Hindawi
Security and Communication Networks
Volume 2018, Article ID 6362010, 9 pages
https://doi.org/10.1155/2018/6362010

Research Article

Analysis on Matrix GSW-FHE and Optimizing Bootstrapping

Xiufeng Zhao,[1] Hefeng Mao,[1] Shuai Liu,[1] Weitao Song,[1] and Bo Zhang[2,3]

[1] *Department of Information Research and Security, Zhengzhou Information Science Technology Institute, Zhengzhou 450001, China*
[2] *School of Information Technology, Deakin University, Victoria 3125, Australia*
[3] *School of Information Science and Engineering, University of Jinan, Jinan 250022, China*

Correspondence should be addressed to Xiufeng Zhao; zhao_xiu_feng@163.com

Received 8 August 2018; Revised 29 October 2018; Accepted 3 December 2018; Published 19 December 2018

Guest Editor: Zhaoqing Pan

With the rapid development of multimedia technologies, the multimedia data storage and outsource computation are delegated to the untrusted cloud, which has led to a series of challenging security and privacy threats. Fully homomorphic encryption can be used to protect the privacy of cloud data and solve the trust problem of third party. In this paper, we analyse circular security of matrix GSW-FHE scheme. We derive a sufficient condition of circular security for matrix GSW-FHE scheme. It allows us to choose a good secret key via "reject sample" technique and furthermore obtain circular secure matrix GSW-FHE scheme. We also give an extended version of matrix GSW-FHE by defining deterministic asymmetric encryption algorithm and propose hybrid homomorphic plaintext slot-wise switching method, which significantly reduces computation and storage complexity of bootstrapping key generation, thus optimizing the bootstrapping procedure.

1. Introduction

With the rapid development of multimedia technologies, for example, high-efficiency video coding (HEVC) is becoming popular due to its excellent coding performance [1]; the multimedia data storage and outsource computation are delegated to the untrusted cloud server, which has led to a series of challenging security and privacy threats. To tackle the security and privacy issues in cloud computing and storage, a lot of researches have been performed, such as fully homomorphic encryption [2, 3], attribute-based encryption, searchable encryption [4], and ciphertext retrieval scheme [5, 6]. The concept of homomorphic encryption is proposed by Rivest et al. [7], and Gentry [2, 3] proposed the first fully homomorphic encryption (FHE) scheme based on ideal lattice. FHE allows us to evaluate any function over ciphertext and obtain the function over corresponding plaintext by decryption. Fully homomorphic encryption can be used to protect the privacy of cloud data and solve the trust problem of untrusted third party. So the fully homomorphic encryption has a broad application prospect in the cloud computation and the big data field. There are many fully homomorphic encryption schemes based NP-hard problems, such as ideal lattice [2, 3], LWE [8, 9], RLWE [10], LWR [11], and so forth.

The difficulty of constructing fully homomorphic encryption scheme is reducing the noise in the ciphertext. The noise increases rapidly during ciphertext evaluations and eventually reaches a threshold beyond which we can no longer decrypt the resulting ciphertext correctly. Therefore, the somewhat homomorphic encryption scheme is constructed, which can homomorphically evaluates arithmetic circuits of limited depth. To get pure fully homomorphic encryption scheme, Gentry proposed bootstrapping technique. The bootstrapping technique is currently the only way to get pure fully homomorphic encryption from somewhat homomorphic encryption. Its main idea is refreshing ciphertext by homomorphic decryption and getting fresh ciphertext and realizing the purpose of reducing ciphertext noise. The critical process of bootstrapping technique is encrypting the pieces of secret key, and the corresponding ciphertexts are viewed as public evaluation key. Thus, the homomorphic encryption scheme must enjoy circular security.

Unfortunately, all known FHE schemes are supposed to be circular secure except [10, 12]. If fully homomorphic encryption scheme satisfies circular security, it is not necessary to generate as many public evaluation keys as the depth of evaluation circuit. But being circular secure is not a naive security attribute, so it is necessary to analyse circular security for concrete fully homomorphic encryption scheme. Meanwhile, bootstrapping is used to refresh ciphertext, and the procedure is implemented frequently to get pure fully homomorphic encryption. Therefore, how to improve the bootstrapping efficiency is worth intensive studying.

Our Results. We analyse circular security of matrix GSW-FHE scheme [13]. From formal definition of circular security, we derive a sufficient condition of circular security for matrix GSW-FHE scheme. That is, the matrix GSW-FHE scheme satisfies circular security with some function, if the equations about secret key have solution over $\mathbb{Z}_q$. Therefore, we can choose a good secret key via "reject sample" technique and furthermore obtain circular secure matrix GSW-FHE scheme.

We also give an extended version of matrix GSW-FHE by defining deterministic asymmetric encryption algorithm. To simplify the homomorphic equality test procedure, we propose hybrid homomorphic plaintext slot-wise switching method using symmetric encryption and deterministic public encryption algorithms, which significantly reduces computational cost of bootstrapping key generation, thus optimizing the bootstrapping procedure of work [13].

We may implement a trade-off between computation and storage complexity of bootstrapping. We delete part of the bootstrapping keys and compute them online when running Rounding procedure. In view of that, their computation involves only matrix additions; this cuts down the size of the large public bootstrapping key by a third, paying matrix additions with negligible computation complex.

Related Works. Encryption scheme achieves circular security, if it remains secure and even the secret key is encrypted under corresponding public key. In other words, circular secure encryption scheme resists key-dependent message (KDM) attack.

In the last few years, circular secure encryption schemes have been studied extensively [14–17]. Boneh et al. constructed a circular secure public key encryption scheme based on the DDH assumption without random oracle [16]. Based on Regev's LWE-based encryption scheme [18], Applebaum et al. constructed efficient cryptosystems enjoying circular secure [17]. Brakerski and Vaikuntanathan [10] proposed circular secure homomorphic encryption scheme based on the ring-LWE assumption. The main idea in the work of [10, 17] is generating a valid ciphertext that decrypts to a message related to secret key. Because the entries of secret key are not in the message space, they introduced "noise flooding technique" and "rerandom technique" to "fit" the entries into the message space.

Brakerski and Vaikuntanathan presented a fully homomorphic encryption scheme based on the LWE assumption using relinearization technique [8]. The relinearization process allows doing one multiplication without increasing the size of the ciphertext and obtaining an encryption of the product under a new secret key. Posting a "chain" of L secret keys allows performing up to L levels of multiplications without blowing up to the ciphertext size. Yang et al. consider that if the relinearization satisfies circular security, the "chain" of L secret keys may be back down to only one secret key, and they proposed a circular secure relinearization by defining a new assumption [12].

EuroCrypt 2013, Gentry, Sahai, and Waters proposed a new fully homomorphic encryption scheme based on the *approximate eigenvector* method, which is called GSW-FHE [19]. In the GSW-FHE scheme, homomorphic addition and multiplication are just matrix addition and multiplication. But GSW scheme operates one bit every running encryption algorithm. PKC 2015, Hiromasa et al. constructed a variant of GSW scheme called matrix GSW-FHE, which encrypts matrices and supports homomorphic matrix addition and multiplication. And they optimized the bootstrapping procedure of Alperin-Sheriff and Peikert [20] using the matrix GSW-FHE scheme [13]. To achieve homomorphic matrix operation, the pubic key of matrix GSW-FHE scheme includes the ciphertexts that encrypt partial information of the secret key, so the matrix GSW-FHE scheme resorts to circular security assumption, but formal circular security proof was not given, and it remains an open problem.

