q^{-1} + QM_1T'M_2Q^{-1} \\ &= Q \begin{pmatrix} M'_1 & R_{1,1} & R_{2,1} \\ O & R_{3,1} & R_{4,1} \\ O & O & R_{5,1} \end{pmatrix} \begin{pmatrix} M'_2 & R_{1,2} & R_{2,2} \\ O & R_{3,2} & R_{4,2} \\ O & O & R_{5,2} \end{pmatrix} Q^{-1} \\ &\quad + Q \begin{pmatrix} M'_1 & R_{1,1} & R_{2,1} \\ O & R_{3,1} & R_{4,1} \\ O & O & R_{5,1} \end{pmatrix} \begin{pmatrix} t & R_{1,T} & R_{2,T} \\ O & t & R_{3,T} \\ O & O & t \end{pmatrix} \\ &\quad \times \begin{pmatrix} M'_2 & R_{1,2} & R_{2,2} \\ O & R_{3,2} & R_{4,2} \\ O & O & R_{5,2} \end{pmatrix} Q^{-1} \\ &= Q \begin{pmatrix} M'_1 \times M'_2 & R^*_1 & R^*_2 \\ O & R^*_3 & R^*_4 \\ O & O & R^*_5 \end{pmatrix} Q^{-1} \\ &\quad + Q \begin{pmatrix} M'_1 t M'_2 & R^*_1 & R^*_2 \\ O & tR^*_3 & R^*_4 \\ O & O & tR^*_5 \end{pmatrix} Q^{-1} \end{aligned}$$

$$= Q \begin{pmatrix} M_1' \times M_2' + M_1' t M' & R_1^* & R_2^* \\ O & tR_3^* & R_4^* \\ O & O & tR_5^* \end{pmatrix} Q^{-1}$$

$$= Q M_{mul} Q^{-1}$$

C_{mul} deciphering:

$$M_{mul} = Q^{-1} C_{mul} G = \begin{pmatrix} M_1' \times M_2' + M_1' t M_2' & R_1^* & R_2^* \\ O & R_3^* & R_4^* \\ O & O & R_5^* \end{pmatrix}$$

$$M_1' \times M_2' = \begin{pmatrix} m_{1,1} & m_{2,1} \\ m_{3,1} & m_{4,1} \end{pmatrix} \begin{pmatrix} m_{1,2} & m_{2,2} \\ m_{3,2} & m_{4,2} \end{pmatrix}$$

$$M_1' t M_2' = \begin{pmatrix} m_{1,1} & m_{2,1} \\ m_{3,1} & m_{4,1} \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} m_{1,2} & m_{2,2} \\ m_{3,2} & m_{4,2} \end{pmatrix}$$

Plaintext m_{mul} calculation:

$$m_{mul} = (m_{1,1} + m_{1,2} + m_{3,1} + m_{4,1}) \times (m_{1,2} + m_{2,2} + m_{3,2} + m_{4,2})$$
$$= m_1 \times m_2$$

The result satisfies the property of homomorphic multiplication, and the proof is valid.

3.2 Performance Testing of Asymmetric HE Algorithm

The study then simulated the proposed algorithm using simulation. The experimental data came from both real trajectory data and simulation data. The real data were from the T-Drive Taxi Trajectory Dataset, which includes fields such as vehicle ID, latitude and longitude coordinates, timestamp, and driving speed. The simulation data were generated using the SUMO traffic simulation platform to supplement scenarios involving high-density road sections and high-concurrency uploads. The study preprocessed the collected data. First, the raw T-Drive trajectory data was cleaned to remove invalid records with missing latitude, longitude, or timestamps, eliminating approximately 1.2% of outlier data. Second, continuous trajectory points were resampled at equal intervals, with the sampling frequency unified to 1 Hz to ensure alignment in the time dimension. Subsequently, the simulated position coordinates generated by SUMO and the actual T-Drive trajectory

Table 1 Hardware and software environment settings

Environment	Settings
Operating system	Ubuntu 20.04 LTS (64-bit)
CPU	Intel Core i7-10700 @ 2.90 GHz
Number of CPU cores	8 cores/16 threads
Memory (RAM)	32 GB DDR4
Storage	1 TB SSD
GPU	NVIDIA GeForce RTX 3080 (10 GB)
Programming language	Python 3.8
Cryptographic library	PyCryptodome 3.15
FL framework	TensorFlow Federated 0.20
Data processing framework	NumPy 1.21, Pandas 1.4
Network simulation	Mininet 2.3
Security level	128-bit security strength

were mapped together onto a 500 m $\times$ 500 m spatial grid and associated with the corresponding RSU coverage area. Finally, a timestamp and vehicle pseudonym were added to each position record to form structured input data. The format of each record was (vehicle_id, timestamp, latitude, longitude, speed, rsu_id). The final dataset contained approximately 1 million location records, each ranging from approximately 64 to 128 bytes in length. The environment settings are presented in Table 1.

