TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 02/13 pp564–578
DOI: 10.26599/TST.2019.9010076
Volume 25, Number 5, October 2020

Modified Multi-Key Fully Homomorphic Encryption Based on NTRU Cryptosystem without Key-Switching

Xiaoliang Che, Tanping Zhou, Ningbo Li, Haonan Zhou, Zhenhua Chen, and Xiaoyuan Yang*

Abstract: The Multi-Key Fully Homomorphic Encryption (MKFHE) based on the NTRU cryptosystem is an important alternative to the post-quantum cryptography due to its simple scheme form, high efficiency, and fewer ciphertexts and keys. In 2012, López-Alt et al. proposed the first NTRU-type MKFHE scheme, the LTV12 scheme, using the key-switching and modulus-reduction techniques, whose security relies on two assumptions: the Ring Learning With Error (RLWE) assumption and the Decisional Small Polynomial Ratio (DSPR) assumption. However, the LTV12 and subsequent NTRU-type schemes are restricted to the family of power-of-2 cyclotomic rings, which may affect the security in the case of subfield attacks. Moreover, the key-switching technique of the LTV12 scheme requires a circular application of evaluation keys, which causes rapid growth of the error and thus affects the circuit depth. In this paper, an NTRU-type MKFHE scheme over prime cyclotomic rings without key-switching is proposed, which has the potential to resist the subfield attack and decrease the error exponentially during the homomorphic evaluating process. First, based on the RLWE and DSPR assumptions over the prime cyclotomic rings, a detailed analysis of the factors affecting the error during the homomorphic evaluations in the LTV12 scheme is provided. Next, a Low Bit Discarded & Dimension Expansion of Ciphertexts (LBD&DEC) technique is proposed, and the inherent homomorphic multiplication decryption structure of the NTRU is proposed, which can eliminate the key-switching operation in the LTV12 scheme. Finally, a leveled NTRU-type MKFHE scheme is developed using the LBD&DEC and modulus-reduction techniques. The analysis shows that the proposed scheme compared to the LTV12 scheme can decrease the magnitude of the error exponentially and minimize the dimension of ciphertexts.

Key words: NTRU-type Multi-Key Fully Homomorphic Encryption (MKFHE); prime cyclotomic rings; Low Bit Discarded (LBD); homomorphic multiplication decryption structure

1 Introduction

The Multi-Key Fully Homomorphic Encryption (MKFHE) can perform arbitrary operations on encrypted data at different public keys (users), and the final ciphertext can be jointly decrypted by all the involved users. Moreover, the operation process between the ciphertexts of different users can be entrusted to the cloud offline, avoiding the interaction between users in the secure Multi-Party Computing (MPC) protocol. So, the MKFHE can be widely used in ciphertext retrieval[1], secure MPC[2–4], privacy-preserving protocol[5], etc.

In 2012, López-Alt et al.[6] proposed the concept of MKFHE for the first time and constructed the first MKFHE scheme LTV12 based on the NTRU cryptosystem (called NTRU-type MKFHE). Many studies on the MKFHE have been reported[7–14]. Similar

• Xiaoliang Che, Tanping Zhou, Ningbo Li, Haonan Zhou, and Xiaoyuan Yang are with the Engineering University of People's Armed Police, Xi'an 710086, China. E-mail: smo_mrche@yeah.net; 850301775@qq.com; 372726936@qq.com; 1957028@qq. com; xawjchexl@126.com.
• Zhenhua Chen is with Xi'an University of Science and Technology, Xi'an 710054, China. E-mail: zhenhua@uow.edu.au.
* To whom correspondence should be addressed.
Manuscript received: 2019-10-29; revised: 2019-12-05; accepted: 2019-12-06

to the traditional single-key Fully Homomorphic Encryption (FHE), the MKFHE mainly includes NTRU-type MKFHE, GSW-type MKFHE[8–11], and BGV-type MKFHE[12]. Among the three types of MKFHE schemes, the NTRU-type MKFHE scheme is the fastest in encryption and decryption, and has the simplest form, and uses the least ciphertexts and keys. The underlying scheme of the NTRU encryption has been used to design various cryptographic primitives, including the digital signatures[15], identity-based encryption[16,17], and multi-linear maps[18,19].

The security of the NTRU-type homomorphic encryption schemes is based on the Ring Learning With Error (RLWE) assumption and Decisional Small Polynomial Ratio (DSPR) assumption. Stehlé and Steinfeld showed that the DSPR assumption could be reduced to the RLWE assumption under certain conditions (refer to Ref. [20] for details). The RLWE represents an algebraic variant of the LWE[21], whose hardness can be reduced to the hardness of the worst-case problems on ideal lattices in the standard model. However, recently, it has been shown that subfield attacks[22–25] affected the asymptotic security of NTRU-type schemes for large moduli q. Yu et al.[26,27] considered a variant of NTRU encrypt over prime cyclotomic rings and obtained the INDistinguish-ability under Chosen-Plaintext Attack (IND-CPA) secure results in the standard model assuming the hardness of the worst-case problems on ideal lattices, which was shown to be a good choice to resist the subfield attacks.

In LTV12[6], the leveled NTRU-type MKFHE scheme was adopted by using the key-switching (known as the relinearization) and modulus-reduction techniques[28,29]. However, the key-switching process needed to be carried out before reducing the modulus during the homomorphic evaluations, which increased the error significantly. Hitherto, much work on the design and security research of the NTRU-type FHE scheme has been done[30,31], but there are few outstanding results of the NTRU-type MKFHE.

The main contributions of this work can be summarized as follows.

(1) The NTRU-type MKFHE (LTV12) over prime cyclotomic rings is adopted, and the parameters that affect the growth of error during the homomorphic evaluations are analyzed.

(2) A Low Bit Discarded and Dimension Expansion of Ciphertexts (LBD&DEC) technique is used to modify the inherent decryption structure of the NTRU-type MKFHE, which eliminates the key-switching in homomorphic multiplication and reduces the ciphertext dimension.

(3) A leveled NTRU-type MKFHE scheme over prime cyclotomic rings, which successfully eliminates the relinearization and greatly decreases the error magnitude, is designed.

The rest of this paper is organized as follows. In Section 2, the basic mathematical techniques used in this work, and the RLWE and DSPR assumptions are presented. In Section 3, the NTRU-type MKFHE over prime cyclotomic rings is introduced, and its cryptographic properties are analyzed. In Section 4, the inherent homomorphic decryption structure is modified by using the LBD&DEC technique, and the detailed analysis of the size of error, ciphertext, and evaluation key, etc., is provided. In Section 5, a multi-key somewhat homomorphic encryption scheme is designed by using the LBD&DEC technique, and the parameter comparison between our scheme and the LTV12 scheme is given. In Section 6, a leveled NTRU-type MKFHE scheme without key-switching is presented. The conclusion is provided in Section 7.

2 Preliminary

Assume λ denotes the security parameter, and $\mathrm{negl}(\lambda)$ denotes a negligible function of λ; $\boldsymbol{a}$ denotes the row vector, a_i represents the i-th element of $\boldsymbol{a}$, and $\boldsymbol{a}^{\mathrm{T}}$ represents the column vector. The element located in the i-th row and the j-th column of matrix $\boldsymbol{C}$ is represented as $\boldsymbol{C}[i,j]$. In general, vectors can be regarded as a row matrix. Let $\boldsymbol{v}$ and $\boldsymbol{w} \in R^m$, where R denotes the cyclotomic rings, and assume the dimension of vector $\boldsymbol{v}$ and $\boldsymbol{w}$ is m, so $\boldsymbol{v}\cdot\boldsymbol{w} = \langle \boldsymbol{v}, \boldsymbol{w} \rangle = v_1 w_1 + v_2 w_2 + \cdots + v_k w_k$ denotes the inner product, $\boldsymbol{v}$ and $\boldsymbol{w} \in R^m$.

In this paper, the prime cyclotomic rings $R = \mathbf{Z}[x]/\Phi_n(x)$ and $R_q = R/qR$ are used, where $\Phi_n(x) = x^{n-1} + x^{n-2} + \cdots + 1$ (n denotes a prime), $q = q(\lambda)$ denotes a prime, and it satisfies $q = 1 \bmod n$. Addition and multiplication operation on these rings are component-wise in their coefficients, and the coefficients of R_q are reduced to the range $[-q/2, q/2)$, except for $q = 2$. We require the ability to sample from the probability distribution χ, i.e., the truncated discrete Gaussian distribution $D_{\mathbf{Z}^n,\sigma}$, with Gaussian parameter σ, deviation $r = \sigma\sqrt{2\pi}$, and Gaussian function $\mathrm{e}^{-\pi x^2/\sigma^2}$. Refer to Ref. [32] for a detailed description of the discrete Gaussian distribution. Let $\chi = \chi(\lambda)$ be a B-bound error distribution over R whose coefficients are

in the range $[-B, B]$. For the probability distribution D, $x \leftarrow D$ denotes that x is sampled from D.

The vector length is generally measured by the Euclidean norm. For $v \in R$, we use $\|v\|_\infty = \max_{0 \leqslant i \leqslant n-1} |v_i|$ to denote the standard l_∞-norm and use $\|v\|_1 = \sum_{i=0}^{n-1} |v_i|$ to denote the standard l_1-norm.

The security of our scheme is based on the RLWE and DSPR assumptions. Following Refs. [26] and [27], a brief introduction to these assumptions over the prime cyclotomic rings is provided.

Definition 1 RLWE assumption: Let λ be a security parameter. $q = q(\lambda) \in \mathbf{Z}$ is a prime integer. $\Phi_n(x) = x^{n-1} + x^{n-2} + \cdots + 1$ (n is a prime) is the sub-cyclotomic polynomial. For the polynomial ring defined by $R = \mathbf{Z}[x]/\Phi_n(x)$ and $R_q = R/qR$, and an error distribution $\chi = \chi(\lambda)$ over R, the RLWE assumption states that the following two distributions cannot be distinguished: (1) one samples $(\boldsymbol{a}_i, b_i)$ uniformly from R_q^{n+1}, and (2) one first draws $\boldsymbol{a}_i \leftarrow R_q^n$ uniformly, and samples $(\boldsymbol{a}_i, b_i) \in R_q^{n+1}$ by choosing $\boldsymbol{s}_i \leftarrow R_q^n$ and $e_i \leftarrow \chi$ uniformly, and set $b_i = \langle \boldsymbol{a}_i, \boldsymbol{s}_i \rangle + e_i$.

Definition 2 DSPR assumption: Let λ be a security parameter. $q = q(\lambda) \in \mathbf{Z}$ is a prime integer. $\Phi_n(x) = x^{n-1} + x^{n-2} + \cdots + 1$ (n is a prime) is the sub-cyclotomic polynomial. For the polynomial ring $R = \mathbf{Z}[x]/\Phi_n(x)$ and $R_q = R/qR$, and a B-bound error distribution $\chi = \chi(\lambda)$ over R, the DSPR assumption states that the following two distributions cannot be distinguished: (1) a polynomial $h = 2g/f$, where $f = 2f' + 1$, and it is reversible over R_q, and (2) a polynomial h sampled uniformly at random over R_q.

Two subroutines: Here two subroutines BitDecomp( ) and Powersof 2( ), which are widely used in the FHE schemes, are introduced. Assuming that $l = \lceil \log q \rceil$, these two subroutines can be expressed as follows.

BitDecomp ($x \in R_q$): $R_q \mapsto R_2^l$. On input $x \in R_q$, outputs $x \mapsto (x_0, x_1, \ldots, x_{l-1}) \in \{0, 1\}^l$ (For convenience, we denote $\mathrm{BitD}(x) \in R_q^l$).

Powersof2 ($x \in R_q$): $R_q \mapsto R_q^l$. On input $x \in R_q$, outputs $x \mapsto (x, 2x, \ldots, 2^{l-1}x)$, where $2^{l-1} < q/2$ (For convenience, we denote $\mathrm{Pof2}(x) \in R_q^l$).

It's obviously to verify that $\langle \mathrm{BitD}(x), \mathrm{Pof}\,2(y) \rangle = \langle x, y \rangle \bmod q$, where $\langle x, y \rangle$ denotes the product of polynomials $x, y \in R_q$.

Key-switching technique: The relinearization technique in the LTV12 scheme is also known as the key-switching technique[28, 29]. The key-switching technique can be used to reduce the dimension of expanded ciphertext to the normal level. Generally, it can be used to transform a ciphertext $c \in R_q$ (under the secret key f) to another ciphertext $c_{\mathrm{evk}} \in R_q$ (under the secret key f_{evk}) while the corresponding message stays unchanged. Let $l = \lceil \log q \rceil$, the key-switching process mainly consists of two procedures.