There are other works to optimize the bootstrapping procedure. Ducas et al. [21] proposed FHEW scheme, which accelerates bootstrapping via embedding the cyclic group $\mathbb{Z}_q$ into the group of roots of unity: $i \longrightarrow X^i$, where i is a primitive q-th root of unity. Wang and Tang [22] proposed an integer bootstrapping scheme by introducing new methods to evaluate integer polynomials with GSW-FHE, and they extended the method to packing by encrypting the integers diagonally in a matrix, as the matrix GSW-FHE proposed by Hiromasa et al. [13]. Similarly, their scheme resorts to circular security assumption.

On the other hand, packing technique is used to evaluate efficiently a large number of ciphertexts, and it allows us to apply single-instruction-multiple-data (SIMD) homomorphic operations to all encrypted data [23, 24]. The bootstrapping procedure [13, 20] is optimized by embedding $\mathbb{Z}_q$ into symmetric group S_q, the multiplication group of q × q permutation matrix, and homomorphic permuting SIMD ciphertexts. The mathematic preliminary of SIMD technique is Chinese Remainder Theorem (CRT). The plaintext space can be split into many small spaces via the CRT. If the plaintext modulus q is a composite that factors into distinct powers $\mathrm{q} = r_1 \ldots r_t$, then the ring R_q can be mapped via the CRT to direct product of ring R_{r_i}'s.

Organization. In Section 2, we describe some preliminaries on the formal definition of homomorphic encryption and circular security and the isomorphic from additive group $\mathbb{Z}_\mathrm{q}$ to a group of cyclic permutations. In Section 3, we review the matrix GSW-FHE scheme and define a new deterministic asymmetric encryption algorithm. We give the analysis on circular security of matrix GSW-FHE scheme in Section 4. In Section 5, we propose hybrid plaintext slot switching method and optimize the bootstrapping procedure. We give conclusions in Section 6.

2. Preliminaries

We denote the set of integers by $\mathbb{Z}$. Let G be some group and let P be some probability distribution, and then we use $a \overset{U}{\longleftarrow} \mathrm{G}$ to denote that a is chosen from G uniformly at random and use $b \overset{R}{\longleftarrow} \mathrm{P}$ to denote that b is chosen along P.

The vector is denoted by bold lowercase letter, for example, $\mathbf{x}$, and the i-th element of a vector $\mathbf{x}$ is denoted by x_i. The inner product between two vectors is denoted by $\langle \mathbf{x},\mathbf{y} \rangle$. Matrices are written by using bold capital letters, for example, $\boldsymbol{X}$, and the i-th column vector of a matrix is denoted by $\boldsymbol{x}_i$. The $n \times n$ identity matrix is denoted by $\boldsymbol{I}_n$.

2.1. Homomorphic Encryption. Let $\mathscr{M}$and $\mathscr{C}$ be the message and ciphertext space. A homomorphic encryption scheme consists of four algorithms $\{KeyGen, Enc, Dec, Eval\}$.

(i) $\boldsymbol{KeyGen}(1^{\lambda})$: input security parameter λ and output a public encryption key pk, a secret decryption key sk, and a public evaluation key evk.

(ii) $\boldsymbol{Enc}_{pk}(m)$: input public key pk and plaintext $m \in \mathscr{M}$ and output ciphertext $c \in \mathscr{C}$.

(iii) $\boldsymbol{Dec}_{sk}(c)$: input secret key sk and ciphertext c and output the message encrypted in the ciphertext c.

(iv) $\boldsymbol{Eval}_{evk}(f, c_1, c_2, \ldots, c_l)$: input the evaluation key evk, function f, and ciphertexts $c_1, c_2, \ldots, c_l$ and output a ciphertext $c_f \in \mathscr{C}$ that is obtained by applying the function $f : \mathscr{M}^l \longrightarrow \mathscr{M}$ to $c_1, c_2, \ldots, c_l$.

2.2. Embedding $\mathbb{Z}_q$ into Symmetric Group. According to Cayley's Theorem, the additive group $\mathbb{Z}_\mathrm{q}$ is isomorphic to a group of cyclic permutations G, where $x \in \mathbb{Z}_\mathrm{q}$ corresponds to a cyclic permutation that can be represented by an indicator vector with 1 in the $(x+1)$-th position. The permutation matrix can be obtained from the cyclic rotation of the indicator vector. The addition in $\mathbb{Z}_\mathrm{q}$ leads to the composition of the permutations; the rounding function $\lfloor x \rceil_2 : \mathbb{Z}_q \longrightarrow \{0, 1\}$ can be computed by summing the entries of the indicator vector corresponding to those in $\mathbb{Z}_\mathrm{q}$ that round 1.

By CRT, $\mathbb{Z}_q$ is isomorphic to the direct product $\mathbb{Z}_{r_1} \times \ldots \times \mathbb{Z}_{r_t}$, where $\mathrm{q} := \prod_{i=1}^{t} r_i$, and r_i are small and powers of distinct primes. Similarly, $\mathbb{Z}_q$ embeds into symmetric group $S = S_{r_1} \times S_{r_2} \times \ldots \times S_{r_t}$.

3. Matrix GSW-FHE

3.1. Review Matrix GSW-FHE Scheme. In this section, we review the matrix GSW-FHE scheme. Let λ be the security parameter. The matrix GSW-FHE scheme is parameterized by an integer lattice dimension n, an integer modulus q, and a distribution χ over $\mathbb{Z}$ which is assumed to be sub-Gaussian; all of the parameters depend on λ. Let $l := \lceil \log q \rceil$, $m := O((n+r) \log q)$, and $\mathrm{N} := (n+r) \cdot l$. Let r be the amount of bits to be encrypted, which defines the message space $\{0, 1\}^{r \times r}$. The ciphertext space is $\mathbb{Z}_q^{(n+r) \times N}$. The scheme uses the rounding function $\lfloor \cdot \rceil_2$ where, for any $x \in \mathbb{Z}_q$, $\lfloor x \rceil_2$ outputs 1 if x is close to $q/4$ and 0 otherwise. Recall that $\boldsymbol{g}^T = (1, 2, \ldots, 2^{l-1})$ and $\mathbf{G} = g^T \bigotimes I_{n+r}$.

(i) KeyGen$(1^{\lambda}, r)$: Sample a uniformly random matrix $\overset{U}{\longleftarrow} \mathbb{Z}_q^{n \times m}$, secret key matrix $\mathbf{S}' \overset{R}{\longleftarrow} \chi^{r \times n}$, and noise matrix $\mathbf{E} \overset{R}{\longleftarrow} \chi^{r \times m}$. Let $\mathbf{S} := [\boldsymbol{I}_r \mid\mid -\boldsymbol{S}']$ and $\mathbf{B} := \left(\begin{smallmatrix} \boldsymbol{S}'\boldsymbol{A}+\boldsymbol{E} \\ \boldsymbol{A} \end{smallmatrix}\right) \in \mathbb{Z}_q^{(n+r) \times m}$. Let $\boldsymbol{M}_{(i,j)} \in \{0, 1\}^{r \times r}(\mathrm{i}, \mathrm{j} = 1, 2, \ldots, \mathrm{r})$ be the matrix with 1 in the (i, j)−th position and 0 in the others. For all i, j = 1, 2, . . . , r, first sample $\boldsymbol{R}_{(i,j)} \overset{U}{\longleftarrow} \{0, 1\}^{m \times N}$, and set

$$\boldsymbol{P}_{(i,j)} := \boldsymbol{B}\boldsymbol{R}_{(i,j)} + \begin{pmatrix} \boldsymbol{M}_{(i,j)}\mathbf{S} \\ \mathbf{0} \end{pmatrix} \boldsymbol{G} \in \mathbb{Z}_q^{(n+r) \times N} \qquad (1)$$

Output public key pk $:= (\{\boldsymbol{P}_{(i,j)}\}_{i,j \in [r]}, \boldsymbol{B})$ and secret key sk $:= \mathbf{S}$.