Under the same hardware and software environment, the CSP-DLP asymmetric HE algorithm, the Paillier HE algorithm and Brakerski-Gentry-Vaikuntanathan (BGV) were used to encrypt and decrypt data, respectively. The average encryption time and average decryption time of a single data entry were statistically analyzed to evaluate the computational efficiency of the proposed algorithm in real-time vehicle networking application scenarios. The test outcomes are presented in Figure 6.

Figures 6(a) and 6(b) show the time consumption of different algorithms during encryption and decryption, respectively. As shown in Figure 6(a), while the lattice-based HE algorithm theoretically provided strong security, it involved a significant amount of mathematical computation. The Paillier algorithm relied on large integer modular exponentiation, resulting in high computational complexity. This proposed algorithm transformed the encryption operation into matrix operations and conjugate operations on a non-commutative group, avoiding the costly modular exponentiation calculations and significantly reducing encryption latency. Furthermore, the proposed algorithm eliminated the need to generate large random numbers

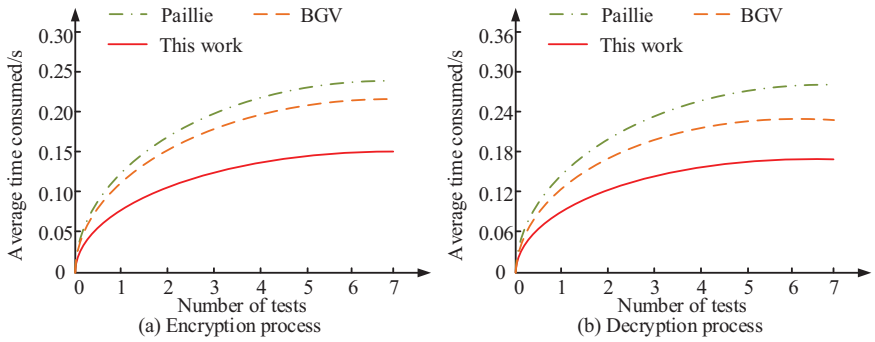

Figure 6 Encryption and decryption overhead test results.

and perform modular exponentiation during encryption; instead, it achieved plaintext obfuscation through lightweight matrix encoding, thus significantly reducing encryption latency while maintaining the same security level. For example, the average encryption time of the CSP-DLP asymmetric HE algorithm was 0.15 s; while the Paillier HE algorithm and BGV algorithms achieved 0.24 s and 0.22 s, respectively. The encryption overhead of the studied algorithm was 62.5% of that of the Paillier algorithm. As shown in Figure 6(b), compared to other encryption algorithms, the CSP-DLP asymmetric HE algorithm still exhibited superior decryption performance. The average decryption times for the CSP-DLP asymmetric HE algorithm, the Paillier HE algorithm, and the BGV algorithms were 0.17 s, 0.28 s and 0.23 s, respectively. This improvement in decryption efficiency indicated that the proposed algorithm could provide better real-time response performance when processing large-scale encrypted data.

To further quantitatively analyze the space optimization capability of the proposed algorithm in high-density V2X data storage scenarios, this study employed the CSP-DLP asymmetric HE algorithm, the Paillier HE algorithm, and the BGV algorithms to encrypt each original location record in the dataset and calculated the average byte length of the generated ciphertext. Based on this, the ratio between the ciphertext size and the original plaintext data size was calculated to measure the storage efficiency under different encryption schemes. Experimental metrics included the size of a single ciphertext record (in bytes) and the storage expansion rate. The test outcomes are presented in Figure 7.