(1) KeySwitchGen ($f \in R$; $f_{\mathrm{evk}} \in R_q$): For $h \in R_q$, $\boldsymbol{s}_\tau, \boldsymbol{e}_\tau \in R_q^l$, output the evaluation key as

$$\mathrm{evk}_{f \to f_{\mathrm{evk}}} := \{\boldsymbol{k}_\tau = h\boldsymbol{s}_\tau + 2\boldsymbol{e}_\tau + \mathrm{Pof2}(f) \in R_q^l\}_{\tau = 1, \ldots, l}.$$

(2) KeySwitch $(c, \boldsymbol{k}_\tau, q)$: Compute the ciphertext vector $\boldsymbol{c}' = \mathrm{BitD}(c) \in R_q^l$, and output $c_{\mathrm{evk}} = \boldsymbol{c}' \cdot \boldsymbol{k}_\tau = \langle \mathrm{BitD}(c), \boldsymbol{k}_\tau \rangle \in R_q$.

There are some useful conflusions in the following.

Lemma 1[26] Let $\Phi_n(x) = x^{n-1} + x^{n-2} + \cdots + 1$, and $R = \mathbf{Z}[x]/\Phi_n(x)$, where n is a prime. For any $a, b \in R$, it holds that

$$\|ab\|_\infty \leqslant 2(n-1)\|a\|_\infty \|b\|_\infty,$$
$$\|ab\| \leqslant 2\sqrt{n-1}\|a\|\|b\|.$$

According to Lemma 1, the following Lemma 2 can be drawn.

Lemma 2 Let $a, b \in R$ be sampled from a discrete Gaussian distribution with parameter $B\sqrt{2\pi}$ and bound B, under the worst-case conditions, the bound of $ab \bmod \Phi_n(x)$ is $\|ab\|_\infty \leqslant 2(n-1)B^2$, for convenience, mod $\Phi_n(x)$ is omitted. In particular, when the bound of b is 1, it holds that $\|ab\|_\infty \leqslant 2(n-1)B$.

Remark According to Lemma 2, if $a, b, c \in R$, we have $\|abc\|_\infty \leqslant 2(n-1)\|ab\|_\infty \|c\|_\infty \leqslant 2^2(n-1)^2 B^3$. So Lemma 2 yields the following corollary.

Corollary 1 Let λ be a security parameter, for the polynomial ring given by $R = \mathbf{Z}[x]/\Phi_n(x)$, $\Phi_n(x) = x^{n-1} + x^{n-2} + \cdots + 1$, where n is a prime, $\chi = \chi(\lambda)$ is a B-bound error distribution, and $q = q(\lambda) \in \mathbf{Z}$ is a prime integer, let $s_1, s_2, \ldots, s_k \leftarrow \chi$. Then, we have $\left\| \prod_{i=1}^k s_i \right\|_\infty \leqslant 2^{k-1}(n-1)^{(k-1)} B^k$.

3 Basic NTRU-Type Multi-Key Somewhat Homomorphic Encryption (MKSHE) over Prime Cyclotomic Rings

The NTRU key pairs consist of ring elements (h, f), such that $h = [2g/f]_q$, where g and f denote small elements sampled from a B-bounded distribution χ and f is invertible in R_q, respectively. Further recall that an NTRU ciphertext has the form of $\hat{c} := [m + h\hat{s} + 2\hat{e}]_q$ for small elements $\hat{s}$ and $\hat{e}$ sampled from χ, and m can be recovered by computing $[f\hat{c}]_q (\bmod 2)$.

The NTRU-type homomorphic encryption naturally supports the homomorphic evaluations between ciphertexts of different users (secret keys), which can be easily proven. Generally, it is assumed that there are four users A, B, C, and D, corresponding to the four public keys $(\mathrm{pk}_A, \mathrm{pk}_B, \mathrm{pk}_C, \mathrm{pk}_D)$ and four secret keys $(\mathrm{sk}_A, \mathrm{sk}_B, \mathrm{sk}_C, \mathrm{sk}_D)$, respectively. Plaintexts (m_A, m_B, m_C, m_D) can be encrypted as $(\hat{c}_A, \hat{c}_B, \hat{c}_C, \hat{c}_D)$, where $\hat{c}_i := h_i\hat{s} + 2\hat{e} + m_i \in R_q$, $i \in \{A, B, C, D\}$. Set $\{\mathrm{pk}_A, \mathrm{pk}_B, \mathrm{pk}_C\} \in K_1$, $\{\mathrm{pk}_B, \mathrm{pk}_C, \mathrm{pk}_D\} \in K_2$, so $K_1 \cap K_2 = \{\mathrm{pk}_B, \mathrm{pk}_C\}$, and $K = K_1 \cup K_2 = \{\mathrm{pk}_A, \mathrm{pk}_B, \mathrm{pk}_C, \mathrm{pk}_D\}$. It should be noted that since the method to process the cubic (or larger order) of a ciphertext product is similar to quadratic order, only the quadratic order of the ciphertext product is considered in this paper.

In particular, the two joint ciphertexts can be denote as $\hat{c}_1 = \hat{c}_A\hat{c}_B\hat{c}_C$ and $\hat{c}_2 = \hat{c}_B\hat{c}_C\hat{c}_D$, which can be decrypted to the joint plaintexts $m_1 = m_A m_B m_C$ and $m_2 = m_B m_C m_D$, respectively, by using the joint secret keys $F_{K_1} = f_A f_B f_C$ and $F_{K_2} = f_B f_C f_D$. That is $F_{K_1}\hat{c}_1 = F_{K_1}m_1 + v_1$ and $F_{K_2}\hat{c}_2 = F_{K_2}m_2 + v_2$. Similarly, to decrypt $\hat{c}_1\hat{c}_2 = \hat{c}_A\hat{c}_B^2\hat{c}_C^2\hat{c}_D$ we need to multiply $F_{K_1}F_{K_2} = f_A f_B^2 f_C^2 f_D$. Thus, the magnitude of the coefficients of $F_{K_1}F_{K_2}$ grows exponentially with the degree of the evaluated circuit. Namely, after L multiplications, the needed joint secret key will represent the product of L polynomials, and the magnitude of the coefficients in this product will increase exponentially with L. In order to solve these problems, the joint secret key $F_K = f_A f_B f_C f_D$, which has no quadratic items, is used to complete homomorphic decryption. Since $K_1, K_2 \subset (K_1 \cup K_2)$, we have

$$[F_K(\hat{c}_1 + \hat{c}_2)]_q = [F_K(m_A m_B m_C + m_B m_C m_D) + f_D v_1 + f_A v_2]_q.$$

However, since there are no f_B^2 and f_C^2 in F_K, the multiplication cannot be decrypted correctly. Thus, we have

$$[F_K\hat{c}_1\hat{c}_2]_q \neq \left[F_K m_A m_B^2 m_C^2 m_D + \mathrm{error}_{\mathrm{mult}}\right]_q,$$

where $\mathrm{error}_{\mathrm{mult}}$ represents the error of homomorphic multiplication decryption.

Therefore, the key-switching technique is used in the LTV12 scheme to re-linearize $\hat{c}_{\mathrm{mult}} = \hat{c}_1\hat{c}_2$, and switch $[F_K\hat{c}_1\hat{c}_2]_q$ to the decryption structure given by

$$[F_K(\hat{c}_1\hat{c}_2)]_q = \left[F_{K_1}F_{K_2}(\hat{c}_1\hat{c}_2) + \mathrm{error}_{\mathrm{mult}}\right]_q.$$

Although $\hat{c}_{\mathrm{mult}} = \hat{c}_1\hat{c}_2$ is decrypted by F_K, and the dependence of the coefficient' magnitude of the joint secret key on the circuit degree is eliminated, the error grows rapidly during the key-switching process. In the following, the error growing trend is explained in detail.

3.1 Scheme

The correctness and error of the NTRU-type multi-key homomorphic encryption scheme in the LTV12 scheme are analyzed. For prime cyclotomic rings, the correctness of the LTV12 scheme does not change, but the error is different compared to the LTV12 scheme on power-of-two cyclotomic rings. The basic LTV12 scheme over prime cyclotomic rings is described in this section, and a detailed analysis of the factors affecting the error growth during the homomorphic evaluations is provided.

Let λ be a security parameter, $q = q(\lambda) \in \mathbf{Z}$ is a prime integer, $\Phi_n(x) = x^{n-1} + x^{n-2} + \cdots + 1$ (n is a prime) is the sub-cyclotomic polynomial. For the polynomial ring $R = \mathbf{Z}[x]/\Phi_n(x)$ and $R_q = R/qR$, if the error distribution $\chi = \chi(\lambda)$ over R is B-bound, the $\chi^{\lceil \log q \rceil}$ is also a B-bound error distribution space.

As stated earlier in this section, we let K_1 and K_2 denote the two public-key sets containing N users. In the LTV12 scheme, the exponential dependence of the error on N is not eliminated, so it is assumed that there is an a-priori upper bound on N, that is $N \approx n^{\varepsilon}$, with constant $\varepsilon \in (0, 1)$. Without loss of generality, we assume $K_1 \cap K_2 = \{\mathrm{pk}_{i_1}, \ldots, \mathrm{pk}_{i_j}\}$, $K_1 \cup K_2 = \{\mathrm{pk}_1, \mathrm{pk}_2, \ldots, \mathrm{pk}_r\}$, where $j \in [0, N]$, $r \in [N, 2N]$.

The basic NTRU-type MKSHE over prime cyclotomic rings can be expressed as follows. This expression represents the basic MKSHE (called BC-MKSHE) scheme, whose security is based on the RLWE and DPSR assumptions over prime cyclotomic rings.

(1) BC-MKSHE. KeyGen(1^{λ}): Sample $f', g \leftarrow \chi$, and set $f = 2f' + 1$, so that $f \equiv 1 (\mathrm{mod}\ 2)$. If f is not invertible in R_q, resample $f' \leftarrow \chi$. Set $h = 2g/f \in R_q$, so pk $:= h \in R_q$, sk $:= f \in R$. For all $\tau \in \{1, \ldots, l\}$ (here $l = \lceil \log q \rceil$), sample $\boldsymbol{s}_\tau, \boldsymbol{e}_\tau \leftarrow \chi^l$, compute the evaluation key vector $\boldsymbol{k}_\tau = h\boldsymbol{s}_\tau + 2\boldsymbol{e}_\tau + \mathrm{Pof2}(f) \in R_q^l$.

Output: $(\mathrm{pk}, \mathrm{sk}, \mathrm{evk}) = (h, f, \boldsymbol{k}_\tau)$.

(2) BC-MKSHE. Enc(pk, m): Sample $\hat{s}, \hat{e} \leftarrow \chi$. Output the ciphertext: $\hat{c} := m + h\hat{s} + 2\hat{e} \in R_q$.

(3) BC-MKSHE. Dec($\mathrm{sk}_1, \mathrm{sk}_2, \ldots, \mathrm{sk}_N, \hat{c}$): Let $u := (\mathrm{sk}_1\mathrm{sk}_2 \cdots \mathrm{sk}_N)\hat{c} \in R_q$.

Output: $m' := u (\mathrm{mod}\ 2)$.

(4) BC-MKSHE. KeySwitch($\tilde{c}, \boldsymbol{k}_\tau, q$): Given the ciphertext $\tilde{c}$ and the evaluation key $\boldsymbol{k}_\tau$, and output $[\langle \mathrm{BitD}(\tilde{c}), \boldsymbol{k}_\tau \rangle]_q$.

(5) BC-MKSHE. Eval. Add($\hat{c}_1, \hat{c}_2$): Given two

ciphertexts $\hat{c}_1$ and $\hat{c}_2$ with the corresponding public-key sets K_1 and K_2, output the ciphertext $\hat{c}_{\text{add}} = [\hat{c}_1 + \hat{c}_2]_q \in R_q$.

(6) BC-MKSHE. Eval. Mult($\hat{c}_1, \hat{c}_2$, KeySwitch): Given two ciphertexts $\hat{c}_1$ and $\hat{c}_2$ with the corresponding public-key sets K_1 and K_2, let $\tilde{c}_0 = \hat{c}_1\hat{c}_2$. For $j \in [0, N]$,

(a) If j=0, output $\hat{c}_{\text{mult}} = \tilde{c}_0 \in R_q$.

(b) If $j \neq 0$, for $t \in [1, j]$, compute $\tilde{c}_t = \text{KeySwitch}(\tilde{c}_{t-1}, \boldsymbol{k}_{t,\tau}, q) \in R_q$.

Let $\hat{c}_{\text{mult}} = \tilde{c}_j$ at the end of the iteration.