(ii) SecEnc$_{sk}(\boldsymbol{M} \in \{0, 1\}^{r \times r})$: Sample random matrixes $\mathbf{A}' \overset{U}{\longleftarrow} \{0, 1\}^{n \times N}$ and $\mathbf{E}' \overset{R}{\longleftarrow} \chi^{r \times N}$, parse $\mathbf{S} = [\boldsymbol{I}_r \mid\mid -\boldsymbol{S}']$, and output the ciphertext

$$\mathrm{C} := \left[\begin{pmatrix} \boldsymbol{S}'\boldsymbol{A}' + \boldsymbol{E}' \\ \boldsymbol{A}' \end{pmatrix} + \begin{pmatrix} \mathbf{MS} \\ \mathbf{0} \end{pmatrix} \boldsymbol{G} \right]_q \in \mathbb{Z}_q^{(n+r) \times N}. \qquad (2)$$

(iii) PubEnc$_{sk}(pk, \boldsymbol{M} \in \{0, 1\}^{r \times r})$: Sample a random matrix $\mathbf{R} \overset{U}{\longleftarrow} \{0, 1\}^{m \times N}$, and output the ciphertext

$$\mathrm{C} := \boldsymbol{B}\mathbf{R} + \sum_{i,j \in [r] : M_{[i,j]} = 1} \boldsymbol{P}_{(i,j)} \in \mathbb{Z}_q^{(n+r) \times N}, \qquad (3)$$

where $\boldsymbol{M}_{[i,j]}$ is the (i, j) − th element of $\boldsymbol{M}$.

(iv) $Dec_{sk}(sk, C)$: Output the matrix $\boldsymbol{M} = (\lfloor \langle \boldsymbol{s}_i, c_{jl-1} \rangle \rceil_2)_{i,j \in [r]}$, where $\boldsymbol{s}_i^T$ is the i^{th} row of $\mathbf{S}$.

3.2. Deterministic Asymmetric Encryption. We define a new deterministic asymmetric encryption algorithm in the matrix GSW-FHE scheme as follows:

(i) DetePubEnc$_{pk}(\boldsymbol{M} \in \{0, 1\}^{r \times r})$: input pk and $\boldsymbol{M} \in \{0, 1\}^{r \times r}$ and output the ciphertext

$$\mathrm{C} := \sum_{i,j \in [r] : M_{[i,j]} = 1} \boldsymbol{P}_{(i,j)} \in \mathbb{Z}_q^{(n+r) \times N}, \qquad (4)$$

where $\boldsymbol{M}_{[i,j]}$ is the (i, j) − th element of $\boldsymbol{M}$. The DetePubEnc algorithm has lower computational cost than SecEnc algorithm and PubEnc algorithm, and it only involves matrix addition, whereas the SecEnc algorithm and PubEnc algorithm involve both matrix multiplication and matrix addition.

4. Analysis on Matrix GSW-FHE

In the KeyGen algorithm of matrix GSW-FHE, $\boldsymbol{M}_{(i,j)}\mathbf{S}$ needs to be computed when generating public key $\boldsymbol{P}_{(i,j)}$. We observe that

$$\begin{aligned} \boldsymbol{M}_{(i,j)}\mathbf{S} &= \boldsymbol{M}_{(i,j)}\left(\boldsymbol{I}_r \parallel -\boldsymbol{S}'\right) \\ &= \left(\begin{array}{c|c} & \mathbf{0} \\ \boldsymbol{M}_{(i,j)} & -s_{j1}{}' \ \dots \ -s_{jn}{}' \\ & \mathbf{0} \end{array}\right), \end{aligned} \tag{5}$$

where right matrix is with $(-s'_{j1},\dots,-s_{jn}{}')$ in the i-th row and 0 in other rows. Let $\boldsymbol{M}_{(i,j)}{}' \in \mathbb{Z}_q^{n\times n}$ be an n × n matrix, which satisfies the following matrix equation:

$$\begin{aligned} &\left(\boldsymbol{I}_r \ \ -\boldsymbol{S}'\right)\cdot\begin{pmatrix} \boldsymbol{M}_{(i,j)} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{M}_{(i,j)}{}' \end{pmatrix} \\ &\quad = \left(\begin{array}{c|c} & \mathbf{0} \\ \boldsymbol{M}_{(i,j)} & -s_{j1}{}' \ \dots \ -s_{jn}{}' \\ & \mathbf{0} \end{array}\right). \end{aligned} \tag{6}$$

That is,

$$-\boldsymbol{S}'\cdot\boldsymbol{M}'_{(i,j)} = \begin{pmatrix} \mathbf{0} \\ -s_{j1}{}' \ \dots \ -s_{jn}{}' \\ \mathbf{0} \end{pmatrix}. \tag{7}$$

Viewing the elements of $\boldsymbol{S}'$ as the equation parameter and the elements of $\boldsymbol{M}'_{(i,j)}$ as variables, we can get equations from the above matrix equation:

$$\begin{gathered} s'_{11}\cdot m'_{11}+\cdots+s'_{1n}\cdot m'_{n1}=0 \\ s'_{11}\cdot m'_{12}+\cdots+s'_{1n}\cdot m'_{n2}=0 \\ \vdots \\ s'_{11}\cdot m'_{1n}+\cdots+s'_{1n}\cdot m'_{nn}=0 \\ \vdots \\ s'_{i1}\cdot m'_{11}+\cdots+s'_{in}\cdot m'_{n1}=s'_{j1} \\ s'_{i1}\cdot m'_{12}+\cdots+s'_{in}\cdot m'_{n2}=s'_{j2} \\ \vdots \\ s'_{i1}\cdot m'_{1n}+\cdots+s'_{in}\cdot m'_{nn}=s'_{jn} \\ \vdots \\ s'_{r1}\cdot m'_{11}+\cdots+s'_{rn}\cdot m'_{n1}=0 \\ s'_{r1}\cdot m'_{12}+\cdots+s'_{rn}\cdot m'_{n2}=0 \\ \vdots \\ s'_{r1}\cdot m'_{1n}+\cdots+s'_{rn}\cdot m'_{nn}=0 \end{gathered} \tag{8}$$

According to the knowledge of linear algebra, the equations exit nontrivial solution if the rank of coefficient matrix is equal to the rank of the augmented matrix as below.

$$\begin{aligned} &\mathit{rank}\begin{pmatrix} s'_{11} & s'_{12} & \dots & s'_{1n} \\ & \cdots & & \\ s'_{11} & s'_{12} & \dots & s'_{1n} \\ & \vdots & & \\ s'_{i1} & s'_{i2} & \dots & s'_{in} \\ & \cdots & & \\ s'_{i1} & s'_{i2} & \dots & s'_{in} \\ & \vdots & & \\ s'_{r1} & s'_{r2} & \dots & s'_{rn} \\ & \cdots & & \\ s'_{r1} & s'_{r2} & \dots & s'_{rn} \end{pmatrix}_{rn\times n} \\ &= \mathit{rank}\begin{pmatrix} s'_{11} & s'_{12} & \dots & s'_{1n} & 0 \\ & & \cdots & & \\ s'_{11} & s'_{12} & \dots & s'_{1n} & 0 \\ & & \vdots & & \\ s'_{i1} & s'_{i2} & \dots & s'_{in} & s'_{j1} \\ & & \cdots & & \\ s'_{i1} & s'_{i2} & \dots & s'_{in} & s'_{jn} \\ & & \vdots & & \\ s'_{r1} & s'_{r2} & \dots & s'_{rn} & 0 \\ & & \cdots & & \\ s'_{r1} & s'_{r2} & \dots & s'_{rn} & 0 \end{pmatrix}_{rn\times(n+1)}. \end{aligned} \tag{9}$$