Figure 7(a) shows the ciphertext size generated by different encryption algorithms, and Figure 7(b) shows the storage expansion rate under different encryption algorithms. The storage expansion rate reflected the extent to

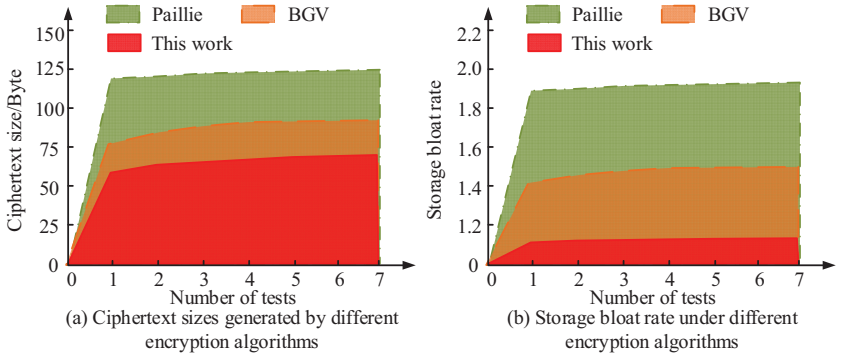

Figure 7 Ciphertext size and storage efficiency test results.

which the ciphertext size expanded relative to the original plaintext data size and was calculated as the ratio of the ciphertext size to the plaintext size. As shown in Figure 7(a), the average ciphertext size generated by the CSP-DLP asymmetric HE algorithm was significantly smaller than that of other algorithms, with an average byte length of approximately 72 bytes per ciphertext. The average ciphertext size of the Paillier HE algorithm was 125 bytes, while BGV algorithms was 95 bytes. It can be calculated that the ciphertext size of the studied algorithm was 0.576 times that of the Paillier algorithm, meaning the ciphertext size was only 57.6% of that of the Paillier algorithm, a reduction of approximately 42.4%. Therefore, the CSP-DLP asymmetric HE algorithm could protect data privacy with a smaller ciphertext size and effectively reduce the consumption of storage resources. As shown in Figure 7(b), the storage expansion rate of the CSP-DLP asymmetric HE algorithm was 1.125, significantly lower than that of the Paillier algorithm (1.875) and BGV algorithm (1.485). This indicated that, with the same amount of original data, the algorithm could significantly reduce the storage requirements after encryption, effectively alleviating storage pressure in high-density data scenarios. For systems such as the Internet of Vehicles that need to process massive amounts of data, a reduced storage expansion rate meant more efficient use of storage space and lower bandwidth consumption.

3.3 Simulation Testing of Data Deduplication Method

This study used the T-Drive Taxi Trajectory Dataset and the Porto Taxi Trajectory Dataset as datasets. Through spatial region mapping and temporal synchronization, it generated vehicle location and status data for multiple RSU coverage areas, totaling approximately 1 million records. It simulated a

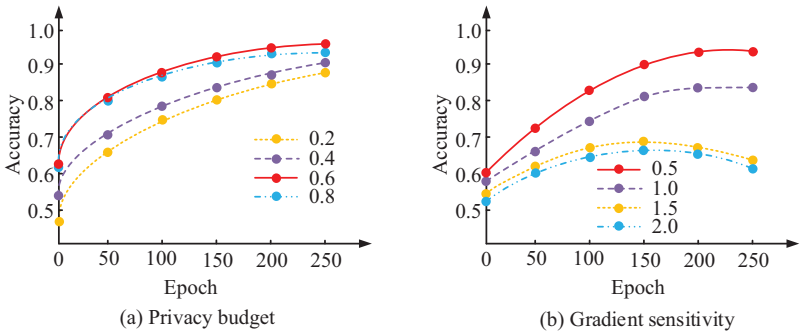

Figure 8 Hyperparameter selection test.

scenario where 50–200 vehicles continuously uploaded data in different road sections. The proportion of duplicate data was controlled at 50% to simulate the large amount of redundant privacy data generated by the V2X system during continuous operation. The study first tested the privacy budget ε and gradient sensitivity Δf of the FL framework using hyperparameter selection. The test results are shown in Figure 8.