3.2 Correctness analysis

Multi-key homomorphism: Let F_{K_1} and F_{K_2} be the joint decryption keys for ciphertext $\hat{c}_1$ and $\hat{c}_2$, respectively. Also, let $F_t = F_{t-1}(f_t)^{-1}$, $t \in [1, j]$, where $F_0 = F_{K_1}F_{K_2}$. According to the LTV12 scheme, the addition and multiplication on ciphertexts can be decrypted using the product of the users' secret keys in set $K = K_1 \cup K_2$. For given two ciphertexts $\hat{c}_1$ and $\hat{c}_2$ with the corresponding public-key sets K_1 and K_2, we have $F_{K_1}\hat{c}_1 = F_{K_1}m_1 + v_1$, $F_{K_2}\hat{c}_2 = F_{K_2}m_2 + v_2$, and $\|F_{K_1}\hat{c}_1\|_\infty = \|F_{K_2}\hat{c}_2\|_\infty < \psi$, where v_1 and v_2 denote the error. Then, in the case of addition, we have

$$F_K\hat{c}_{\text{add}} = F_K(m_1+m_2) + F_{K-K_1}v_1 + F_{K-K_2}v_2 + v_1v_2.$$

Therefore, $\hat{c}_{\text{add}}$ decrypts correctly. The multiplication case is more complex, and in that case, we have

$$F_t\tilde{c}_t = F_tf_t^{-1}(\text{BitD}(\tilde{c}_{t-1}) \cdot 2g_t\boldsymbol{s}_\tau) + 2F_t(\text{BitD}(\tilde{c}_{t-1}) \cdot \boldsymbol{e}_\tau) + F_{t-1}\tilde{c}_{t-1}.$$

For all $t \in [1, j]$, by using the key-switching technique, we can get the decryption structure in the following form:

$$F_K\hat{c}_{\text{mult}} = F_j\tilde{c}_j = \text{error}_{\text{mult}} + F_{K_1}F_{K_2}\tilde{c}_0 \quad (1)$$

where $F_K = F_j = f_1f_2\cdots f_r$ has no quadratic item of f_t. Equation (1) represents the inherent decryption structure of the NTRU-type homomorphic multiplication. By rounding of $[F_{K_1}F_{K_2}\tilde{c}_0]_q$, we get $F_{K_1}F_{K_2}m_1m_2$, and thus $\hat{c}_{\text{mult}}$ can be decrypted correctly by F_K. This inherent decryption structure that is deduced by using the key-switching technique is a guarantee that we can successfully decrypt $\hat{c}_{\text{mult}}$ by F_K.

Error analysis: Assume that there is no intersection between the public keys of the N users in K_1 or K_2. For instance, consider the worst case of the ciphertext $\hat{c}_1$ encrypted by the public-key set K_1, and let $\hat{c}_1$ be the multiplication of all users' fresh ciphertexts, i.e., $\hat{c}_1 = \hat{c}_1^1\hat{c}_2^1\cdots\hat{c}_N^1$ denotes the ciphertext of $m_1 = m_1^1m_2^1\cdots m_N^1$, where $\hat{c}_i^1$ denotes a fresh ciphertext, so $\hat{c}_i^1 f_i = m_i^1 f_i + v_i^1$.

By applying Lemma 2 and Corollary 1, we easily get $\|\hat{c}_i^1 f_i\|_\infty < (2B+1) + 3(n-1)(2B+1)^2$. Let $(2B+1) + 3(n-1)(2B+1)^2 = \psi_0$, then in the above-mentioned worst case, $\hat{c}_1F_{K_1} = (\hat{c}_1^1f_1)(\hat{c}_2^1f_2)\cdots(\hat{c}_N^1f_N)$. So ψ can be bounded by

$$\psi \leqslant 2^{N-1}(n-1)^{N-1}(\psi_0)^N < 2^{3N-1}(n-1)^{2N-1}(2B+1)^{2N}.$$

Necessarily, let $\psi < q/2$.

Based on Corollary 1, we have $\|F_K\|_\infty \leqslant 2^{r-1}(n-1)^{r-1}(2B+1)^r$, and for convenience, we set $E_r \leqslant 2^{r-1}(n-1)^{r-1}(2B+1)^r$.

Since the error generated by homomorphic addition is much smaller than the multiplication, we analyze only the error generated by the homomorphic multiplication. Thus, it can be written

$$\|F_j\tilde{c}_j\|_\infty < 3l(n-1)E_{r+1} + \|F_{j-1}\tilde{c}_{j-1}\|_\infty \quad (2)$$

Then, we obtain

$$\|F_{j-1}\tilde{c}_{j-1}\|_\infty < 3l(n-1)E_{r+2} + \|F_{j-2}\tilde{c}_{j-2}\|_\infty.$$

Thus, the final error is bounded in the following:

$$\|F_j\tilde{c}_j\|_\infty < 3l(n-1)\sum_{t=1}^{j}E_{r+t} + 2(n-1)\psi^2 = 3l(n-1)E_{r+j}\left(1+\sum_{t=1}^{j-1}E_{-t}\right) + 2(n-1)\psi^2 < 6l(n-1)E_{2N} + 2(n-1)\psi^2 \quad (3)$$

Based on Eq. (3), it can be found that there are two main factors affecting the error growth, which are the number of union secret keys N and the length of vector $\text{BitD}(\tilde{c}_j)$. Hence, these two factors can be used to control the error growth by applying the two following methods:

(1) Exponential dependence of the error on N can be reduced by eliminating the key-switching operations;

(2) Error magnitude can be decreased by reducing the length of a ciphertext vector.

4 Modified Multi-Key Homomorphic Multiplication Decryption Structure

The key-switching technique causes fast error growth. The main way to decrease the error is to eliminate the key-switching in the BC-MKSHE. However, to ensure that the scheme works correctly, the decryption structure of the homomorphic multiplication has to be modified. In Refs. [31] and [33], the decryption structure of a single-key homomorphic encryption scheme was studied. Chen[33] extended the ciphertext to the vector form, and performed the homomorphic decryption in the vector space. The main disadvantage of this approach

is that the increase in the ciphertext dimension makes the homomorphic evaluations more complex. However, in our multi-key homomorphic encryption scheme, the homomorphic decryption structure is improved by expanding the ciphertext dimension, and the Low Bit Discarded (LBD) method is used to control the ciphertext space size.

4.1 LBD technique

In Ref. [34], the efficiency of the fully homomorphic encryption scheme was enhanced by discarding the lower bits. In this section, the LBD based on the functions BitDecomp( ) and Powersof2( ) is presented.

LBD: By discarding elements with small coefficients (i.e., low bits) of Powersof2( ) vector, a lower-dimension vector that has no influence on the final decryption is obtained. For instance, for a given ciphertext c_i and a secret key f_i, the inner product of $\mathrm{BitD}(c_i)$ and $\mathrm{Pof}\,2(f_i)$ can be obtained,

$$\begin{aligned}&\mathrm{BitD}(c_i)\cdot \mathrm{Pof2}(f_i)=\\&\quad(c_{i,1},c_{i,2},\ldots,c_{i,l})(2^0 f_i,2^1 f_i,\ldots,2^{l-1}f_i)=\\&\quad 2^0 c_{i,1}f_i+c_{i,2}2^1 f_i+\cdots+c_{i,l}2^{l-1}f_i\end{aligned}\tag{4}$$

where $c_{i,\tau}\in R_2, l=\lceil\log q\rceil, \tau\in[1,l]$. Compared to $2^{l-1}c_{i,l}f_i$, the value of $2^{d-1}c_{i,d}f_i$ $(d\ll l)$ is small. Thus, when l is large, discarding the terms of $c_{i,1}f_i, 2c_{i,2}f_i, 2^2c_{i,3}f_i,\ldots,2^{d-1}c_{i,d}f_i$ has a little effect on the overall value of Eq. (4). Further, the LBD function can be defined as $\mathrm{LBD}_{d_1\to d_2}(\mathrm{Pof2}(f_i))$, which means that the columns from d_1 to d_2 (we denote as $(d_1\to d_2)$) of vector $\mathrm{Pof2}(f_i)$ are discarded. According to Eq. (4), assume that the $(1\to d)$ columns of $\mathrm{Pof2}(f_i)$ are discarded, to ensure the correctness of the mathematical operation, we should discard the $(1\to d)$ columns of $\mathrm{BitD}(c_i)$. Therefore, we get

$$\begin{aligned}&\mathrm{LBD}_{1\to d}(\mathrm{BitD}(c_i))\cdot\mathrm{LBD}_{1\to d}(\mathrm{Pof2}(f_i))=\\&\quad(c_{i,d+1},\ldots,c_{i,l})(2^d f_i,\ldots,2^{l-1}f_i)=\\&\quad c_{i,d+1}2^d f_i+\cdots+c_{i,l}2^{l-1}f_i\end{aligned}\tag{5}$$

According to Eq. (5), the dimensions of vectors $\mathrm{BitD}(c_i)$ and $\mathrm{Pof}\,2(f_i)$ are reduced after the LBD, while the operation efficiency is improved.

If we set $l=\lceil\log q\rceil$ and $d=\lceil\log q'\rceil$, the LBD technique can be described,

$$\begin{aligned}&\mathrm{LBD}_{1\to d}(\mathrm{BitD}(c_i))\cdot\mathrm{LBD}_{1\to d}(\mathrm{Pof2}(f_i))=\\&\quad\sum_{\tau=1}^{l}c_{i,\tau}2^{\tau-1}f_i-\sum_{\tau=1}^{d}c_{i,\tau}2^{\tau-1}f_i=\\&\quad(\mathrm{BitD}(c_i)\cdot\mathrm{Pof2}(f_i))_{l=\lceil\log q\rceil}-\\&\quad(\mathrm{BitD}(c_i)\cdot\mathrm{Pof2}(f_i))_{d=\lceil\log q'\rceil}\end{aligned}\tag{6}$$

According to Eq. (6), the LBD is based only on the functions BitDecomp() and Powersof 2().

4.2 Modified method

In this section, the LBD technique is employed to improve the homomorphic decryption process.

Method 1: In the BC-MKSHE scheme, the LBD technique can be used to discard redundant bits of the evaluation key vector to simply the key-switching operations. So we have Method 1 in the following.

Step 1 (Discard low bits of the evaluation key vector): Let β and d be positive constants, and $l=\lceil\log q\rceil$. Perform the LBD functions to obtain the evaluation key. Output: $\tilde{\boldsymbol{k}}_\tau=\mathrm{LBD}_{\beta\to d+\beta-1}(\boldsymbol{k}_\tau)\in R_q^{l-d}$.

Step 2 (Simplify KeySwitch function): Compute the ciphertext vector $\mathrm{LBD}_{\beta\to d+\beta-1}(\mathrm{BitD}(\tilde{c}))\in R_q^{l-d}$. Output is as follows:

$$\begin{aligned}&\mathrm{KeySwitch}_{\mathrm{LBD}}(\tilde{c},\tilde{\boldsymbol{k}}_\tau,q)=\\&\quad[\langle\mathrm{LBD}_{\beta\to d+\beta-1}(\mathrm{BitD}(\tilde{c})),\mathrm{LBD}_{\beta\to d+\beta-1}(\boldsymbol{k}_\tau)\rangle]_q.\end{aligned}$$

Step 3 (Compute the homomorphic multiplication of ciphertexts): Given two ciphertexts $\hat{c}_1$ and $\hat{c}_2$ with the corresponding public-key sets K_1 and K_2, let $\tilde{c}_0=\hat{c}_1\hat{c}_2$. For $j\in[0,N]$,

$$\tilde{\tilde{c}}_t=\mathrm{KeySwitch}_{\mathrm{LBD}}(\tilde{c}_{t-1},\tilde{\boldsymbol{k}}_{t,\tau},q)\in R_q.$$

Set $c_{\mathrm{mult}}=\tilde{\tilde{c}}_j$ at the end of the iteration.