That is,

$$\begin{aligned} &\mathit{rank}\begin{pmatrix} s'_{11} & s'_{12} & \dots & s'_{1n} \\ & \vdots & & \\ s'_{i1} & s'_{i2} & \dots & s'_{in} \\ & \vdots & & \\ s'_{r1} & s'_{r2} & \cdots & s'_{rn} \end{pmatrix}_{r\times n} \\ &= \mathit{rank}\begin{pmatrix} s'_{11} & s'_{12} & \dots & s'_{1n} & 0 \\ & & \vdots & & \\ s'_{i1} & s'_{i2} & \dots & s'_{in} & s'_{j1} \\ & & \cdots & & \\ s'_{i1} & s'_{i2} & \dots & s'_{in} & s'_{jn} \\ & & \vdots & & \\ s'_{r1} & s'_{r2} & \dots & s'_{rn} & 0 \end{pmatrix}_{(r+n-1)\times(n+1)}. \end{aligned} \tag{10}$$

We denote the solution by $\overline{M}_{(i,j)}$, so we have

$$-\boldsymbol{S}' \cdot \overline{\boldsymbol{M}}_{(i,j)} = \begin{pmatrix} \boldsymbol{0} \\ -s_{j1}{}' \ \ \ldots \ \ -s_{jn}{}' \\ \boldsymbol{0} \end{pmatrix} = \boldsymbol{M}_{(i,j)} \cdot \left(-\mathbf{S}'\right). \quad (11)$$

From the above analysis, we can derivate the circular security of the matrix GSW-FHE scheme.

Theorem 1 (circular security). *If the equation*

$$-\boldsymbol{S}' \cdot \boldsymbol{M}'_{(i,j)} = \begin{pmatrix} \boldsymbol{0} \\ -s_{j1}{}' \ \ \ldots \ \ -s_{jn}{}' \\ \boldsymbol{0} \end{pmatrix} \quad (12)$$

exits nontrivial solution $\overline{\mathbf{M}}_{(i,j)}$ *over* $\mathbb{Z}_q$*, then the matrix GSW-FHE scheme is circular secure with function* $f_{M_{(i,j)}}(S)$.

Proof. Let c_1 be a ciphertext encrypting function $f_{M_{(i,j)}}(\boldsymbol{S}) = \begin{pmatrix} \boldsymbol{M}_{(i,j)}\mathbf{S} \\ \boldsymbol{0} \end{pmatrix}\boldsymbol{G} \in \mathbb{Z}_q^{(n+r)\times N}$, $c_1 = \boldsymbol{BR} + \boldsymbol{P}_{(i,j)}$, and $\mathbf{R} \overset{U}{\longleftarrow} \{0,1\}^{m\times N}$. Then we have

$$c_1 = \boldsymbol{BR} + \boldsymbol{P}_{(i,j)} = \boldsymbol{BR} + \boldsymbol{B} \cdot \boldsymbol{R}_{(i,j)} + \begin{pmatrix} \boldsymbol{M}_{(i,j)}\mathbf{S} \\ \boldsymbol{0} \end{pmatrix} \cdot \boldsymbol{G}$$

$$= \begin{pmatrix} \left(\boldsymbol{I}_r \ \ -\mathbf{S}'\right) \cdot \begin{pmatrix} \boldsymbol{E} \\ -\boldsymbol{A} \end{pmatrix} \cdot \left(\mathbf{R} + \boldsymbol{R}_{(i,j)}\right) \\ \boldsymbol{A} \cdot \left(\mathbf{R} + \boldsymbol{R}_{(i,j)}\right) \end{pmatrix} + \begin{pmatrix} \boldsymbol{M}_{(i,j)}\mathbf{S} \\ \boldsymbol{0} \end{pmatrix}$$

$$\cdot \boldsymbol{G} \quad (13)$$

$$= \begin{pmatrix} \left(\boldsymbol{I}_r \ \ -\mathbf{S}'\right) \cdot \begin{pmatrix} \boldsymbol{E} \\ -\boldsymbol{A} \end{pmatrix} \cdot \left(\mathbf{R} + \boldsymbol{R}_{(i,j)}\right) \\ \boldsymbol{A} \cdot \left(\mathbf{R} + \boldsymbol{R}_{(i,j)}\right) \end{pmatrix}$$

$$+ \begin{pmatrix} \left(\begin{array}{c|c} \boldsymbol{M}_{(i,j)} & \begin{matrix} \boldsymbol{0} \\ -s_{j1}{}' \ \ \ldots \ \ -s_{jn}{}' \\ \boldsymbol{0} \end{matrix} \end{array} \right) \\ \boldsymbol{0} \end{pmatrix} \cdot \boldsymbol{G}$$

From (12), we have

$$c_1 = \begin{pmatrix} \left(\boldsymbol{I}_r \ \ -\mathbf{S}'\right)\begin{pmatrix} \boldsymbol{E} \\ -\boldsymbol{A} \end{pmatrix} \cdot \left(\mathbf{R} + \boldsymbol{R}_{(i,j)}\right) + \left(\boldsymbol{I}_r \ \ -\mathbf{S}'\right) \cdot \begin{pmatrix} \boldsymbol{M}_{(i,j)} & \boldsymbol{0} \\ \boldsymbol{0} & \overline{\boldsymbol{M}}_{(i,j)} \end{pmatrix}\boldsymbol{G} \\ \boldsymbol{A} \cdot \left(\mathbf{R} + \boldsymbol{R}_{(i,j)}\right) \end{pmatrix}$$

$$= \begin{pmatrix} \left(\boldsymbol{I}_r \ \ -\mathbf{S}'\right)\begin{pmatrix} \boldsymbol{E} \\ -\boldsymbol{A} \end{pmatrix} \cdot \left(\mathbf{R} + \boldsymbol{R}_{(i,j)}\right) + \left(\boldsymbol{I}_r \ \ -\mathbf{S}'\right) \cdot \begin{pmatrix} \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \overline{\boldsymbol{M}}_{(i,j)} \end{pmatrix}\boldsymbol{G} + \left(\boldsymbol{I}_r \ \ -\mathbf{S}'\right) \cdot \begin{pmatrix} \boldsymbol{M}_{(i,j)} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix}\boldsymbol{G} \\ \boldsymbol{A} \cdot \left(\mathbf{R} + \boldsymbol{R}_{(i,j)}\right) \end{pmatrix}$$

$$= \begin{pmatrix} \left(\boldsymbol{I}_r \ \ -\mathbf{S}'\right)\begin{pmatrix} \boldsymbol{E} \cdot \left(\mathbf{R} + \boldsymbol{R}_{(i,j)}\right) \\ -\boldsymbol{A} \cdot \left(\mathbf{R} + \boldsymbol{R}_{(i,j)}\right) + \overline{\boldsymbol{M}}_{(i,j)} \cdot \left(\boldsymbol{g}^T \otimes \boldsymbol{I}_n\right) \end{pmatrix} + \left(\boldsymbol{I}_r \ \ -\mathbf{S}'\right) \cdot \begin{pmatrix} \boldsymbol{M}_{(i,j)} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix}\boldsymbol{G} \\ \boldsymbol{A} \cdot \left(\mathbf{R} + \boldsymbol{R}_{(i,j)}\right) \end{pmatrix} \quad (14)$$