Figure 8(a) indicates the test results for the privacy budget ε selection, and Figure 8(b) indicates the test results for the gradient sensitivity Δf selection. As shown in Figure 8(a), the overall model accuracy generally increased as the privacy budget increased from 0.2 to 0.8. When the privacy budget ε was 0.6, the model accuracy increased rapidly and stabilized at 0.94 after about 250 iterations. At this point, the model reached its optimal state, indicating that a moderate privacy setting could significantly improve model performance while ensuring privacy. When the privacy budget was further increased to 0.8, the model accuracy improved slightly in the early stages of training, but the final accuracy was slightly lower than when the privacy budget was 0.6. As shown in Figure 8(b), a lower gradient sensitivity helped stabilize model training. The model accuracy was highest when Δf was 0.5, eventually stabilizing at 0.92. However, as the gradient sensitivity gradually increased, the model's accuracy dropped in multiple iterations, indicating that excessive sensitivity amplified the interference of noise injection on the gradient direction, rendering it challenging for the model to converge accurately. In conclusion, the study ultimately selected $\varepsilon = 0.6$ and $\Delta f = 0.5$ as the default configuration combination for the data deduplication method.

To comprehensively evaluate the impact of the proposed data deduplication method on the training process of the FL model, this study further compared and analyzed the model convergence behavior and final

Table 2 Comparison of FL model training performance under different configurations

Configuration	Convergence Rounds (Epochs to Reach 94% Accuracy)	Final Model Accuracy (%)
No deduplication (Baseline)	205	95.2
Deduplication without encryption (Plaintext deduplication)	210	94.8
Deduplication with encryption (Proposed method)	218	94.6

performance under three different configurations: Configuration 1 (no deduplication), Configuration 2 (deduplication but no encryption, i.e., plaintext deduplication), and Configuration 3 (deduplication and encryption, i.e., the proposed data deduplication method based on CSP-DLP and FL). The experiments recorded the number of convergence epochs (measured by the number of global iterations required to achieve 94% validation accuracy) and the final model accuracy (the highest stable accuracy after complete convergence) for each configuration. The results are shown in Table 2.

As shown in Table 2, compared to the baseline environment without deduplication, introducing a data deduplication mechanism (whether plaintext deduplication or in-paper encrypted deduplication) slightly increased the number of convergence rounds (by 5 and 13 rounds, respectively), while the final model accuracy decreased slightly (by 0.4% and 0.6%, respectively). This was because deduplication reduced the total amount of data used in training, especially removing a large number of duplicate samples that might enhance the model's ability to fit common patterns, thus having a limited impact on convergence speed and final accuracy. Nevertheless, the encrypted deduplication method proposed in this paper only required eight more convergence rounds than the plaintext deduplication method, and the final accuracy was only 0.2% lower, indicating that the encryption operation itself did not have a significant additional negative impact on model training. In summary, the proposed method maintained a model training effect that is highly close to the baseline environment while achieving privacy protection and storage optimization, verifying its feasibility and practicality within the FL framework.

To evaluate the practical capability of the introduced deduplication method in reducing redundant data transmission, the system was run in three modes: without deduplication enabled, with deduplication based on CSP-DLP and FL enabled, and with deduplication based on BGV and FL enabled. Data transmission time from vehicle nodes to RSUs and from RSUs to the

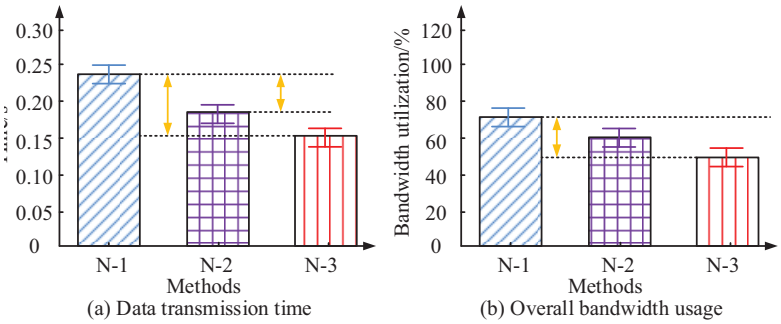

Figure 9 Data transmission overhead and deduplication efficiency test results.

cloud server, as well as overall bandwidth usage, were statistically analyzed. The test outcomes are presented in Figure 9.