Correctness verification of Method 1: Sample $\tilde{\boldsymbol{s}}_\tau,\tilde{\boldsymbol{e}}_\tau\leftarrow\chi^{l-d}$, we have

$$\begin{aligned}F_t\tilde{\tilde{c}}_t=&F_t f_t^{-1}(\mathrm{LBD}_{\beta\to d+\beta-1}(\mathrm{BitD}(\tilde{c}_{t-1}))\cdot 2g_t\tilde{\boldsymbol{s}}_\tau)+\\&2F_t(\mathrm{LBD}_{\beta\to d+\beta-1}(\mathrm{BitD}(\tilde{c}_{t-1}))\cdot\tilde{\boldsymbol{e}}_\tau)-\\&F_{t-1}\left(\sum_{\varsigma=\beta+1}^{d+\beta-2}2^{\varsigma-1}\tilde{c}_{t-1,\varsigma}\right)+F_{t-1}\tilde{c}_{t-1}\end{aligned}\tag{7}$$

where $\tilde{c}_{t-1,\varsigma}\in R_2$ and ς is a constant variable. According to Eq. (6) and since $F_{j-1}(\mathrm{mod}\ 2)\equiv 1$, we set $\beta>1$ to ensure that $F_{t-1}(\sum_{\varsigma=\beta+1}^{d+\beta-2}2^{\varsigma-1}\tilde{c}_{t-1,\varsigma})$ is an even element. Thus, if $\beta>1$, we have $F_t\tilde{\tilde{c}}_t\cdot(\mathrm{mod}\ 2)=F_{t-1}\tilde{c}_{t-1}(\mathrm{mod}\ 2)$. Further, according to Eq. (7), the error magnitude is given in the following:

$$\begin{aligned}\|F_j\tilde{\tilde{c}}_j\|_\infty=&\|F_j f_j^{-1}(\mathrm{LBD}_{\beta\to d+\beta-1}(\mathrm{BitD}(\tilde{c}_{j-1}))\cdot\\&2g_j\tilde{\boldsymbol{s}}_\tau)\|_\infty+\|2F_j(\mathrm{LBD}_{\beta\to d+\beta-1}(\mathrm{BitD}(\tilde{c}_{j-1}))\cdot\\&\tilde{\boldsymbol{e}}_\tau)\|_\infty+\left\|F_{j-1}\left(\sum_{\varsigma=\beta+1}^{d+\beta-2}2^{\varsigma-1}\tilde{c}_{j-1,\varsigma}\right)\right\|_\infty+\\&\|F_{j-1}\tilde{c}_{j-1}\|_\infty<(3(l-d)+(2^{d+2}-4))\times\\&(n-1)E_{r+1}+\|F_{j-1}\tilde{c}_{j-1}\|_\infty\end{aligned}\tag{8}$$

For any $a,b\in R_q$, it holds that $\|a-b\|_\infty\leqslant\|a\|_\infty+\|b\|_\infty$. Since $F_t=F_{t-1}(f_t)^{-1}$, the magnitude of $\|F_j f_j^{-1}g_j\tilde{\boldsymbol{s}}_\tau\|_\infty$, $\|F_j\tilde{\boldsymbol{e}}_\tau\|_\infty$, and $\|F_{j-1}\tilde{c}_{j-1,\varsigma}\|_\infty$ in Eq.

(8) are the same. Compared to Eq. (2), it can be found that when $d > 1$, $\|F_j\tilde{\tilde{c}}_j\|_\infty > \|F_j\tilde{c}_j\|_\infty$, which means that low bits are discarded, the error increases. Moreover, the value of $\|F_j\tilde{\tilde{c}}_j\|_\infty$ increases with the use frequency of key-switching. Thus, it seems that this method does not provide satisfactory results.

Method 2: LBD&DEC is proposed to modify the decryption structure of the NTRU-type MKSHE. First, the LBD technique is employed to discard redundant elements in the plaintext vector, and then the plaintext vector is encrypted to expand the ciphertext dimension. Finally, the decryption structure in the vector space is improved. Referring to Method 1, some bits of the plaintext vector from the second column are discarded to ensure the correctness of the decryption. Accordingly, the modified decryption structure can be improved by the following steps:

Step 1 (Discard the plaintext vector): Let $l = \lceil \log q \rceil$, d is a positive constant. Compute the plaintext vector $\hat{\boldsymbol{m}} = \mathrm{LBD}_{2\to d+1}(\mathrm{Pof}\,2(m))$ $(m \in \{0,1\})$.

Output: $\hat{\boldsymbol{m}} = (m, 2^{d+1}m, \cdots, 2^{l-1}m) \in R_q^{l-d}$.

Step 2 (Ciphertext expansion): Let pk $:= h \in R_q$, sk $:= f \in R$. Sample $\boldsymbol{s}, \boldsymbol{e} \leftarrow \chi^l$. Use public key pk to encrypt the plaintext vector $\hat{\boldsymbol{m}} = (m, 2^{d+1}m, \ldots, 2^{l-1}m)$.

Output: $\boldsymbol{c} := \hat{\boldsymbol{m}} + h\boldsymbol{s} + 2\boldsymbol{e} \in R_q^{l-d}$.

Step 3 (Set the decryption structure): For given two ciphertext vectors $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ with the corresponding public-key sets K_1 and K_2, compute the matrix $\boldsymbol{C}$,

$$\boldsymbol{C} = \mathrm{LBD}_{2\to d+1}(\mathrm{BitD}(\boldsymbol{c}_1^{\mathrm{T}})) \in R_q^{(l-d)\times(l-d)}.$$

Output: $\boldsymbol{c}_{\mathrm{mult}} = \boldsymbol{C}\cdot\boldsymbol{c}_2^{\mathrm{T}} \in R_q^{l-d}$.

Step 4 (Select decryption element): Select the first element $c_{\mathrm{mult},1}$ from the ciphertext vector $\boldsymbol{c}_{\mathrm{mult}}$. Here, $F_K = F_j = f_1 f_2 \cdots f_r$ and $K = K_1 \cup K_2$.

Output $\mathrm{Dec}(F_K, c_{\mathrm{mult},1})(\mathrm{mod}\ 2)$.

We modify $\hat{c} \in R_q$ to obtain $\boldsymbol{c} := \mathrm{Pof2}(m) + h\hat{\boldsymbol{s}} + 2\hat{\boldsymbol{e}} \in R_q^l$ instead of $\boldsymbol{c} := \mathrm{Pof2}(m + h\hat{s} + 2\hat{e})$, so that the error generated by the term of $\mathrm{Pof2}(h\hat{s} + 2\hat{e})$ can be removed. In Section 4.3, the advantages of this change will be introduced when calculating the error magnitude.

Correctness verification of Method 2: According to Step 2, set $\alpha = \{1, 2\}$, we get

$$\boldsymbol{c}_\alpha^{\mathrm{T}} := \begin{bmatrix} 2^0 m_\alpha + hs_{\alpha,1} + 2e_{\alpha,1} \\ 2^{d+1}m_\alpha + hs_{\alpha,d+2} + 2e_{\alpha,d+2} \\ \vdots \\ 2^{l-1}m_\alpha + hs_{\alpha,l} + 2e_{\alpha,l} \end{bmatrix} \in R_q^{l-d}.$$

Thus, we have $\mathrm{BitD}(\boldsymbol{c}_1^{\mathrm{T}}) \in R_2^{(l-d)\times l}$. To keep the correctness of $\mathrm{BitD}(\boldsymbol{c}_1^{\mathrm{T}})\cdot\boldsymbol{c}_2^{\mathrm{T}}$, we perform Step 3. After discarding the $(2 \to d+1)$ columns of matrix $\mathrm{BitD}(\boldsymbol{c}_1^{\mathrm{T}})$, an $(l-d)\times(l-d)$ matrix is obtained,

$$\boldsymbol{C} = \mathrm{LBD}_{2\to d+1}(\mathrm{BitD}(\boldsymbol{c}_1^{\mathrm{T}})) = \begin{bmatrix} (c_{1,1})_1 & (c_{1,1})_{d+2} & \cdots & (c_{1,1})_l \\ (c_{1,d+2})_1 & (c_{1,d+2})_{d+2} & \cdots & (c_{1,d+2})_l \\ \vdots & \vdots & \ddots & \vdots \\ (c_{1,l})_1 & (c_{1,l})_{d+2} & \cdots & (c_{1,l})_l \end{bmatrix}.$$

So, according to Step 4, we use the joint secret key F_K to decrypt $\boldsymbol{c}_{\mathrm{mult}}$.

$$\boldsymbol{c}_{\mathrm{mult}}\cdot F_K(\mathrm{mod}\ 2) = \boldsymbol{C}\cdot\boldsymbol{c}_2^{\mathrm{T}}\cdot F_K(\mathrm{mod}\ 2) = \mathrm{LBD}_{2\to d+1}(\mathrm{BitD}(\boldsymbol{c}_1^{\mathrm{T}}))\cdot\boldsymbol{c}_2^{\mathrm{T}}\cdot F_K(\mathrm{mod}2) =$$

$$\begin{bmatrix} (c_{1,1})_1 & (c_{1,1})_{d+2} & \cdots & (c_{1,1})_l \\ (c_{1,d+2})_1 & (c_{1,d+2})_{d+2} & \cdots & (c_{1,d+2})_l \\ \vdots & \vdots & \ddots & \vdots \\ (c_{1,l})_1 & (c_{1,l})_{d+2} & \cdots & (c_{1,l})_l \end{bmatrix} \cdot \begin{bmatrix} 2^0 m_2 + hs_{2,1} + 2e_{2,1} \\ 2^{d+1}m_2 + hs_{2,d+2} + 2e_{2,d+2} \\ \vdots \\ 2^{l-1}m_2 + hs_{2,l} + 2e_{2,l} \end{bmatrix} \cdot F_K \overset{e'_{2,\varsigma}=hs_{2,\varsigma}+2e_{2,\varsigma}}{=}$$

$$\begin{bmatrix} (c_{1,1})_1 & (c_{1,1})_{d+2} & \cdots & (c_{1,1})_l \\ (c_{1,d+2})_1 & (c_{1,d+2})_{d+2} & \cdots & (c_{1,d+2})_l \\ \vdots & \vdots & \ddots & \vdots \\ (c_{1,l})_1 & (c_{1,l})_{d+2} & \cdots & (c_{1,l})_l \end{bmatrix} \cdot \begin{bmatrix} 2^0 m_2 F_K + F_K e'_{2,1} \\ 2^{d+1}m_2 F_K + F_K e'_{2,d+2} \\ \vdots \\ 2^{l-1}m_2 F_K + F_K e'_{2,l} \end{bmatrix} =$$

$$\begin{bmatrix} (c_{1,1})_1(2^0 m_2 F_K + F_K e'_{2,1}) + \sum_{\varsigma=d+2}^{l} (c_{1,1})_\varsigma (2^{\varsigma-1}m_2 F_K + F_K e'_{2,\varsigma}) \\ (c_{1,d+2})_1(2^0 m_2 F_K + F_K e'_{2,1}) + \sum_{\varsigma=d+2}^{l} (c_{1,d+2})_\varsigma (2^{\varsigma-1}m_2 F_K + F_K e'_{2,\varsigma}) \\ \vdots \\ (c_{1,l})_1(2^0 m_2 F_K + F_K e'_{2,1}) + \sum_{\varsigma=d+2}^{l} (c_{1,l})_\varsigma (2^{\varsigma-1}m_2 F_K + F_K e'_{2,\varsigma}) \end{bmatrix}.$$

We set $\boldsymbol{c}_1 F_{K_1} = \hat{\boldsymbol{m}}_1 F_{K_1} + \boldsymbol{v}'_1$ and $\boldsymbol{c}_2 F_{K_2} = \hat{\boldsymbol{m}}_2 F_{K_2} + \boldsymbol{v}'_2$, let $\|\boldsymbol{v}'_1\|_\infty = \|\boldsymbol{v}'_2\|_\infty < \eta$. Thus, the first element is selected as decryption,

$$
\begin{aligned}
&c_{\text{mult},1} F_K = (c_{1,1})_1 (2^0 m_2 F_K + F_K e'_{2,1}) + \\
&\sum_{\varsigma=d+2}^{l} (c_{1,1})_\varsigma (2^{\varsigma-1} m_2 F_K + F_K e'_{2,\varsigma}) = \\
&\sum_{\varsigma=1}^{l} (c_{1,1})_\varsigma (2^{\varsigma-1} m_2 F_K + F_K e'_{2,\varsigma}) - \\
&\sum_{\varsigma=2}^{d+1} (c_{1,1})_\varsigma (2^{\varsigma-1} m_2 F_K + F_K e'_{2,\varsigma}) = \\
&\sum_{\varsigma=1}^{l} (2^{\varsigma-1} c_{1,1})_\varsigma m_2 F_K + \sum_{\varsigma=1}^{l} (c_{1,1})_\varsigma F_K e'_{2,\varsigma} - \\
&\left(\sum_{\varsigma=2}^{d+1} (c_{1,1})_\varsigma F_K e'_{2,\varsigma} + \sum_{\varsigma=2}^{d+1} (2^{\varsigma-1} c_{1,1})_\varsigma m_2 F_K \right) = \\
&m_1 m_2 F_K + \left(m_2 F_{K-K_1} v'_{1,1} + \sum_{\varsigma=1}^{l} (c_{1,1})_\varsigma F_{K-K_2} v'_{2,1} - \right. \\
&\left. \sum_{\varsigma=2}^{d+1} (c_{1,1})_\varsigma F_{K-K_2} v'_{2,1} - m_2 \sum_{\varsigma=2}^{d+1} (2^{\varsigma-1} c_{1,1})_\varsigma F_K \right) \quad (9)
\end{aligned}
$$

where $\|v'_{1,1}\|_\infty = \|\boldsymbol{v}'_1\|_\infty$ and $\|v'_{2,1}\|_\infty = \|\boldsymbol{v}'_2\|_\infty$. So we can get $c_{\text{mult},1} F_K (\text{mod } 2) = m_1 m_2$. The error magnitude is bounded by $\|c_{\text{mult},1} F_K\|_\infty \leqslant q/2$.

4.3 Parameters analysis

As already explained, by using the LBD&DEC technique, the original, inherent decryption structure of the NTRU-type MKHE can be modified.