$$= \begin{pmatrix} \left(\boldsymbol{I}_r \ \ -\mathbf{S}'\right)\begin{pmatrix} \boldsymbol{E} \cdot \left(\mathbf{R} + \boldsymbol{R}_{(i,j)}\right) \\ -\boldsymbol{A} \cdot \left(\mathbf{R} + \boldsymbol{R}_{(i,j)}\right) + \overline{\boldsymbol{M}}_{(i,j)} \cdot \left(\boldsymbol{g}^T \otimes \boldsymbol{I}_n\right) \end{pmatrix} \\ \boldsymbol{A} \cdot \left(\mathbf{R} + \boldsymbol{R}_{(i,j)}\right) - \overline{\boldsymbol{M}}_{(i,j)} \cdot \left(\boldsymbol{g}^T \otimes \boldsymbol{I}_n\right) \end{pmatrix} + \begin{pmatrix} \left(\boldsymbol{I}_r \ \ -\mathbf{S}'\right) \cdot \begin{pmatrix} \boldsymbol{M}_{(i,j)} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix}\boldsymbol{G} \\ \overline{\boldsymbol{M}}_{(i,j)} \cdot \left(\boldsymbol{g}^T \otimes \boldsymbol{I}_n\right) \end{pmatrix} = \begin{pmatrix} \left(\boldsymbol{I}_r \ \ -\mathbf{S}'\right)\begin{pmatrix} \widetilde{\boldsymbol{E}} \\ -\widetilde{\boldsymbol{A}} \end{pmatrix} \\ \widetilde{\boldsymbol{A}} \end{pmatrix}$$

$$+ \begin{pmatrix} \left(\boldsymbol{I}_r \ \ -\mathbf{S}'\right) \cdot \begin{pmatrix} \boldsymbol{M}_{(i,j)} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix}\boldsymbol{G} \\ \overline{\boldsymbol{M}}_{(i,j)} \cdot \left(\boldsymbol{g}^T \otimes \boldsymbol{I}_n\right) \end{pmatrix} = \begin{pmatrix} \boldsymbol{S}'\widetilde{\boldsymbol{A}} + \widetilde{\boldsymbol{E}} \\ \widetilde{\boldsymbol{A}} \end{pmatrix} + \begin{pmatrix} \begin{pmatrix} \boldsymbol{M}_{(i,j)} \cdot \left(\boldsymbol{g}^T \otimes \boldsymbol{I}_r\right) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix} \\ \overline{\boldsymbol{M}}_{(i,j)} \cdot \left(\boldsymbol{g}^T \otimes \boldsymbol{I}_n\right) \end{pmatrix}.$$

$\widetilde{\boldsymbol{E}} \triangleq \boldsymbol{E} \cdot (\mathbf{R} + \boldsymbol{R}_{(i,j)})$; $\widetilde{\boldsymbol{A}} \triangleq \boldsymbol{A} \cdot (\mathbf{R} + \boldsymbol{R}_{(i,j)}) - \overline{\boldsymbol{M}}_{(i,j)} \cdot (\boldsymbol{g}^T \otimes \boldsymbol{I}_n)$; therefore, we derivate that

$$c_1 = \begin{pmatrix} \boldsymbol{S}'\widetilde{\boldsymbol{A}} + \widetilde{\boldsymbol{E}} \\ \widetilde{\boldsymbol{A}} \end{pmatrix} + \begin{pmatrix} \begin{pmatrix} \boldsymbol{M}_{(i,j)} \cdot \left(\boldsymbol{g}^T \otimes \boldsymbol{I}_r\right) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix} \\ \overline{\boldsymbol{M}}_{(i,j)} \cdot \left(\boldsymbol{g}^T \otimes \boldsymbol{I}_n\right) \end{pmatrix}. \quad (15)$$

As $(\widetilde{\boldsymbol{A}}, \boldsymbol{S}'\widetilde{\boldsymbol{A}} + \widetilde{\boldsymbol{E}})$ is an instance of LWE over $\mathbb{Z}_q^{(n+r)\times N}$, it satisfies uniform distribution over $\mathbb{Z}_q^{(n+r)\times N}$. Furthermore, c_1 obeys uniform distribution over $\mathbb{Z}_q^{(n+r)\times N}$.

On the other hand, suppose that c_0 is a ciphertext encrypting $\mathbf{0}$; that is,

$$c_0 = \boldsymbol{B}\boldsymbol{R}' = \begin{pmatrix} \boldsymbol{S}'\boldsymbol{A} + \boldsymbol{E} \\ \boldsymbol{A} \end{pmatrix} \cdot \boldsymbol{R}' \in \mathbb{Z}_q^{(n+r)\times N}, \qquad \mathbf{R}' \xleftarrow{U} \{0,1\}^{m\times N}. \tag{16}$$

It is also an instance of LWE over $\mathbb{Z}_q^{(n+r)\times N}$ and obeys uniform distribution over $\mathbb{Z}_q^{(n+r)\times N}$, too. Therefore, distributions of c_0 and c_1 are computationally indistinguishable, and the advantage of probabilistic polynomial-time adversary $\mathscr{A}$ is negligible. So we can conclude that the matrix GSW-FHE is circular secure with function $f_{M_{(i,j)}}(S)$.

From Theorem 1, we can choose a good secret key that satisfies that (12) has solution via "reject sample" technique and obtain circular secure matrix GSW-FHE scheme. □

5. Optimizing Bootstrapping

In this section, we describe how to optimize the bootstrapping procedure of [13] by introducing deterministic homomorphic plaintext slot-wise permutation.

5.1. Motivation. The decryption of all LWE-based FHE schemes consists of the inner product and rounding: for secret key s $\in \mathbb{Z}_q^d$ and a binary ciphertext $\boldsymbol{c} \in \{0,1\}^d$, the decryption algorithm computes

$$\mathrm{Dec}\,(\mathrm{s},\mathrm{c}) = \lfloor \langle s,c \rangle \rceil_2 \in \{0,1\}. \tag{17}$$

Note that the inner product itself is just a subset-sum of the $\mathbb{Z}_q$-entries of s indicated by c and uses only the additive group structure of $\mathbb{Z}_q$. Alperin-Sheriff and Peikert [20] proposed an efficient bootstrapping algorithm by embedding $\mathbb{Z}_q$ into permutation group S_q. Thus the rounding function is no longer just a sum, and it can be expressed as

$$\lfloor x \rceil_2 = \sum_{v\in\mathbb{Z}_q \ s.t. \lfloor v \rceil_2 = 1} [x = v], \tag{18}$$

where each equality test $[x = v]$ returns 0 for false and 1 for true. The equality test operation has homomorphic counterpart, called homomorphic equality test. Homomorphic equality test is an important primitive for optimizing bootstrapping procedure, and it has many other applications as mentioned in [25].

For $x, v \in \mathbb{Z}_r$, they map to the r-by-r permutation matrices of group S_r and are denoted as τ *and* σ, respectively. The Eq? algorithm is described as follows:

(i) **Eq**? ($C^\tau = c_{i,j}^\tau, \sigma \in S_r$): given a ciphertext encrypting some permutation $\tau \in S_r$ and a permutation $\sigma \in S_r$ (in the clear), output a ciphertext c encrypting 1 if $\tau = \sigma$; otherwise, output a ciphertext c encrypting 0:

$$\mathrm{c} \longleftarrow \boxdot_{i\in[r]} c_{\sigma(i),j}^{\tau} \boxdot \mathrm{g}. \tag{19}$$

Note that the permutation σ goes through all permutations in S_r, and it is not masked in the homomorphic equality test **Eq**? Algorithm; that is, $\sigma \in S_r$ is *in the clear*.

Let φ_i: $\mathbb{Z}_q \longrightarrow \{0,1\}^r$ be the isomorphism of an element in $\mathbb{Z}_q$ (q $:= \prod_{i=1}^t r_i$) into the cyclic permutation that corresponds to an element in $\mathbb{Z}_{r_i}$, where r $\triangleq \max_i\{r_i\}$. During homomorphic rounding process of work [13], $\varphi_i(x)$ is encrypted as part of public bootstrapping key and used in the homomorphic equality test algorithm.