Figure 9(a) compares the time it took for vehicle nodes to transmit data to the RSU, and Figure 9(b) shows the bandwidth usage of the RSU transmitting data to the cloud server. As shown in Figure 9(a), without data deduplication, each vehicle node needed to upload approximately 64-128 bytes of original location data, with duplicate data accounting for 50%, resulting in a large average data upload volume per vehicle. Under these conditions, the average time for vehicle nodes to transmit data to the RSU was 0.24 seconds. After enabling the deduplication method based on BGV and FL, the average time for vehicle nodes to transmit data to the RSU was 0.19 seconds. However, after enabling the data deduplication method based on CSP-DLP and FL, the average transmission time decreased to 0.16 seconds, a reduction of approximately 33.3%. As shown in Figure 9(b), without data deduplication, the bandwidth usage of the RSU transmitting data to the cloud server was quite severe, with an average bandwidth utilization rate reaching 70%. After enabling the data deduplication method based on CSP-DLP and FL, the bandwidth utilization rate significantly decreased to 50%, a reduction of approximately 28.6%, due to the reduction in redundant data. The test results of the duplication detection rate, storage requirements, and cross-node communication overhead of different deduplication methods are shown in Table 3.

As shown in Table 3, even with the proportion of duplicate data increasing to 75%, the proposed method maintained a high duplication detection rate of 88.3%. The rate of reduction in storage requirements decreased as the proportion of duplicate data increased. The denominator of the rate of reduction in storage requirements was the total amount of original data, while the

Table 3 Multi-indicator performance test results

Methods	Duplicate Data/%	Repeat Detection Rate/%	Storage Requirements/ Reduction%	Cross-Node Communication Overhead/Reduction%
This work	0	97.4	38.6	29.4
	25	94.5	35.0	25.1
	50	92.0	30.5	21.7
	75	88.3	25.0	18.3
BGV + FL	0	92.0	27.1	19.3
	25	89.4	23.4	17.5
	50	85.1	18.7	14.6
	75	79.8	14.2	12.2

numerator was the storage space saved by deduplication. When the proportion of duplicate data increased, although the absolute amount of duplicates increased, the total amount of original data also increased. Furthermore, data deduplication methods primarily eliminated completely identical redundant data. However, duplicate data in connected vehicle scenarios often exhibited highly similar but not identical characteristics (e.g., slight differences in the coordinates of trajectory points uploaded by adjacent vehicles on the same road segment). This portion of data cannot be identified as duplicates by deterministic public key comparison mechanisms. Therefore, when the proportion of duplicate data increased from 0 to 75%, a significant portion of the newly added duplicate data was "approximately duplicated" rather than "completely duplicated," leading to diminishing marginal returns on deduplication efficiency, manifested as a slower increase in the rate of reduction in storage requirements. The BGV scheme, due to its higher ciphertext expansion rate (storage expansion rate of 1.485), showed a more significant decrease in its rate of reduction in storage requirements as the proportion of duplicate data increased. The proposed method reduced storage requirements by an average of 38.6% and cross-node communication overhead by approximately 29.4% while maintaining dense processing throughout the process. The proposed method effectively reduced storage requirements and communication overhead.

4 Summary

The core contributions of this paper can be summarized in three aspects. Firstly, in terms of the research problem, it resolves the fundamental

contradiction between ciphertext indistinguishability and comparability from a cryptographic perspective. Secondly, in terms of methodology, it proposes an asymmetric HE algorithm based on CSP-DLP. This algorithm replaces traditional modular exponentiation with matrix encoding and conjugation operations, achieving semantic security at a 128-bit security strength. Thirdly, in terms of innovative results, it deeply integrates the CSP-DLP-based asymmetric HE algorithm with a FL framework, designing a deterministic deduplication mechanism that supports ciphertext comparison. This method effectively alleviates the computational and storage pressure in high-density scenarios while ensuring data privacy and system security. Simulation results showed that, in terms of encryption and decryption efficiency, the average encryption time of the CSP-DLP asymmetric HE algorithm was 0.15 s, and the average decryption time was 0.17 s. In terms of storage and communication overhead, after enabling the data deduplication method based on the CSP-DLP asymmetric HE algorithm and FL, storage requirements were reduced by 38.6%, and cross-node communication overhead was reduced by approximately 29.4%. Overall testing showed that the proposed method not only improved the data transmission efficiency of the V2X system but also optimized the use of storage and bandwidth resources.