Theorem 1 Set LBD constant d, $\boldsymbol{c}_\alpha F_{K_\alpha} = \hat{\boldsymbol{m}}_\alpha F_{K_\alpha} + \boldsymbol{v}'_\alpha$, and $\|\boldsymbol{v}'_\alpha\|_\infty < \eta$. Let $c_{\text{mult},1}$ be the first column of the ciphertext vector $\boldsymbol{c}_{\text{mult}} = \boldsymbol{C} \cdot \boldsymbol{c}_2^{\text{T}} \in R_q^{l-d}$, and $\boldsymbol{C} = \text{LBD}_{2 \to d+1}(\text{BitD}(\boldsymbol{c}_1^{\text{T}})) \in R_q^{(l-d)\times(l-d)}$. Then, we have

$$c_{\text{mult},1} F_K = m_1 m_2 F_K + v_{\text{mult},1}$$

and

$$
\begin{aligned}
\|v_{\text{mult},1}\|_\infty \leqslant\ & (2(n-1)(l-d)+1) \times \\
& \left(\frac{3}{2} E_{r+2} + 3N(n-1)(l-d) E_{r+1} \right) + \\
& (1 + N(2^{d+2} - 4)(n-1)\,(2(n-1)(l-d)+1)) \times \\
& (2^{d+2} - 4)(n-1) E_r,
\end{aligned}
$$

where $E_r = 2^{r-1}(n-1)^{r-1}(2B+1)^r$.

Proof According to Eq. (9), the error of $c_{\text{mult},1} F_K$ can be bounded as follows:

$$
\begin{aligned}
\|c_{\text{mult},1} F_K\|_\infty = & \left\| m_1 m_2 F_K + \right. \\
& \left(m_2 F_{K-K_1} v'_{1,1} + \sum_{\varsigma=1}^{l} (c_{1,1})_\varsigma F_{K-K_2} v'_{2,1} - \right. \\
& \left. \left. \sum_{\varsigma=2}^{d+1} (c_{1,1})_\varsigma F_{K-K_2} v'_{2,1} - m_2 \sum_{\varsigma=2}^{d+1} (2^{\varsigma-1} c_{1,1})_\varsigma F_K \right) \right\|_\infty \leqslant \\
& \|m_1 m_2 F_K\|_\infty + \|m_2 F_{K-K_1} v'_{1,1}\|_\infty + \\
& \left\| \left(\sum_{\varsigma=1}^{l} (c_{1,1})_\varsigma - \sum\nolimits_{\varsigma=2}^{d+1} (c_{1,1})_\varsigma \right) F_{K-K_2} v'_{1,1} \right\|_\infty + \\
& \left\| m_2 \sum_{\varsigma=2}^{d+1} (2^{\varsigma-1} c_{1,1})_\varsigma F_K \right\|_\infty \leqslant E_r + 2(n-1) E_{r-N} \eta + \\
& (4(n-1)^2 (l-d)) E_{r-N} \eta + (2^{d+2} - 4)(n-1) E_r \quad (10)
\end{aligned}
$$

Thus, the starting error can be easily obtained as $\eta_0 < 3(n-1)(2B+1)^2$. Then, the magnitude of η is calculated. For N users having a set of public-keys K_1, the worst case is that $\boldsymbol{c}_1$ is the product of all the users' fresh ciphertexts. Specifically, for $N = 2$, we have $\{\boldsymbol{c}_1^1, f_1\}$ and $\{\boldsymbol{c}_2^1, f_2\}$, where $\boldsymbol{c}_1^1$ and $\boldsymbol{c}_2^1$ denote the fresh ciphertext vectors. The magnitude of η is bounded,

$$
\begin{aligned}
\eta_{N=2} < & (2(n-1)(2B+1))\,(2(n-1)(l-d)+1)\,\eta_0 + \\
& (2^{d+2} - 4)(n-1) E_2.
\end{aligned}
$$

Further, as $N=3$, set $\{\boldsymbol{c}'_1 = \boldsymbol{c}_1^1 \boldsymbol{c}_2^1,\ f'_1 = f_1 f_2\}$ and $\{\boldsymbol{c}_3^1, f_3\}$. So the magnitude of η is bounded,

$$
\begin{aligned}
\eta_{N=3} < & (2(n-1)(2B+1))\,\eta_{N=2} + \\
& (4(n-1)^2 (l-d)) E_2 \eta_0 + \\
& (2^{d+2} - 4)(n-1) E_3.
\end{aligned}
$$

Accordingly, for N users, set $\{\boldsymbol{c}'_1 = \boldsymbol{c}_1^1 \ldots \boldsymbol{c}_{N-1}^1,\ f'_1 = f_1 \ldots f_{N-1}\}$ and $\{\boldsymbol{c}_N^1, f_N\}$. So, the magnitude of η is bounded,

$$
\begin{aligned}
\eta \leqslant & (2(n-1)(2B+1))\eta_{N-1} + \\
& (4(n-1)^2 (l-d)) E_{N-1} \eta_0 + \\
& (2^{d+2} - 4)(n-1) E_N.
\end{aligned}
$$

This yields to the following relationship:

$$
\begin{aligned}
\eta \leqslant & (2(n-1)(2B+1))^N \eta_0 + \\
& (3N(n-1)(l-d) E_{N+1}) + \\
& N(2^{d+2} - 4)(n-1) E_N = \\
& \frac{3}{2} E_{N+2} + 3N(n-1)(l-d) E_{N+1} + \\
& N(2^{d+2} - 4)(n-1) E_N \quad (11)
\end{aligned}
$$

By combining Eqs. (10) and (11), we get

$$\begin{aligned}\|c_{\text{mult},1}F_K\|_\infty \leqslant\ & E_r+(2(n-1)(l-d)+1)\times\\ & \left(\frac{3}{2}E_{r+2}+3N(n-1)(l-d)E_{r+1}\right)+\\ & (1+N(2^{d+2}-4)(n-1)(2(n-1)(l-d)+1))\times\\ & (2^{d+2}-4)(n-1)E_r.\end{aligned}$$

So, the decryption structure is obtained $c_{\text{mult},1}F_r = m_1m_2F_r + v_{\text{mult},1}$. Since

$$\|c_{\text{mult},1}F_r\|_\infty = \|m_1m_2F_r\|_\infty + \|v_{\text{mult},1}\|_\infty.$$

Further, we obtain

$$\begin{aligned}\|v_{\text{mult},1}\|_\infty \leqslant\ & (2(n-1)(l-d)+1)\times\\ & \left(\frac{3}{2}E_{r+2}+3N(n-1)(l-d)E_{r+1}\right)+\\ & (1+N(2^{d+2}-4)(n-1)\,(2(n-1)(l-d)+1))\times\\ & (2^{d+2}-4)(n-1)E_r.\end{aligned}$$

■

According to Theorem 1, it can be found that when $d=0$, $\|v_{\text{mult},1}\|_\infty \leqslant (2(n-1)l+1)\left(\frac{3}{2}E_{r+2}+3N(n-1)l\times E_{r+1}\right)$. This denotes the error generated only by the ciphertext dimension extension technique.

Theorem 2 Set LBD constant d. The LBD&DEC technique can decrease both the ciphertext dimension and error magnitude, and d satisfies the following relationship:

$$d = \{\max(d)|(2^d-1)/d \leqslant 3(n-1)(2B+1)/2,\ d>0\}.$$

Proof Take the homomorphic multiplication as an example, the LBD technique is to decrease the ciphertext dimension and error magnitude. If the LBD technique is not used in the ciphertext vector, the decryption structure is assumed as $\boldsymbol{c}^*_{\text{mult}}\cdot F_K = \text{BitD}((\boldsymbol{c}^*_1)^{\text{T}})\cdot(\boldsymbol{c}^*_2)^{\text{T}}\cdot F_K$. The decryption can be completed by using the first column of $\boldsymbol{c}^*_{\text{mult}}\cdot F_K$,

$$\begin{aligned}c^*_{\text{mult},1}F_K =\ & m_1m_2F_K + m_2F_{K-K_1}v^*_{1,1}+\\ & \sum_{\varsigma=1}^{l}(c^*_{1,1})_\varsigma F_{K-K_2}v^*_{2,1}.\end{aligned}$$

Assume the magnitude of errors $v^*_{1,1}$ and $v^*_{2,1}$ is η'. According to Eq. (10),

$$\|c^*_{\text{mult},1}F_K\|_\infty \leqslant E_r+(2(n-1)l+1)\,2(n-1)E_{r-N}\eta'.$$

Obviously, the LBD can reduce the ciphertext dimension from l to $(l-d)$, but we want to decrease the error magnitude at the same time. Note that the starting error is $\eta_0 < 3(n-1)(2B+1)^2$. According to Theorem 1, when $d=0$, we can obtain

$$\eta' \leqslant 3E_{N+2}/2 + 3N(n-1)lE_{N+1}.$$

Compared to Eq. (11), there is a constant $d>0$ that makes $\eta' > \eta$, which is given by

$$\begin{aligned}& 3N(n-1)dE_{N+1} > N(2^{d+2}-4)(n-1)E_N\\ & \Rightarrow 3(n-1)(2B+1) > \frac{(2^{d+1}-2)}{d}\end{aligned}\tag{12}$$

It can be easily found that $(2^d-1)/d$ is incremental of d ($d>0$). So, d has an upper bound that makes η be the closest to η'. We let d_1 be the upper bound of d, if $d=d_1$, then $\eta \approx \eta'$. Therefore, we need to verify the correctness of the relationship $\|c^*_{\text{mult},1}F_K\|_\infty \geqslant \|c_{\text{mult},1}F_K\|_\infty$. So, we have

$$\begin{aligned}& \|c^*_{\text{mult},1}F_K\|_\infty > \|c_{\text{mult},1}F_K\|_\infty\\ & \Rightarrow d_1 2(n-1)E_{r-N}\eta' > (2^{d_1+1}-2)E_r\\ & \Rightarrow \frac{3}{2}E_{r+2}+3N(n-1)lE_{r+1} > \frac{(2^{d_1+1}-2)E_r}{d_1}\\ & \Rightarrow 6(n-1)^2(2B+1)(Nl+2B+1) > \frac{(2^{d_1+1}-2)}{d_1}\end{aligned}\tag{13}$$

Since $6(n-1)^2(2B+1)(Nl+2B+1) > 3(n-1)(2B+1)$ is obviously satisfied, Eq. (13) holds. Thus, for any value of N, we can select d that satisfies $d = \{\max(d)|(2^d-1)/d \leqslant 3(n-1)(2B+1)/2, d>0\}$ to ensure $\|c^*_{\text{mult},1}F_K\|_\infty \geqslant \|c_{\text{mult},1}F_K\|_\infty$. ■

According to Theorems 1 and 2, the LBD&DEC can be used to improve the decryption structure, while decreasing the error magnitude. Consequently, Method 2 can be used to modify the NTRU-type multi-key homomorphic encryption schemes.

5 Modified NTRU-Type Multi-Key Somewhat Homomorphic Encryption

According to the analysis provided in Section 4, Method 2 has two advantages.

(1) The DEC technique improves the decryption structure of the NTRU-type scheme and eliminates the key-switching operations, which significantly decreases the dependence of the error on N (the dependence is exponentially decreasing).

(2) The LBD technique reduces the ciphertext dimension and further decreases the error magnitude.

Based on Method 2, we propose an NTRU-type MKSHE by using the LBD&DEC technique.

5.1 Modified NTRU-type MKSHE

Let λ be a security parameter; $q=q(\lambda)\in\mathbf{Z}$ is a prime integer; and $\Phi_n(x) = x^{n-1}+x^{n-2}+\cdots+1$ (n is a prime) is the sub-cyclotomic polynomial. For the polynomial ring given by $R=\mathbf{Z}[x]/\Phi_n(x)$ and $R_q = R/qR$, and a B-bound error distribution $\chi=\chi(\lambda)$ over R, set χ^l as a B-bound error distribution space, where $l=\lceil\log q\rceil$. The modified NTRU-type MKSHE (denote as M-MKSHE) can be described as follows.

(1) M-MKSHE. KeyGen(1^λ): Sample $f', g \leftarrow \chi$, and set $f = 2f' + 1$, so that $f \equiv 1 (\mathrm{mod}\ 2)$. If f is not invertible in R_q, resample $f' \leftarrow \chi$. Set $h = 2g/f \in R_q$, so pk $:= h \in R_q$, sk $:= f \in R$. Set $l = \lceil \log q \rceil$, and let discard constant d satisfy the following relationship:

$$d = \{\max(d) | (2^d - 1)/d \leqslant 3(n-1)(2B+1)/2,\ d > 0\}.$$

Output: (pk, sk, int) $= (h, f, d)$.

(2) M-MKSHE. Enc(pk, m): Compute $\hat{\boldsymbol{m}} = \mathrm{LBD}_{2\to d+1}(\mathrm{Pof}\,2(m))$, and sample $\boldsymbol{s}, \boldsymbol{e} \leftarrow \chi^{l-d}$.