In fact, x traverses $\mathbb{Z}_q$ and does not carry any privacy information. It is not necessary to encrypt $\pi_{\varphi_i(x)}$ using SecEnc algorithm, which would increase computation cost. We propose optimizing homomorphic equality test algorithm by defining hybrid homomorphic plaintext slot-wise switching method, which reduces the computation cost of bootstrapping key generation.

5.2. Hybrid Homomorphic Plaintext Slot-Wise Switching. Plaintext slot-wise permutation is an important operation in application of packed FHE [23, 24]. It can be achieved by multiplying the encryption of a permutation and its inverse from left and right. We propose hybrid homomorphic plaintext slot switching procedure where the switch key is encrypted by symmetric and asymmetric encryption algorithm. The nice feature of our switching procedure is that part of switch key can be computed by deterministic public encryptions, which makes our procedure more efficient than that of [13].

(i) SwitchKeyGen($\boldsymbol{S},\sigma$): Input a secret key matrix $\boldsymbol{S} \in \mathbb{Z}_q^{r\times(n+r)}$ and a permutation σ; let $\pi_\sigma \in \{0,1\}^{r\times r}$ be a matrix corresponding to σ, and compute

$$\begin{aligned} W_\sigma &\longleftarrow \mathrm{SecEnc}_S(\pi_\sigma), \\ W_{\sigma^{-1}} &\longleftarrow \mathrm{SecEnc}_S\left({\pi_\sigma}^T\right). \end{aligned} \tag{20}$$

Output the switch key ssk$_\sigma$:= $(W_\sigma, W_{\sigma^{-1}})$. The algorithm is the same as the work in [13].

(ii) *SlotSwitch*$_{ssk_\sigma}$(C): Input a switch key ssk$_\sigma$ and a ciphertext C; output

$$C_\sigma \longleftarrow W_\sigma \bigodot \left(C \bigodot \left(W_{\sigma^{-1}} \bigodot G\right)\right), \tag{21}$$

where $G \in \mathbb{Z}_q^{(n+r)\times N}$ is the fixed encryption of I_r with noise zero.

(iii) DeteSwitchKeyGen($\boldsymbol{S},\sigma$): Input a secret key matrix $\boldsymbol{S} \in \mathbb{Z}_q^{r\times(n+r)}$ and a permutation σ, and compute

$$\begin{aligned} DW_\sigma &\longleftarrow \mathrm{DetePubEnc}_S(\pi_\sigma), \\ DW_{\sigma^{-1}} &\longleftarrow \mathrm{DetePubEnc}_S\left({\pi_\sigma}^T\right). \end{aligned} \tag{22}$$

Output the deterministic switch key dssk$_\sigma$:= $(DW_\sigma, DW_{\sigma^{-1}})$.

(iv) *DeteSlotSwitch*$_{dssk_\sigma}$(C): Input a deterministic switch key dssk$_\sigma$ and a ciphertext C; output

$$C_\sigma \longleftarrow DW_\sigma \bigodot \left(C \bigodot \left(DW_{\sigma^{-1}} \bigodot G\right)\right), \tag{23}$$

where $G \in \mathbb{Z}_q^{(n+r)\times N}$ is the fixed encryption of I_r with noise zero.

5.3. Optimized Bootstrapping Procedure. Our optimized bootstrapping procedure can be used to refresh ciphertexts of all standard LWE-based FHE. Let $\boldsymbol{c} \in \{0,1\}^d$ be the ciphertext to be bootstrapped, and let $\mathrm{s} \in \mathbb{Z}_q^d$ be a secret key that corresponds to $\boldsymbol{c}$**.** The optimized bootstrapping procedure consists of two algorithms, HybirdBootKeyGen and HybirdBootstrap.

(i) **HybridBootKeyGen**$(pk, sk, \boldsymbol{s})$: Input a secret key sk and public key pk for our bootstrapping scheme and the secret key $\mathbf{s} = (s_1, \ldots, s_d) \in \mathbb{Z}_q^d$ for ciphertext to be refreshed; output a bootstrapping key bk. For every $\mathrm{i} \in [\mathrm{t}]$ and $\mathrm{j} \in [\mathrm{d}]$, let $\pi_{\varphi_i(s_j)}$ be the permutation corresponding to $\varphi_i(s_j)$, and generate

$$\begin{aligned} \tau_{i,j} &\overset{R}{\longleftarrow} \mathrm{SecEnc}_{sk}\left(\mathrm{diag}\left(\varphi_i\left(s_j\right)\right)\right), \\ ssk_{i,j} &\overset{R}{\longleftarrow} \mathrm{SwitchKeyGen}\left(\mathrm{sk}, \pi_{\varphi_i(s_j)}\right), \end{aligned} \tag{24}$$

where, for a vector $x \in \mathbb{Z}^r$, $\mathrm{diag}(x) \in \mathbb{Z}^{r\times r}$ is the square integer matrix that has x in its diagonal entries and 0 in the others. Then compute the hints used in homomorphic equality test on packed indictor vectors. For every $\mathrm{i} \in [\mathrm{t}]$ and $x \in \mathbb{Z}_q$ such that $\lfloor x \rceil_2 = 1$, compute

$$dssk_{\varphi_i(x)} \longleftarrow DeteSwitchKeyGen\left(sk, \pi_{\varphi_i(x)}\right). \tag{25}$$

Output the bootstrapping key

$$bk := \left\{\tau_{i,j}, ssk_{i,j}, dssk_{\varphi_i(x)}\right\}_{i\in[t], j\in[d], x\in\mathbb{Z}_q : \lfloor x \rceil_2 = 1}. \tag{26}$$

(ii) $\boldsymbol{HybridBootstrap}_{bk}(\mathbf{c})$: Input a bootstrapping key bk and a ciphertext $\boldsymbol{c} \in \{0,1\}^d$; output the refreshed ciphertext C^*. All the FHE schemes based on the LWE problem have similar decryption algorithm; that is, the decryption algorithm needs to compute $\lfloor \langle s, c\rangle \rceil_2$. There are two phases in the HybridBootstrap algorithm: evaluate the inner product and rounding.

Inner Product: For every $\mathrm{i} \in [\mathrm{t}]$, homomorphically compute an encryption of $\varphi_i(\langle s, c\rangle)$. Let $\mathrm{h} := \min\{\mathrm{j} \in [\mathrm{d}] : c_j = 1\}$. For $\mathrm{i} = 1, 2, \ldots, \mathrm{t}$, set $C_i^* := \tau_{i,h}$, and iteratively compute

$$C_i^* \overset{R}{\longleftarrow} SlotSwitch_{ssk_{i,j}}\left(C_i^*\right) \tag{27}$$

for $\mathrm{j} = \mathrm{h}+1, \ldots, \mathrm{d}$ such that $c_j = 1$.