While fully acknowledging the aforementioned achievements, the study also systematically reflected on its limitations. Regarding potential biases in the experimental results, the experimental data came from the T-Drive Taxi Trajectory Dataset and the SUMO simulation platform. The proportion of duplicate data was artificially controlled between 50% and 75% in the experiments. This setting might introduce some bias in the extrapolation of experimental results in low-repetition-rate scenarios. However, the study focused on high-density, high-redundancy environments, and a 50–75% repetition rate was a typical characteristic of such scenarios. Therefore, this setting did not affect the validity of the core conclusions of this paper. In terms of security assumptions, this paper adopted the RSU controllability assumption and the node honesty assumption, which are mainstream security models in the field of V2X privacy protection. If stronger attackers exist in future deployment environments, additional protection layers can be added to the proposed method. Furthermore, the scalability and real-time processing capabilities of the system in extremely large-scale, high-density environments require further verification. Future research will focus on optimizing the algorithm's performance on large datasets, exploring its adaptability in different V2X architectures, and evaluating its application in high-density scenarios such as the Industrial IoT and smart cities.

References

[1] Guo Z, Liu Q, Gao Z. Modular-Based Compression Scheme for Address Data in the Blockchain System for IoV Applications. IEEE Transactions on Vehicular Technology, 2024, 73(10): 15567–15583. DOI: 10.1109/TVT.2024.3411568.

[2] Yakhni S, Tekli J, Mansour E. Using fuzzy reasoning to improve redundancy elimination for data deduplication in connected environments. Soft Computing, 2023, 27(17): 12387–12418. DOI: 10.1007/s00500-023-07880-z.

[3] Luo R, Jin H, He Q. Enabling balanced data deduplication in mobile edge computing. IEEE Transactions on Parallel and Distributed Systems, 2023, 34(5): 1420–1431. DOI: 10.1109/TPDS.2023.3247061.

[4] Tang X, Zhu Y, Fu M. Comments on "Privacy Aware Data Deduplication for Side Channel in Cloud Storage". IEEE Transactions on Cloud Computing, 2024, 12(2): 814–817. DOI: 10.1109/TCC.2024.3376996.

[5] Zhou C, Ansari N. Securing federated learning enabled NWDAF architecture with partial homomorphic encryption. IEEE Networking Letters, 2023, 5(4): 299–303. DOI: 10.1109/LNET.2023.3294497.

[6] Xie Q Jiang S, Jiang L. Efficiency optimization techniques in privacy-preserving federated learning with homomorphic encryption: A brief survey. IEEE Internet of Things Journal, 2024, 11(14): 24569–24580. DOI: 10.1109/JIOT.2024.3382875.

[7] Cai Y, Ding W, Xiao Y. Secfed: A secure and efficient federated learning based on multi-key homomorphic encryption. IEEE Transactions on Dependable and Secure Computing, 2023, 21(4): 3817–3833. DOI: 10.1109/TDSC.2023.3336977.

[8] Mantey E A, Zhou C, Anajemba J H. Federated learning approach for secured medical recommendation in internet of medical things using homomorphic encryption. IEEE Journal of Biomedical and Health Informatics, 2024, 28(6): 3329–3340. DOI: 10.1109/JBHI.2024.3350232.

[9] Shah R, Mukherjee K, Tyagi A. R2D2: reducing redundancy and duplication in data lakes. Proceedings of the ACM on Management of Data, 2023, 1(4): 1–25. DOI: 10.1145/3626762.

[10] Hijazi N M, Aloqaily M, Guizani M. Secure federated learning with fully homomorphic encryption for IoT communications. IEEE Internet of Things Journal, 2023, 11(3): 4289–4300. DOI: 10.1109/JIOT.2023.3302065.

[11] Lejun Z, Minghui P, Shen S. Redundant data detection and deletion to meet privacy protection requirements in blockchain-based edge computing environment. China Communications, 2024, 21(3): 149–159. DOI: 10.23919/JCC.fa.2021-0815.202403.

[12] Rani R, Kumar N, Khurana M. Redundancy elimination in IoT oriented big data: A survey, schemes, open challenges and future applications. Cluster Computing, 2024, 27(1): 1063–1087. DOI: 10.1007/s10586-023-04209-1.

[13] Song M, Hua Z, Zheng Y. FCDedup: A two-level deduplication system for encrypted data in fog computing. IEEE Transactions on Parallel and Distributed Systems, 2023, 34(10): 2642–2656. DOI: 10.1109/TPDS.2023.3298684.