Output: $\boldsymbol{c} := \hat{\boldsymbol{m}} + h\boldsymbol{s} + 2\boldsymbol{e} \in R_q^{l-d}$.

(3) M-MKSHE. Dec($\mathrm{sk}_1, \mathrm{sk}_2, \ldots, \mathrm{sk}_N, \boldsymbol{c}$): Select the first element c_1 from the ciphertext vector $\boldsymbol{c}$, let $u := (\mathrm{sk}_1 \mathrm{sk}_2 \cdots \mathrm{sk}_N) c_1 \in R_q$.

Output: $m' := u (\mathrm{mod}\ 2)$.

(4) M-MKSHE. Eval. Add($\boldsymbol{c}_1, \boldsymbol{c}_2$): Compute the addition of $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ as $\boldsymbol{c}_{\mathrm{add}} = [\boldsymbol{c}_1 + \boldsymbol{c}_2]_q$.

(5) M-MKSHE. Eval. Mult($\boldsymbol{c}_1, \boldsymbol{c}_2$): Compute the matrix $\boldsymbol{C} = \mathrm{LBD}_{2\to d+1}(\mathrm{BitD}(\boldsymbol{c}_1^{\mathrm{T}})) \in R_q^{(l-d)\times(l-d)}$.

Output: $\boldsymbol{c}_{\mathrm{mult}} = \boldsymbol{C} \cdot \boldsymbol{c}_2^{\mathrm{T}} \in R_q^{l-d}$.

5.2 Analysis results

(1) Correctness of M-MKSHE scheme

Homomorphic addition: Since $\hat{\boldsymbol{m}}_\alpha = (m_\alpha, 2^{d+1} m_\alpha, \ldots, 2^{l-1} m_\alpha)$, by using $F_K = f_1 f_2 \cdots f_r$ to decrypt the ciphertext vector, we can get $F_K \boldsymbol{c}_\alpha = (F_K m_\alpha, 2^{d+1} F_K m_\alpha, \ldots, 2^{l-1} F_K m_\alpha) + \boldsymbol{v}_\alpha$, where $\boldsymbol{v}_\alpha$ denotes the error vector. For $\boldsymbol{c}_{\mathrm{add}} = [\boldsymbol{c}_1 + \boldsymbol{c}_2]_q$, we get

$$\boldsymbol{c}_{\mathrm{add}} \cdot F_K = F_K \cdot (\boldsymbol{c}_1 + \boldsymbol{c}_2) = F_K \cdot ((m_1 + m_2), 2^{d+1}(m_1 + m_2), \ldots, 2^{l-1}(m_1 + m_2)) + F_{K-K_1} \boldsymbol{v}_1' + F_{K-K_2} \boldsymbol{v}_2',$$

where $\boldsymbol{v}_1'$ and $\boldsymbol{v}_2'$ are the error vectors with the magnitude of η. So, by selecting the first column of $\boldsymbol{c}_{\mathrm{add}} F_K$, we have

$$[c_{\mathrm{add},1} F_K]_q (\mathrm{mod}\ 2) = (m_1 + m_2).$$

Homomorphic multiplication: Considering the decryption of homomorphic multiplication, we can get $\boldsymbol{c}_{\mathrm{mult}} \cdot F_K = (\mathrm{LBD}_{2\to d+1}(\mathrm{BitD}(\boldsymbol{c}_1^{\mathrm{T}})) \cdot \boldsymbol{c}_2^{\mathrm{T}}) \cdot F_K \in R_q^{l-d}$. By selecting the first column of $\boldsymbol{c}_{\mathrm{mult}} \cdot F_K$, we get

$$c_{\mathrm{mult},1} F_K = m_1 m_2 F_K + \Bigg(m_2 F_{K-K_1} v_{1,1}' + \sum_{\varsigma=1}^{l} (c_{1,1})_\varsigma F_{K-K_2} v_{2,1}' - \sum_{\varsigma=2}^{d+1} (c_{1,1})_\varsigma F_{K-K_2} v_{2,1}' - m_2 \sum_{\varsigma=2}^{d+1} (2^{\varsigma-1} c_{1,1})_\varsigma F_K \Bigg).$$

Note that $[c_{\mathrm{mult},1} F_K]_q (\mathrm{mod}\ 2) = m_1 m_2$.

Remark: It is not necessary to consider whether $K_1 \cap K_2$ is empty.

(2) Circuit depth of M-MKSHE scheme

Theorem 3 For the parameter values provided above, the M-MKSHE can evaluate any circuit depth,

$$L = \frac{\log q}{(N - j + 1)\log(n-1) + \log\log q + O(1)}.$$

Proof As already mentioned, in this work we consider only the error of multiplication. Without loss of generation, we set $K_1 \cap K_2 = \{\mathrm{pk}_{i_1}, \ldots, \mathrm{pk}_{i_j}\}$, so $r = 2N - j$. For any level of multiplication operations, the multiplication of ciphertexts can be decrypted by F_K. According to Theorem 1, after the first level homomorphic multiplication evaluation, the error of $c_{\mathrm{mult},1}^{(1)} F_K$ (here $c_{\mathrm{mult},1}^{(1)}$ denotes the first element of vector $\boldsymbol{c}_{\mathrm{mult}}$ in the first level) is bounded,

$$\|v_{\mathrm{mult},1}^{(1)}\|_\infty \leqslant (2(n-1)(l-d)+1)\left(\frac{3}{2}E_{r+2} + 3N(n-1)(l-d)E_{r+1}\right) + (2^{d+2}-4)(n-1)E_r + N(2(n-1)(l-d)+1)((2^{d+2}-4)(n-1))^2 E_r.$$

At the second level, we set

$$\boldsymbol{c}_1^{(2)} = \begin{bmatrix} \sum_{\varsigma=d+2}^{l} (c_{1,1})_\varsigma (2^{\varsigma-1} m_2 + e_{2,\varsigma}') + (c_{1,1})_1 (2^0 m_2 + e_{2,1}') \\ \sum_{\varsigma=d+2}^{l} (c_{1,d+2})_\varsigma (2^{\varsigma-1} m_2 + e_{2,\varsigma}') + (c_{1,d+2})_1 (2^0 m_2 + e_{2,1}') \\ \vdots \\ \sum_{\varsigma=d+2}^{l} (c_{1,l})_\varsigma (2^{\varsigma-1} m_2 + e_{2,\varsigma}') + (c_{1,l})_1 (2^0 m_2 + e_{2,1}') \end{bmatrix} \in R_q^{l-d}.$$

In the same way, we can get $\boldsymbol{c}_2^{(2)} \in R_q^{l-d}$. Let $\boldsymbol{c}_1^{(2)}$ and $\boldsymbol{c}_2^{(2)}$ correspond to the public-key sets K_1 and K_2, respectively. It should be noted,

$$\boldsymbol{c}_{\mathrm{mult}}^{(2)} \cdot F_K^{(2)} = (\mathrm{LBD}_{2\to d+1}(\mathrm{BitD}(\boldsymbol{c}_1^{(2)})) \cdot \boldsymbol{c}_2^{(2)}) \cdot F_K^{(2)} \in R_q^{l-d}.$$

Further, the first column of $\boldsymbol{c}_{\mathrm{mult}}^{(2)} \cdot F_K^{(2)}$ is selected for homomorphic decryption,

$$c^{(2)}_{\text{mult},1}F^{(2)}_K = m^{(2)}_1 m^{(2)}_2 F^{(2)}_K + \left(m^{(2)}_2 F_{K-N} v^{(1)}_{\text{mult},1} + \sum_{\varsigma=1}^{l} (c_{1,1})_\varsigma F_{K-N} v^{(1)}_{\text{mult},1} - \sum_{\varsigma=2}^{d+1} (c_{1,1})_\varsigma F_{K-N} v^{(1)}_{\text{mult},1} + m^{(2)}_2 \sum_{\varsigma=2}^{d+1} (2^{\varsigma-1} c_{1,1})_\varsigma F_K \right) = m^{(2)}_1 m^{(2)}_2 F^{(2)}_K + v^{(2)}_{\text{mult},1}.$$

So, we obtain the bound of $\|v^{(2)}_{\text{mult},1}\|_\infty$,

$$\|v^{(2)}_{\text{mult},1}\|_\infty \leqslant 2(n-1)(1+2(n-1)(l-d))E_{r-N}\|v^{(1)}_{\text{mult},1}\|_\infty + (n-1)(2^{d+2}-4)E_r.$$

For convenience, let $P = (1+2(n-1)(l-d))$, $Q = (2^{d+2}-4)(n-1)$. So, we have

$$\|v^{(2)}_{\text{mult},1}\|_\infty \leqslant 2(n-1)P \times E_{r-N}\|v^{(1)}_{\text{mult},1}\|_\infty + Q \times E_r \leqslant \frac{3}{2}P^2 \times E_{2r-N+2} + 3N \times P(n-1)(l-d)E_{2r-N+1} + (1+N\times Q\times P)(P\times Q)E_{2r-N} + Q\times E_r.$$

After L levels of homomorphic operations, the error magnitude can grow up,

$$\begin{aligned}\|v^{(L)}_{\text{mult},1}\|_\infty < {} & 2(n-1)(1+2(n-1)(l-d))E_{r-N}\|v^{(L-1)}_{\text{mult},1}\|_\infty + \\ & Q\times E_r < \frac{3}{2}P^L \times E_{Lr-(L-1)N+2} + \\ & 3N\times P^{L-1}(n-1)(l-d)E_{Lr-(L-1)N+1} + \\ & (1+N\times Q\times P)(P^L\times Q)E_{Lr-(L-1)N} + \\ & Q\times\sum_{\partial=2}^{L} P^{\partial-1}\times E_{(\partial-1)r-(\partial-2)N} \qquad (14)\end{aligned}$$

where ∂ is a constant variable. The magnitude of $\|m^{(L)}_1 m^{(L)}_2 F_K\|_\infty$ is ignored because it is much smaller than $\|v^{(L)}_{\text{mult},1}\|_\infty$, and let $\|v^{(L)}_{\text{mult},1}\|_\infty < q/2$.

According to Theorem 2, if the LBD technique is not used in our scheme, the value of d is 0, which yields to the following error bound of $\|v^{(L)}_{\text{mult},1}\|_\infty$:

$$\|\tilde{v}^{(L)}_{\text{mult},1}\|_\infty < \frac{3}{2}(1+2(n-1)l)^L E_{Lr-(L-1)N+2} + 3N(1+2(n-1)l)^{L-1}(n-1)l\,E_{Lr-(L-1)N+1}.$$

Thus, by selecting $d = \{\max(d)|(2^{d+1}-2)/d \leqslant 3(n-1)(2B+1), d>0\}$, the magnitude of $\|v^{(L)}_{\text{mult},1}\|_\infty$ becomes infinitely close to $\|\tilde{v}^{(L)}_{\text{mult},1}\|_\infty$. The limit state is selected at each level, and the final error after the circuit depth of L satisfies the following relationship:

$$\begin{aligned}&\|\tilde{v}^{(L)}_{\text{mult},1}\|_\infty < \frac{q}{2} \\ &\Rightarrow L\log(1+2(n-1)l) + (Lr-(L-1)N+2)\times \log(2(n-1)(2B+1)^2) + \log\left(1+\frac{N}{4(n-1)(2B+1)^2}\right) < \\ &\quad \log q + \log 3 \\ &\Rightarrow L \approx \frac{\log q}{(N-j+1)\log(n-1)+\log\log q + O(1)} \qquad (15)\end{aligned}$$

■

According to Eqs. (14) and (15), with the increase of parameter N, the error magnitude increases, and the circuit depth decreases. However, j can reduce the impact of N, which is contrary to the BC-MKSHE scheme.

5.3 Parameters comparison

In the BC-MKSHE scheme, after one homomorphic multiplication operation, the error satisfies the following:

$\text{Error}_{\text{BC-MKSHE}} < 6l(n-1)E_{2N} + 2^{2N}E_{4N}$.

However, in our M-MKSHE scheme, the upper bound of the error is given by:

$$\text{Error}_{\text{M-MKSHE}} \leqslant (2(n-1)l+1)\times \left(\frac{3}{2}E_{r+2} + 3N(n-1)lE_{r+1}\right) + E_r.$$

The ratio of the two previous error bounds is given,

$$\begin{aligned}\text{Ratio} = \frac{\text{Error}_{\text{BC-MKSHE}}}{\text{Error}_{\text{M-MKSHE}}} \approx {} & \frac{6l(n-1)E_{2N} + 2^{2N}E_{4N}}{(2(n-1)l+1)\left(\frac{3}{2}E_{r+2}+3N(n-1)lE_{r+1}\right)+E_r} \approx \\ & \frac{2^{2N}E_{2N}}{N(n-1)\log q + O(1)} \qquad (16)\end{aligned}$$

According to Eq. (16), the error magnitude of our M-MKSHE scheme is decreased exponentially compared to the BC-MKSHE scheme.

In the following, the comparison of these two schemes regarding the other parameters is provided, such as the ciphertext size, secret key size, public key size, and evaluation key size.

Take one homomorphic multiplication operation as an example. In the BC-MKSHE scheme, the ciphertexts are two polynomials in R_q, whose degree is smaller than $(n-1)$, so the size of ciphertexts is $2(n-1)\log q$. Also, the public keys are $2N$ polynomials in R_q, so the size of public keys is $2N(n-1)\log q$. Further, the joint secret keys for decrypting are r polynomials in R, and their coefficients are smaller than $(2B+1)$, so the size of joint secret keys is $r(n-1)\log(2B+1)$. Furthermore, the evaluation keys are $\lceil \log q \rceil$-dimensional polynomials whose degree is smaller than $(n-1)$. Then, after j-times evaluations, the size of the evaluation keys is $j(n-1)\lceil \log q \rceil \log q$.

In our modified scheme, the key-switching technique is not used and none of the evaluation keys is

required. The ciphertexts are $(\lceil \log q \rceil - d)$-dimensional polynomial vectors, so their size is $2(n-1)(\lceil \log q \rceil - d)\log q$. The same as for the BC-MKSHE, the size of public keys is $2N(n-1)\log q$, and the size of joint secret keys is $r(n-1)\log(2B+1)$. See Table 1 for details, the comparison of the parameters of the M-MKSHE and BC-MKSHE schemes is provided.

As shown in Table 1, our scheme does not require the evaluation key, and the error magnitude is reduced exponentially, but the ciphertext size is increased by $(\lceil \log q \rceil - d)$ times.

6 Leveled NTRU-Type Fully Homomorphic Encryption

According to Theorem 3, the circuit depth is reduced with the decrease of N, so the modulus-reduction technique has to be used to decrease the error magnitude after every homomorphic evaluation.

Modulus-reduction[28,29]: Modulus-reduction technique can change the inner modulus q of a ciphertext c to the smaller modulus p ($p = q \bmod 2$) while roughly scaling down the error by the ratio of p/q and preserving the correctness of the decryption under the same secret key.

ModulusSwitch(c, q, p): For input $c \in R_p$, and a smaller modulus p, output is $c' \in R_p$, which is the closest element to $(p/q)\cdot c$ and $c' = c \bmod 2$.

Lemma 3[28] Let p and q be two odd modulus, let $c \in R_q$, and define $c' \in R_q$, whose value is the closest to $(p/q)c$, then $c' \equiv c (\bmod 2)$. So for any f, if $\|[fc]_q\|_\infty < q/2 - (q/p)\|f\|_1$, there is

$$[fc']_p = [fc]_q (\bmod 2), \|[fc']_p\|_\infty < (p/q)\|[fc]_q\|_\infty + \|f\|_1.$$

Then, by using the modulus-reduction technique, a leveled MKFHE scheme is designed.

6.1 Leveled NTRU-type MKFHE

The M-MKSHE is changed so that it uses modulus reduction during the homomorphic evaluations. KeyGen(1^λ) will sample a ladder of decreasing moduli $q_0 > q_1 > \cdots > q_L$. The error distribution χ is chosen in order to guarantee that any sample is B-bounded, where $B \ll q_L$. In contrast to the M-MKSHE, the M-MKFHE adopts the LBD technique twice to keep the right dimension of ciphertexts. Therefore, in the following, two kinds of LBD functions are introduced.

Let $\mathrm{LBD}^{\rightarrow}_{d_1\to d_2}(\boldsymbol{V})$ denote the $(d_1 \to d_2)$ columns of the matrix $\boldsymbol{V}$ are discarded.

Let $\mathrm{LBD}^{\downarrow}_{d_1\to d_2}(\boldsymbol{V})$ denote the $(d_1 \to d_2)$ rows of the matrix $\boldsymbol{V}$ are discarded.

The modified leveled scheme is as presented below.

(1) M-MKFHE. KeyGen(1^λ): For every $i \in \{0, 1, \ldots, L\}$, sample $g^{(i)}, f'^{(i)} \leftarrow \chi$, and set $f^{(i)} = 2f'^{(i)} + 1$, so that $f^{(i)} \equiv 1 (\bmod 2)$. If $f^{(i)}$ is not invertible in R_{q_i}, resample $f'^{(i)} \leftarrow \chi$. Let $h^{(i)} = 2g^{(i)}/f^{(i)} \in R_{q_{i-1}}$, and set pk $:= h^{(0)} \in R_{q_0}$, sk $:= f^{(L)} \in R_{q_L}$. Set the low bit discarded constant d on each ladder, where $d = \{\max(d) | (2^{d+1}-2)/d \leqslant 3(n-1)(2B+1), d > 0\}$.

Output: $\{\mathrm{pk}, \mathrm{sk}, \mathrm{int}\} = \{h^{(0)}, f^{(L)}, d\}$.

(2) M-MKFHE. Enc(pk, m): Sample $\boldsymbol{s}^{(0)}, \boldsymbol{e}^{(0)} \leftarrow \chi^{l_0-d}$, let $l_i = \lceil \log q_i \rceil$ and $\hat{\boldsymbol{m}} = (\mathrm{Pof}\,2(m))^{\mathrm{T}} \in R^{l_0}_{q_0}$. Output the ciphertext vector,

$$\boldsymbol{c}^{(0)} := h^{(0)}\boldsymbol{s}^{(0)} + 2\boldsymbol{e}^{(0)} + \mathrm{LBD}^{\downarrow}_{2\to d+1}(\hat{\boldsymbol{m}}) \in R^{l_0-d}_{q_0}.$$

(3) M-MKFHE.Dec$(\mathrm{sk}_1, \mathrm{sk}_2, \ldots, \mathrm{sk}_N, \boldsymbol{c}^{(L)})$: Select the first element $c_1^{(L)} \in R_{q_L}$ from ciphertext vector $\boldsymbol{c}^{(L)} \in R^{l_L}_{q_L}$, set $u := (\mathrm{sk}_1\mathrm{sk}_2\cdots\mathrm{sk}_N)c_1^{(L)} \in R_{q_L}$.

Output: $m' := u (\bmod 2)$.

(4) M-MKFHE . Eval. Add$(\boldsymbol{c}_1^{(i)}, \boldsymbol{c}_2^{(i)})$: For the two ciphertexts $\boldsymbol{c}_1^{(i)}, \boldsymbol{c}_2^{(i)} \in R^{l_i-d}_{q_i}$ in the i-th level, compute the addition of $\boldsymbol{c}_1^{(i)}$ and $\boldsymbol{c}_2^{(i)}$ as $\boldsymbol{c}^{(i)}_{\mathrm{add}} = [\boldsymbol{c}_1^{(i)} + \boldsymbol{c}_2^{(i)}]_{q_i} \in R^{l_i-d}_{q_i}$. Then, reduce the modulus, so we have $\boldsymbol{c}^{(i+1)}_{\mathrm{add}} = (q_{i+1}/q_i)\cdot \boldsymbol{c}^{(i)}_{\mathrm{add}} (\bmod 2)$.

Output: $\tilde{\boldsymbol{c}}^{(i+1)}_{\mathrm{add}} = \mathrm{LBD}^{\downarrow}_{l_{i+1}-d+1\to l_i-d}(\boldsymbol{c}^{(i+1)}_{\mathrm{add}})$.

(5) M-MKFHE . Eval. Mult$(\boldsymbol{c}_1^{(i)}, \boldsymbol{c}_2^{(i)})$: For the two

Table 1 Comparison of parameters between BC-MKSHE and M-MKSHE.

Parameter	BC-MKSHE scheme	M-MKSHE scheme
Maximum size of error	$6l(n-1)E_{2N} + 2^{2N}E_{4N}$	$E_r + (2(n-1)(l-d)+1)\left(\frac{3}{2}E_{r+2} + 3N(n-1)(l-d)E_{r+1}\right) + (2^{d+2}-4)(n-1)E_r + (2(n-1)(l-d)+1)\,(N(2^{d+2}-4)(n-1))^2 E_r$
Ciphertext size	$2(n-1)\log q$	$2(n-1)(\lceil \log q \rceil - d)\log q$
Evaluation key size	$j(n-1)\lceil \log q \rceil \log q$	0
Public key size	$2N(n-1)\log q$	$2N(n-1)\log q$
Secret key size	$r(n-1)\log(2B+1)$	$r(n-1)\log(2B+1)$

Note: $d = \{\max(d) | (2^{d+1}-2)/d \leqslant 3(n-1)(2B+1), d > 0\}$.

ciphertexts $\boldsymbol{c}_1^{(i)}, \boldsymbol{c}_2^{(i)} \in R_{q_i}^{\lceil \log q_i \rceil - d}$ in the i-th level, compute the multiplication of $\boldsymbol{c}_1^{(i)}$ and $\boldsymbol{c}_2^{(i)}$ as $\boldsymbol{c}_{\text{mult}}^{(i)} = \text{LBD}_{2\to d+1}^{\rightarrow}(\text{BitD}(\boldsymbol{c}_1^{(i)})) \cdot \boldsymbol{c}_2^{(i)} \in R_{q_i}^{l_i - d}$. Then, by reducing the modulus, we get $\boldsymbol{c}_{\text{mult}}^{(i+1)} = (q_{i+1}/q_i) \cdot \boldsymbol{c}_{\text{mult}}^{(i)} (\text{mod } 2)$.

Output: $\tilde{\boldsymbol{c}}_{\text{mult}}^{(i+1)} = \text{LBD}_{l_{i+1}-d+1 \to l_i - d}^{\downarrow}(\boldsymbol{c}_{\text{mult}}^{(i+1)})$.

6.2 Analysis

(1) Scheme framework

The process of homomorphic operation in the M-MKFHE scheme is shown in Fig. 1. The flowchart presented in Fig. 1 can be used as a model framework for algorithm design. In Fig. 1, the ciphertext is expanded to the vector starting from the plaintext vector, i.e., $\boldsymbol{c} := \text{LBD}_{2\to d+1}^{\downarrow}(\text{Pof2}^{\text{T}}(m)) + h\boldsymbol{s} + 2\boldsymbol{e} \in R_q^{l-d}$. Only the first element of the ciphertext vector is decrypted. Therefore, the correctness of the first term of $\text{LBD}_{2\to d+1}^{\rightarrow}(\text{BitD}(\boldsymbol{c}_1^{(i)})) \cdot \boldsymbol{c}_2^{(i)}$ should be ensured. In order to complete the homomorphic operation, the ciphertext has to be maintained in the vector form. Although the complexity of the ciphertext calculation is increased when the ciphertext vectors are multiplied, the key-switching technique is not used in our scheme.

(2) Correctness

For $B \ll q_L$, the selected LBD constant d is suitable for all levels of homomorphic operations. Thus, to reduce the modulus every time, we need to perform the $\text{LBD}_{i\to j}^{\downarrow}()$ to discard some rows of the ciphertext vector. For instance, at the i-th level, when the homomorphic operations are completed, we get the $(l_i - d)$-dimensional ciphertexts $\boldsymbol{c}_{\text{add}}^{(i)}, \boldsymbol{c}_{\text{mult}}^{(i)} \in R_{q_i}^{l_i - d}$. After the ciphertext is decomposed by BitDecomp(): $R_{q_{i+1}} \to R_{q_{i+1}}^{\lceil \log q_{i+1} \rceil}$ at the $(i+1)$-th level, $\boldsymbol{c}_{\text{add}}^{(i)}$ and $\boldsymbol{c}_{\text{mult}}^{(i)}$ are decomposed to $(l_i - d) \times (l_{i+1} - d)$-dimension matrices. Therefore, the following algorithm has to be performed, and the last $(l_i - l_{i+1})$-th rows of the matrixes have to be discarded to keep the correctness of the next homomorphic operation. The conversion progress is provided in Algorithm 1.

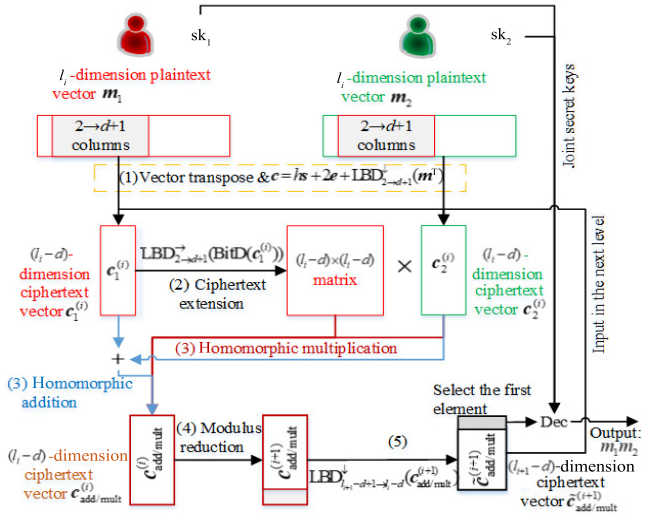


Fig. 1 Process of homomorphic operation in the M-MKFHE scheme.

The conversion algorithm is important to achieve a fully homomorphic operation in our M-MKFHE scheme. It can be seen that the ciphertext dimension is reduced with the increase in the circuit depth L by using the LBD&DEC technique. So, the modulus-reduction of our M-MKFHE scheme has two main advantages: (1) Reducing the modulus can decrease the error magnitude. (2) Reducing the modulus can also decrease the ciphertext dimension at different levels. Both of these advantages can improve the efficiency of the MKFHE scheme.

(3) Security

Our leveled M-MKFHE denotes a modified MKFHE in the LTV12 scheme. The techniques of LBD& DEC are used. The security of dimension expansion depends on the RLWE and DSPR assumptions over prime cyclotomic rings. The LBD is based on functions BitDecomp() and Powersof 2(). As known in Ref. [6], functions BitDecomp() and Powersof 2() have no effect on security. Thus, the LBD technique does not affect the security of our scheme. According to Refs. [6, 26], our M-MKFHE scheme is IND-CPA secured under the RLWE and DSPR assumptions over prime cyclotomic rings.

7 Conclusion

By using the LBD&DEC technique, our modified multi-key FHE improves the inherent homomorphic

Algorithm 1 Conversion at different levels

At the i-th level, the output is expressed as
$\boldsymbol{c}_{\text{add}}^{(i)} = [\boldsymbol{c}_1^{(i)} + \boldsymbol{c}_2^{(i)}]_{q_i} \in R_{q_i}^{l_i - d}$ and
$\boldsymbol{c}_{\text{mult}}^{(i)} = \text{LBD}_{2\to d+1}^{\rightarrow}(\text{BitD}(\boldsymbol{c}_1^{(i)})) \cdot \boldsymbol{c}_2^{(i)} \in R_{q_i}^{l_i - d}$.
At the $(i+1)$-th level, compute
$\boldsymbol{c}_{\text{add}}^{(i+1)} = (q_{i+1}/q_i)[\boldsymbol{c}_1^{(i)} + \boldsymbol{c}_2^{(i)}] (\text{mod } 2)$ and
$\boldsymbol{c}_{\text{mult}}^{(i+1)} = (q_{i+1}/q_i) \cdot \boldsymbol{c}_{\text{mult}}^{(i)} (\text{mod } 2)$.
then input
$\tilde{\boldsymbol{c}}_{\text{add}}^{(i+1)} = \text{LBD}_{l_{i+1}-d+1 \to l_i - d}^{\downarrow}(\boldsymbol{c}_{\text{add}}^{(i+1)}) \in R_{q_{i+1}}^{l_{i+1}-d}$ and
$\tilde{\boldsymbol{c}}_{\text{mult}}^{(i+1)} = \text{LBD}_{l_{i+1}-d+1 \to l_i - d}^{\downarrow}(\boldsymbol{c}_{\text{mult}}^{(i+1)}) \in R_{q_{i+1}}^{l_{i+1}-d}$.

Note: $\tilde{\boldsymbol{c}}_{\text{add}}^{(i+1)}$ or $\tilde{\boldsymbol{c}}_{\text{mult}}^{(i+1)}$ is the input ciphertext at the $(i+1)$-th level.

multiplication decryption structure of the NTRU in the LTV12 scheme, and successfully eliminates the key-switching operations and decreases the magnitude of error exponentially. Moreover, our scheme can more effectively process the quadratic part of a ciphertext product. The LBD technique used in our M-MKFHE can minimize the ciphertext dimension and improve the efficiency of the homomorphic operation.

Acknowledgment

This work was supported by the National Key R&D Program of China (No. 2017YFB0802000), the National Natural Science Foundation of China (Nos. U1636114 and 61872289), and National Cryptography Development Fund of China (No. MMJJ20170112).

References

[1] A. Hamlin, A. Shelat, M. Weiss, and D. Wichs, Multi-key searchable encryption, revisited, in *Proceedings of IACR International Workshop on Public Key Cryptography*, Berlin, Germany, 2018, pp. 95–124.

[2] O. Goldreich, S. Micali, and A. Wigderson, How to play any mental game or a completeness theorem for protocols with honest majority, in *Proceedings of the 19th Annual ACM Symposium on Theory of Computing*, New York, NY, USA, 1987, pp. 218–229.

[3] M. Ben-Or, S. Goldwasser, and A. Wigderson, Completeness theorems for non-cryptographic fault-tolerant distributed computation, in *Proceedings of the 20th Annual ACM Symposium on Theory of Computing*, Chicago, IL, USA, 1988, pp. 1–10.

[4] D. Chaum, C. Crépeau, and I. Damgård, Multiparty unconditionally secure protocols (abstract), in *Proceedings of Advances in Cryptology-CRYPTO'87*, Berlin, Germany, 1987, pp. 462–462.

[5] H. Huang, T. Gong, P. Chen, R. Malekian, and T. Chen, Secure two-party distance computation protocol based on privacy homomorphism and scalar product in wireless sensor networks, *Tsinghua Science & Technology*, vol. 21, no. 4, pp. 385–396, 2016.

[6] A. López-Alt, E. Tromer, and V. Vaikuntanathan, On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption, in *Proceedings of the Forty-Fourth Annual ACM Symposium on Theory of Computing*, New York, NY, USA, 2012, pp. 1219–1234.

[7] C. Gentry, A. Sahai, and B. Waters, Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attributebased, in *Proceedings of Advances in Cryptology-CRYPTO 2013*, Berlin, Germany, 2013, pp. 75–92.

[8] M. Clear and C. McGoldrick, Multi-identity and multi-key leveled FHE from learning with errors, in *Proceedings of Advances in Cryptology - CRYPTO 2015*, Berlin, Germany, 2015, pp. 630–656.

[9] P. Mukherjee and D. Wichs, Two round multiparty computation via multi-key FHE, in *Proceedings of Advances in Cryptology - EUROCRYPT 2016*, Berlin, Germany, 2016, pp. 735–763.

[10] C. Peikert and S. Shiehian, Multi-key FHE from LWE, revisited, in *Proceedings of Theory of Cryptography-14th International Conference*, Berlin, Germany, 2016, pp. 217–238.

[11] Z. Brakerski and R. Perlman, Lattice-based fully dynamic multi-key FHE with short ciphertexts, in *Proceedings of Advances in Cryptology-CRYPTO 2016*, Berlin, Germany, 2016, pp. 190–213.

[12] L. Chen, Z. Zhang, and X. Wang, Batched multi-hop multi-key FHE from ring-LWE with compact ciphertext extension, in *Proceedings of Theory of Cryptography Conference*, Berlin, Germany, 2017, pp. 597–627.

[13] W. Chongchitmate and R. Ostrovsky, Circuit-private multi-key FHE, in *Proceedings of IACR International Workshop on Public Key Cryptography*, Berlin, Germany, 2017, pp. 241–270.

[14] T. Li, Q Liu, and R Huang, Multi-user fully homomorphic encryption scheme based on proxy re-encryption for cloud computing, *Tsinghua Science & Technology*, vol. 58, no. 2, pp. 143–149, 2018.

[15] J. Hoffstein, N. Howgrave-Graham, J. Pipher, J. H. Silverman, and W. Whyte, NTRUSign: Digital signatures using the NTRU lattice, in *Proceedings of Cryptographers Track at the RSA Conference*, Berlin, Germany, 2003, pp. 122–140,

[16] L. Ducas, V. Lyubashevsky, and T. Prest, Efficient identity-based encryption over NTRU lattices, in *Proceedings of International Conference on the Theory and Application of Cryptology and Information Security*, Berlin, Germany, 2014, pp. 22–41.

[17] D. Li, J. Liu, Z. Zhang, Q. Wu, and W. Liu, Revocable hierarchical identity-based broadcast encryption, *Tsinghua Science & Technology*, vol. 5, no. 2, pp. 539–549, 2018.

[18] S. Garg, C. Gentry, and S. Halevi, Candidate multilinear maps from ideal lattices, in *Proceedings of Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Berlin, Germany, 2013, pp. 1–17.

[19] A. Langlois, D. Stehlé, and R. Steinfeld, GGHLite: More efficient multilinear maps from ideal lattices, in *Proceedings of EUROCRYPT 2014*, *Lecture Notes in Computer Science*, Berlin, Germany, 2014, pp. 239–256.

[20] D. Stehlé and R. Steinfeld, Making NTRU as secure as worst-case problems over ideal lattices, in *Proceedings of EUROCRYPT 2011*, *Lecture Notes in Computer Science*, Berlin, Germany, 2011, pp. 27–47.

[21] V. Lyubashevsky, C. Peikert, and O. Regev, On ideal lattices and learning with errors over rings, in *Proceedings of Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Berlin, Germany, 2010, pp. 1–23.

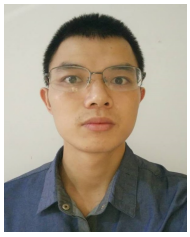
[22] M. Albrecht, S. Bai, and L. Ducas, A subfield lattice attack on overstretched NTRU assumptions, in *Proceedings of Annual Cryptology Conference*, Berlin, Germany, 2016, pp. 153–178.

[23] J. H. Cheon, J. Jeong, and C. Lee, An algorithm for NTRU problems and cryptanalysis of the GGH multilinear map without an encoding of zero, *LMS Journal of Computation and Mathematics*, vol. 19, no. 1, pp. 255–266, 2016.

[24] Y. Wang, R. Chen, C. Liu, B. Wang, and Y. Wang, Asymmetric subversion attacks on signature and

identification schemes, https://doi.org/10.1007/s00779- 018-01193-x, 2019.
[25] Z. Yang, R. Chen, C. Li, L. Qu, and G. Yang, On the security of LWE cryptosystem against subversion attacks, https://doi.org/10.1093/comjnl/bxz084, 2019.
[26] Y. Yu, G. Xu, and X. Wang, Provably secure NTRU instances over prime cyclotomic rings, in *Proceedings of IACR International Workshop on Public Key Cryptography*, Berlin, Germany, 2017, pp. 409–434.
[27] Y. Yu, G. Xu, and X. Wang, Provably secure NTRU encrypt over more general cyclotomic rings, https://eprint.iacr.org/2017/304.pdf, 2017.
[28] Z. Brakerski and V. Vaikuntanathan, Efficient fully homomorphic encryption from (standard) LWE, in *Proceedings of Annual Symposium on Foundations of Computer Science*, Los Alamitos, CA, USA, 2011, pp. 97–106.
[29] Z. Brakerski, C. Gentry, and V. Vaikuntanathan, (Leveled) Fully homomorphic encryption without bootstrapping, in *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, Cambridge, MA, USA, 2012, pp. 309–325.
[30] Y. Doröz, Y. Hu, and B. Sunar, Homomorphic AES evaluation using the modified LTV scheme, *Designs Codes and Cryptography*, vol. 80, no. 2, pp. 1–26, 2015.
[31] J. W. Bos, K. Lauter, J. Loftus, and M. Naehrig, Improved security for a ring-based fully homomorphic encryption scheme, in *Proceedings of International Conference on Cryptography and Coding*, Berlin, Germany, 2013, pp. 45–64.
[32] D. Micciancio and O. Regev. Worst-case to average-case reductions based on Gaussian measures, *Society for Industrial and Applied Mathematics Journal on Computing*, vol. 37, no. 1, pp. 267–302, 2004.
[33] Z. Chen, Research and design of fully homomorphic encryption based on lattice, PhD dissertation, Nanjing University of Aeronautics and Astronautics, Nanjing, China, 2015.
[34] T. Zhou, X. Yang , L. Liu, W. Zhang, and N. Li, Faster bootstrapping with multiple addends, *IEEE Access*, vol. 1, no. 1, pp. 49868– 49876, 2018.

Xiaoliang Che is a PhD candidate in the Engineering University of People's Armed Police, Xi'an, China. His main research interests include fully homomorphic encryption and encryption scheme based on lattice.



Tanping Zhou received the PhD degree from the Engineering University of People's Armed Police in 2018. He is now a lecturer in Engineering University of People's Armed Police. His main research interests include fully homomorphic encryption and encryption scheme based on lattice.

Ningbo Li is a PhD candidate in the Engineering University of People's Armed Police. His main research interests include fully homomorphic encryption and encryption scheme based on lattice.

Haonan Zhou is a master student in the Engineering University of People's Armed Police. His main research interests include fully homomorphic encryption and encryption scheme based on lattice.

Zhenhua Chen received the PhD degree from Shaanxi Normal University, Xi'an, China in 2014. Currently, she is an associate professor at Xi'an University of Science and Technology. Her research interests include secure multiparty computation, public-key encryption, etc.

Xiaoyuan Yang received the MS degree from Xi'an Electronic Science and Technology University, Xi'an, China in 1991. He is now a professor and a PhD supervisor in the Engineering University of People's Armed Police. His main research interests include information security and cryptology.