Rounding: For each $x \in \mathbb{Z}_q$ such that $\lfloor x \rceil_2 = 1$, homomorphically test the equality between x and $\langle s, c\rangle$, and sum their results. The refreshed ciphertext is computed as

$$\begin{aligned} C^* \longleftarrow \bigoplus_{x\in\mathbb{Z}_q : \lfloor x \rceil_2 = 1} &\left(\bigodot_{i\in[t]} \left(DeteSlotSwitch_{dssk_{\varphi_i(x)}}\left(C_i^*\right)\right)\right. \\ &\left.\bigodot P_{1,1}\right). \end{aligned} \tag{28}$$

5.4. Correctness Analysis

Lemma 2 (correctness). *Let sk be the secret key for our scheme. Let c and s be a ciphertext and secret key of LWE-based FHE scheme. Then, for bk* $\longleftarrow$ HybridBootKeyGen$(pk, sk, \boldsymbol{s})$, *the refreshed ciphertext* $C^* \longleftarrow HybridBootstrap_{bk}(\mathrm{c})$ *is designed to encrypt* $Dec_s(c) = \lfloor \langle s, c\rangle \rceil_2 \in \{0,1\}$ *in the first slot.*

Proof. Firstly, C_i^* is designed to encrypt $\varphi_i([\langle s, c\rangle]_q)$, and

$$\bigodot_{i\in[t]} \left(DeteSlotSwitch_{dssk_{\varphi_i(x)}}\left(C_i^*\right)\right) \bigodot P_{1,1} \tag{29}$$

is designed to encrypt 1 in the first slot if and only if $x = \langle s, c\rangle \ mod\ q$. Finally, since the homomorphic sum is taken over every $x \in \mathbb{Z}_q$ such that $\lfloor x \rceil_2 = 1$, C^* is designed to encrypt 1 if and only if $\lfloor \langle s, c\rangle \rceil_2 = 1$. □

5.5. Security Analysis. If the bootstrapping scheme secret key sk is generated independently of the secret keys s of FHE scheme from LWE, then Ind-CPA security of the bootstrapping key follows immediately from the security of hybrid homomorphic plaintext slot-wise switching, and the security of hybrid homomorphic plaintext slot-wise switching scheme resorts to the security of matrix GSW-FHE and hence the security of our bootstrapping scheme from LWE assumption.

5.6. Performance Analysis. Let $q = \widetilde{O}(\lambda)$ be the modules of the ciphertext to be refreshed, and q has the form $q := \prod_{i=1}^{t} r_i$, where r_i are small and powers of distinct primes. The following lemma allows us to choose a sufficiently large q by letting it be the product of all maximal prime powers r_i bounded by $\mathrm{O}(\log\lambda)$, and then there exists $\mathrm{t} = \mathrm{O}(\log\lambda/\log\log\lambda)$, where λ is security parameter.

Lemma 3 (see [13, 20]). *For all* $x \geq 7$, *the product of all maximal prime powers* $r_i \leq x$ *is all at least* $\exp(3x/4)$.

On one hand, our DetePubEnc algorithm involves matrix additions operation only, whereas SecEnc algorithm involves many matrix multiplication operations. Our bootstrapping key $dssk_{\varphi_i(x)}$ is optimized from $ssk_{\varphi_i(x)}$. Therefore, our optimized bootstrapping key generation has lower computation complexity. The comparison of computational complexity is illustrated in Table 1.

On the other hand, we may implement a trade-off between computation and storage complexity. For every $k, l \in [\mathrm{r}]$, $P_{k,l} = \mathrm{SecEnc}_{sk}(M_{k,l})$ can be used as public bootstrapping key, delete $dssk_{\varphi_i(x)}$ from the bootstrapping key, and compute $dssk_{\varphi_i(x)}$ online when running rounding procedure. In view of $dssk_{\varphi_i(x)}$ being obtained by DetePubEnc algorithm, its computation involves only matrix additions. Therefore, our optimized bootstrapping drastically cuts down the size of the large public bootstrapping key by a third, paying matrix additions with negligible computation complex. The comparison of storage complexity is illustrated in Table 2.

Table 1: Comparison of computational complexity.

Bootstrapping key	MM	MA
$ssk_{\varphi_i(x)}$ [13],$0 \leq i \leq t$	$\mathrm{O}\left(\log \lambda / \log \log \lambda\right)$	$\mathrm{O}\left(\log \lambda / \log \log \lambda\right)$
$dssk_{\varphi_i(x)}$ [ours], $0 \leq i \leq t$	0	$\mathrm{O}\left(\log^2 \lambda / \log \log \lambda\right)$

Note: MM denotes matrix multiplication operation; MA denotes matrix addition operation.

Table 2: Comparison of storage complexity of bootstrapping key.

Work	Bootstrapping key
[13]	$\left\{\left(\tau_{i,j}, ssk_{i,j}, ssk_{\varphi_i(x)}\right)\right\}_{i\in[t],j\in[d],x\in\mathbb{Z}_q:\lfloor x \rceil_2=1}$
[ours]-1	$\left\{\left(\tau_{i,j}, ssk_{i,j}, dssk_{\varphi_i(x)}\right)\right\}_{i\in[t],j\in[d],x\in\mathbb{Z}_q:\lfloor x \rceil_2=1}$
[ours]-2	$\left\{\left(\tau_{i,j}, ssk_{i,j}\right)\right\}_{i\in[t],j\in[d]}$

Note: [ours]-1 denotes save computation complexity in the cost of the storage complexity; [ours]-2 denotes save storage complexity in the cost of computation complexity.

6. Conclusions

Matrix GSW-FHE scheme encrypts multibit message and supports complex homomorphic matrix operations and can be used to optimize the bootstrapping procedure. We analyse circular security of matrix GSW-FHE scheme and derive a sufficient condition of circular security for matrix GSW-FHE scheme. That is, if the equations about secret key have solution over $\mathbb{Z}_q$, the matrix GSW-FHE scheme satisfies circular security with function $f_{M_{(i,j)}}(S)$. Therefore, we can choose a good secret key that satisfies the sufficient condition via "reject sample" technique and furthermore obtain circular secure matrix GSW-FHE scheme.

We also propose hybrid homomorphic plaintext slot-wise switching method by defining deterministic public encryption algorithm in matrix GSW-FHE, which significantly reduces computational complex or space complex of bootstrapping key generation, thus optimizing the bootstrapping procedure of Hiromasa and so forth. Meanwhile, performance analysis validates the effectiveness of the proposed optimized bootstrapping scheme.

Some questions remain for further study, such as the probability analysis of our sufficient condition and the sufficient and necessary condition for circular security of the matrix GSW-FHE scheme [26]. And to make a fair comparison with the state-of-the-art bootstrapping schemes such as FHEW [21], WT [22], and so forth, detailed security, parameters, and efficiency experiment analysis remain to be a future work.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Disclosure

The abstract of this manuscript has been submitted to the 4th International Conference on Cloud Computing and Security, but it has not been published; and this manuscript cites the conference paper in the references.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant no. 61601515 and Natural Science Foundation of Henan Province under Grant no. 162300410332.

References

[1] Z. Pan, J. Lei, Y. Zhang, and F. L. Wang, "Adaptive fractional-Pixel motion estimation skipped algorithm for efficient HEVC motion estimation," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 14, no. 1, pp. 1–19, 2018.

[2] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proceedings of the 41st annual ACM symposium on Theory of Computing (STOC '09)*, pp. 169–178, ACM, Bethesda, Md, USA, 2009.

[3] C. Gentry, *A fully homomophic encryption scheme [Ph.D. thesis]*, Stanford University, 2009, http://crypto.stanford.edu/craig.

[4] Y. Liu, H. Peng, and J. Wang, "Verifiable diversity ranking search over encrypted outsourced data," *CMC*, vol. 55, no. 1, pp. 37–57, 2018.

[5] W. Xu, S. Xiang, and V. Sachney, "A cryptography domain image retrieval method based on Paillier homomorphic block encryption," *CMC*, vol. 55, no. 2, pp. 285–295, 2018.

[6] R. Xie, C. He, D. Xie, C. Gao, and X. Zhang, "A Secure Ciphertext Retrieval Scheme against Insider KGAs for Mobile Devices in Cloud Storage," *Security and Communication Networks*, vol. 2018, Article ID 7254305, 7 pages, 2018.

[7] R. L. Rivest, L. Adleman, and M. L. Dertouzos, *On Data Banks And Privacy Homomorphism Proc of Foundations of Secure Computation*, Academic Press, New York, NY, USA, 1978.