[14] Zhao J, Yang Z, Li J. Encrypted data reduction: Removing redundancy from encrypted data in outsourced storage. ACM Transactions on Storage, 2024, 20(4): 1–30. DOI: 10.1145/3685278.

[15] Qi S, Wei W, Wang J. Secure data deduplication with dynamic access control for mobile cloud storage. IEEE Transactions on Mobile Computing, 2023, 23(4): 2566–2582. DOI: 10.1109/TMC.2023.3263901.

[16] Jatoth C, Doriya R. IoV block secure: Blockchain based secure data collection and validation framework for internet of vehicles network. Peer-to-Peer Networking and Applications, 2024, 17(6): 3964–3990. DOI: 10.1007/s12083-024-01802-y.

[17] Yang H, Xue D, Ge M. Fast generation-based gradient leakage attacks: An approach to generate training data directly from the gradient. IEEE Transactions on Dependable and Secure Computing, 2024, 22(1): 132–145. DOI: 10.1109/TDSC.2024.3387570.

[18] Xu Y, Mao Y, Li S. Privacy-preserving federal learning chain for internet of things. IEEE Internet of Things Journal, 2023, 10(20): 18364–18374. DOI: 10.1109/JIOT.2023.3279830.

[19] Guo L, Gao W, Cao Y. Research on medical data security sharing scheme based on homomorphic encryption. Math. Biosci. Eng., 2023, 20(2): 2261–2279. DOI: 10.3934/mbe.2023106.

[20] Song M, Hua Z, Zheng Y. Enabling transparent deduplication and auditing for encrypted data in cloud. IEEE Transactions on Dependable and Secure Computing, 2023, 21(4): 3545–3561. DOI: 10.1109/TDSC.2023.3334475.

[21] Mi B, Zhou J, Huang D. Privacy-preserving data processing method for IoV based on homomorphic conjugacy search problem. IEEE Transactions on Intelligent Transportation Systems, 2024, 25(7): 7374–7387. DOI: 10.1109/TITS.2024.3351837.

[22] Xue J, Yu K, Zhang T. Cooperative deep reinforcement learning enabled power allocation for packet duplication URLLC in multi-connectivity vehicular networks. IEEE Transactions on Mobile Computing, 2024, 23(8): 8143–8157. DOI: 10.1109/TMC.2023.3347580.

Biographies

Qiongfeng Mo (August 1981–), male, graduated from Guangdong University of Technology, China, with a master's degree in Electrical Engineering and Automation. After graduation, he worked as a senior economist at Qingyuan Power Supply Bureau of Guangdong Power Grid Co. Ltd. His current research direction is warehousing and logistics.

Wei Liao (July 1979–), male, graduated from Guangdong University of Technology, China, with a major in Control Engineering and obtained a

master's degree. After graduation, he worked as a mid-level engineer at Qingyuan Power Supply Bureau of Guangdong Power Grid Co. Ltd. His current research direction is intelligent warehousing work.

Hengjian Liao (January 1988–), male, graduated from South China Agricultural University, China, with a major in Electrical Engineering and Automation, and obtained a bachelor's degree. After graduation, he worked as a senior engineer at Qingyuan Power Supply Bureau of Guangdong Power Grid Co. Ltd. His current research direction is warehousing and logistics.

Jiarong Guo (January 1997–), male, graduated from Hong Kong Baptist University, China, with a master's degree. After graduation, he worked as a mid-level economist at Qingyuan Power Supply Bureau of Guangdong Power Grid Co. Ltd. His current research direction is warehousing and logistics.

Xiaodong Feng (December 1990–), male, graduated from the University of Electronic Science and Technology of China, China, with a major in Electrical Engineering and Automation, and obtained a bachelor's degree. After graduation, he worked as an engineer at the Qingyuan Power Supply Bureau of Guangdong Power Grid Co. Ltd. His current research direction is warehousing and logistics.

Ruyan Guo (December 1990–), male, graduated from Hunan University, China, with a major in Electrical Engineering and obtained a master's degree. After graduation, he worked as a junior operator at Qingyuan Power Supply Bureau of Guangdong Power Grid Co. Ltd. His current research direction is transformer fault classification based on machine learning.