[8] Z. Brakerski and V. Vaikuntanathan, "Efficient fully homomorphic encryption from (standard) LWE," in *Proceedings of the IEEE 52nd Annual Symposium on Foundations of Computer Science (FOCS '11)*, pp. 97–106, Palm Springs, Calif, USA, October 2011.

[9] M. R. Albrecht, R. Player, and S. Scott, "On the concrete hardness of learning with errors," *Journal of Mathematical Cryptology*, vol. 9, no. 3, pp. 169–203, 2015.

[10] Z. Brakerski and V. Vaikuntanathan, "Fully homomorphic encryption from ring-LWE and security for key dependent

messages," in *Advances in Cryptology—CRYPTO 2011*, R. Phillip, Ed., vol. 6841, pp. 505–524, Springer, Berlin, Germany, 2011.

[11] F. Luo, F. Wang, K. Wang, J. Li, and K. Chen, "LWR-Based Fully Homomorphic Encryption," *Security and Communication Networks*, vol. 2018, Article ID 5967635, 12 pages, 2018.

[12] X. Yang, T. Zhou, W. Zhang, and L. Wu, "Application of a circular secure variant of LWE in the homomorphic encryption," *Jisuanji Yanjiu yu Fazhan/Computer Research and Development*, vol. 52, no. 6, pp. 1389–1393, 2015.

[13] R. Hiromasa, M. Abe, and T. Okamoto, "Packing messages and optimizing bootstrapping in GSW-FHE," in *Public-key cryptography—PKC 2015*, vol. 9020 of *Lecture Notes in Comput. Sci.*, pp. 699–715, Springer, Heidelberg, 2015.

[14] D. Hofheinz and D. Unruh, "Towards key-dependent message security in the standard model," in *Advances in cryptology—EUROCRYPT 2008*, vol. 4965 of *Lecture Notes in Comput. Sci.*, pp. 108–126, Springer, Berlin, 2008.

[15] I. Haitner and T. Holenstein, "On the (im)possibility of key dependent encryption," in *Theory of cryptography*, vol. 5444 of *Lecture Notes in Comput. Sci.*, pp. 202–219, Springer, Berlin, 2009.

[16] D. Boneh, S. Halevi, M. Hamburg, and R. Ostrovsky, "Circular-secure encryption from decision Diffie-Hellman," in *Advances in Cryptology*, D. Wagner, Ed., vol. 5157 of *Lecture Notes in Computer Science*, pp. 108–125, Springer, 2008.

[17] B. Applebaum, D. Cash, C. Peikert, and A. Sahai, "Fast cryptographic primitives and circular-secure encryption based on hard learning problems," in *Advances in Cryptology—CRYPTO 2009*, vol. 5677 of *Lecture Notes in Computer Science*, pp. 595–618, Springer, Germany, Berlin, 2009.

[18] O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," in *Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC '05)*, pp. 84–93, ACM, Baltimore, Md, USA, May 2005.

[19] C. Gentry, A. Sahai, and B. Waters, "Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based," *Proceedings of CRYPTO 2013*, vol. 8042, no. 1, pp. 75–92, 2013.

[20] J. Alperin-Sheriff and C. Peikert, "Faster bootstrapping with polynomial error," in *Proceedings of the International Cryptology Conference*, pp. 297–314, Springer, Berlin, Germany, 2014.

[21] L. Ducas and D. Micciancio, "FHEW: Bootstrapping Homomorphic Encryption in Less Than a Second," in *Proceedings of the Advances in Cryptology – EUROCRYPT*, pp. 617–640, Springer Berlin Heidelberg, 2015.

[22] H. Wang and Q. Tang, "Efficient homomorphic integer polynomial evaluation based on GSW FHE," *The Computer Journal*, vol. 61, no. 4, pp. 575–585, 2018.

[23] N. P. Smart and F. Vercauteren, "Fully homomorphic SIMD operations," *Designs, Codes and Cryptography*, vol. 71, no. 1, pp. 57–81, 2014.

[24] Z. Brakerski, C. Gentry, and S. Halevi, "Packed Ciphertexts in LWE-Based Homomorphic Encryption," in *Public-Key Cryptography – PKC 2013*, vol. 7778 of *Lecture Notes in Computer Science*, pp. 1–13, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[25] Y. Wang, H. Pang, N. H. Tran, and R. H. Deng, "CCA Secure encryption supporting authorized equality test on ciphertexts in standard model and its applications," *Information Sciences*, vol. 414, pp. 289–305, 2017.

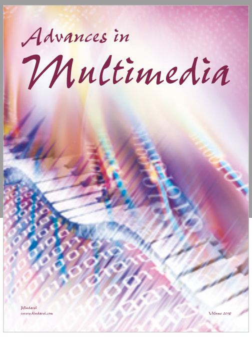
[26] X. Zhao, H. Mao, S. Liu, and W. Song, "Circular-secure analysis on matrix GSW-FHE and optimizing bootstrapping," in *Proceedings of the International Conference on Cloud Computing and Security, ICCCS 2018*, 2018.

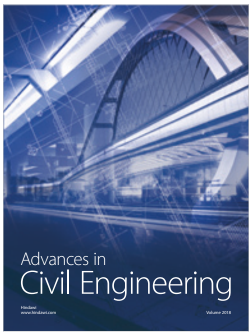
Journal of
Engineering

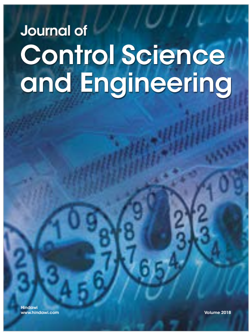
The Scientific
World Journal

International Journal of
Rotating
Machinery

Journal of
Sensors

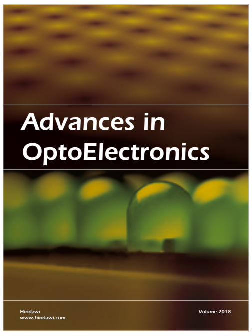

Advances in
Multimedia


Advances in
Civil Engineering


Journal of
Control Science
and Engineering

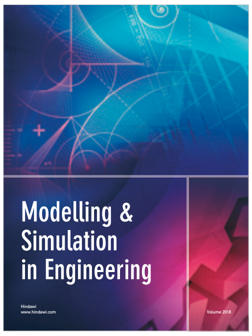
Journal of
Robotics

Journal of
Electrical and Computer
Engineering

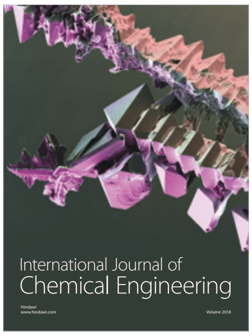

Advances in
OptoElectronics

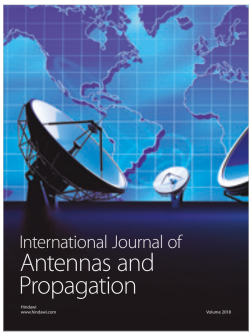
VLSI Design

International Journal of
Navigation and
Observation

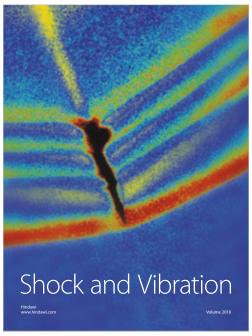

Modelling &
Simulation
in Engineering

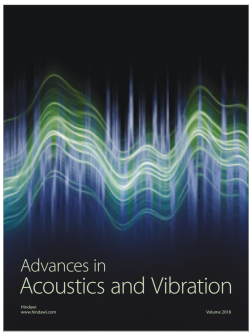
International Journal of
Aerospace
Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and
Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration