SCIENCE CHINA
Information Sciences

• **Supplementary File** •

Multi-Key FHE on Multi-Bit Messages

Zengpeng LI[1,2*], Chunguang MA[1*] & Hong-Sheng ZHOU[2*]

[1]*College of Computer Science and Technology, Harbin Engineering University, Harbin* 150001*, China;*
[2]*Department of Computer Science, Virginia Commonwealth University, Virginia, VA* 23284-3019*, USA.*

Appendix A Background and Related Works

Fully homomorphic encryption (FHE) [1] allows us to perform any complex computation on encrypted data without decrypting it. In particular, given a fully homomorphic encryption of a message m, anyone can compute an arbitrary (polynomial-size) arithmetic circuit C on the ciphertext, and produce an encryption of the output $C(m)$, without knowing the secret key and without decrypting the ciphertext. FHE is very useful. For example, FHE immediately gives us a solution to the problem of private computation outsourcing: the client encrypts his input m with his private key sk using an FHE scheme and sends $\mathsf{Enc}(\mathsf{sk}, m)$ to the server. The server homomorphically computes an encryption of the result $\mathsf{Enc}(\mathsf{sk}, C(m))$, and sends it back to the client. Then, the client uses his private key to decrypt and recover $C(m)$. To keep the efficient computations of the client, it is important that the bit-length of $\mathsf{Enc}(\mathsf{sk}, C(m))$ as well as the time required to decrypt it are both *independent* of the complexity of computing C; this property is called compactness.

Although encryption schemes that perform basic computations such as addition or multiplication of encrypted numbers were known for a long time, the first *fully* homomorphic encryption scheme came with the breakthrough work of Gentry [2]. Since then, several constructions were proposed, and the first generation of fully homomorphic encryption schemes (e.g., [2]) were extremely inefficient: indeed, the keys and ciphertexts in these systems were many gigabytes long, which are difficult to even store in memory. Moreover, the schemes were rather complex and relied on non-standard assumptions to argue security. Since Gentry's breakthrough, researchers have made significant efforts to improve the constructions for FHE. More concretely, the FHE constructions are now **1)** more efficient and nearly practical, **2)** based on well-understood assumptions such as learning with errors (LWE), and **3)** supporting rich functionalities (e.g., multi-key FHE). Notably, in [3–5], the authors introduced new mathematical techniques such as key switching and modulus switching that resulted in very efficient leveled FHE schemes. Furthermore, the constructions can surprisingly be based on the well understood LWE. Since then, lots of efforts have been made to improve the performance and have resulted in implementations which showed orders of magnitude speedup than before. Please see the great example of homomorphic encryption library by Halevi and Shoup [6,7]. Techniques along this line have recently been further improved [8, 9].

Multi-key FHE (for single bit). While standard FHE schemes allow computation on data encrypted under a single user's key, many scenarios require computations on data belonging to multiple unrelated users. What we need is a multi-key FHE (hereafter MFHE) which is capable of operating on inputs encrypted under multiple, unrelated keys. A ciphertext resulting from a multi-key evaluation can be jointly decrypted using the secret keys of all the users involved in the computation. In [10], the authors demonstrate a multi-key FHE scheme based on NTRU. This line of research has been improved recently, and in [11–14], the authors can construct multi-key FHE schemes based on more standard LWE assumptions. Specifically, we first consider a scenario in which Alice and Bob have their own key pair, in order to achieve the fact that Alice can decrypt Bob's ciphertexts by using her secret key and without knowing Bob's secret key. In this setting, Mukherjee and Wichs [12] constructed an important "Linear Combination Procedure" (hereafter LCP) which could offer auxiliary information to Alice to help her decrypt Bob's ciphertexts in their multi-key FHE scheme [12].

Concretely, utilizing Gentry-Sahai-Waters [9] (GSW) encryption algorithm to encrypt the individual entries $\mathbf{R}[i, j]$ of random matrix $\mathbf{R} \in \{0, 1\}^{m \times m}$, then, LCP takes all of the encryption of $\mathbf{R}[i, j]$ as input and outputs the auxiliary information. Obviously, the variant of multi-bit FHE (hereafter mFHE) is at the core of LCP construction.

Subsequently, Brakerski et al. [13] proposed a multi-key FHE scheme with short ciphertexts for designing MPC protocol. They adopt a variant of LCP to encrypt each entry of secret key. Moreover, Peikert et al. [14] proposed two multi-key FHE schemes: the first one is suitable for dynamic on-the-fly MPC, and the second one supports efficient homomorphic operations. However, in their scheme, they need to encrypt the commitment random matrix, which is similar to the LCP of [12]. Apparently, LCP construction is an important tool to obtain multi-key FHE for single-bit message.

* Corresponding author (email: lizengpeng@hrbeu.edu.cn, machunguang@hrbeu.edu.cn, hszhou@vcu.edu)

Multi-key FHE for multiple bits. We are making efforts along this line. In this paper, we study multi-key FHE for multi-bit messages.

As discussed above, we observe that, in Mukherjee-Wichs scheme [12], (i.e., multi-key and single-bit FHE), the core of LCP construction is a variant of "single-key and multi-bit" FHE scheme which uses GSW-like encryption algorithm to encrypt each entry of random matrix (or bit-by-bit).

It seems that undesirable overhead is introduced. This leads to the following natural question:

Is that possible to improve LCP *so that more efficient multi-key* FHE *scheme can be developed for encrypting multi-bit messages?*

We give an affirmative answer to the above question. To avoid encrypting each entry of the random matrix $\mathbf{R}$, in our paper, we improve the "LCP" construction of Mukherjee-Wichs scheme by encrypting the random *diagonal* matrix $\mathbf{R}$ directly, which can be viewed as a variant of multi-bit FHE.

Note that, LCP could be improved from different angles. For example, very recently, in [15], the authors proposed an interesting idea to construct *efficient* multi-bit (single-key) FHE. Unfortunately, their scheme cannot be used for achieving our goal. We expect that the LCP can recover the whole plaintexts by using "*one-time*" decryption in the sense that one can decrypt the multi-bit ciphertext directly rather than decrypting it in bit-by-bit manner.

In our design, we first, construct single-key and multi-bit FHE (i.e., mFHE) with the one-time decryption. Then, we use mFHE to construct an improved version of LCP where the ciphertext matrix can be decrypted directly. Finally, we utilize the improved LCP to construct multi-key FHE on multi-bit messages. We remark that, our mFHE scheme has additional "*flexible decryption*" feature: we can decrypt certain specified bits of the underlying plaintexts; we do not need to decrypt the entire ciphertexts if our goal is to open a small portion of them.

Appendix B Preliminaries

In this section we provide required preliminaries. We note that, the definitions and lemmas are taken from previous work.

Appendix B.1 Notation

For $n \in \mathbb{N}$, we let $[n]$ denote the set $\{1, \cdots, n\}$. For a real number $x \in \mathbb{R}$, we let $\lfloor x \rfloor$ denote the largest integer not greater than x, and $\lfloor x \rceil := \lfloor x + \frac{1}{2} \rfloor$ denote the integer closest to x, with ties broken upward. We use bold lower-case letters, e.g., $\mathbf{x}$, to denote vectors, and bold upper-case letters, e.g., $\mathbf{A}$, to denote matrices. We use ":=" to denote deterministic assignment.

Appendix B.2 Discrete Gaussian

Definition 1. For a subset $L \in \mathbb{Z}^m$, a vector $\mathbf{u} \in \mathbb{R}^m$ and a positive parameter $\sigma \in \mathbb{R}$, define the Gaussian function $\rho_{\sigma,\mu}(x)$ and the probability density function $\rho_{\sigma,\mu}(L)$ as follows:

$$\rho_{\sigma,\mu}(x) = \exp\big(-\pi \cdot \frac{||x-\mu||^2}{\sigma^2}\big) \quad and \quad \rho_{\sigma,\mu}(L) = \sum_{x \in L} \rho_{\sigma,\mathbf{u}}(x)$$

For any y in L, there exists the discrete Gaussian distribution $D_{L,\sigma,\mu}(y) = \dfrac{\rho_{\sigma,\mu}(y)}{\rho_{\sigma,\mu}(L)}$ over L with center μ and parameter σ.

Definition 2 (B-bounded distributions)**.** A distribution ensemble $\chi = \chi(\lambda)$ over the integers is called B-bounded (denoted $|\chi| \leqslant B$) if there exists: $\Pr\limits_{x \xleftarrow{\$} \chi}[|x| \geqslant B] \leqslant 2^{-\tilde{\Omega}(n)}$.

Lemma 1.
1. For $\forall\ k > 0$, $Pr[|e| > k \cdot \sigma, e \leftarrow D_\sigma^1] \leqslant 2 \cdot \exp(-\frac{k^2}{2})$;
2. For $\forall\ k > 0$, $Pr[\|\mathbf{e}\| > k \cdot \sigma \cdot \sqrt{m}, \mathbf{e} \leftarrow D_\sigma^m] \leqslant k^m \cdot \exp(\frac{m}{2} \cdot (1-k^2))$.

Note that the above lemma is from [16, Lemma 4.4].

Remark 1. Throughout the paper, we suppose $\sigma \geqslant 2\sqrt{n}$. Therefore, if $\mathbf{e} \leftarrow D_\sigma^m$ then we have, on average, that $\|\mathbf{e}\| \approx \sqrt{m} \cdot \sigma$. Lemma 1 (2) implies that $\|\mathbf{e}\| \leqslant 2\sigma\sqrt{m}$ with overwhelming probability.

Lemma 2 (Smudging Lemma)**.** Let $B_1 = B_1(\lambda)$, and $B_2 = B_2(\lambda)$ be positive integers and let $e_1 \in [-B_1, B_1]$ be a fixed integer. Let $e_2 \leftarrow [-B_2, B_2]$ be chosen uniformly at random. Then the distribution of e_2 is statistically indistinguishable from that of $e_2 + e_1$ as long as $B_1/B_2 = \mathsf{negl}(\lambda)$.

Note that the above lemma is from [17, Lemma 2.1].

Appendix B.3 Learning with Errors

Definition 3 (LWE Distribution)**.** For the security parameter λ, let $n = n(\lambda)$ and $m = m(\lambda)$ be integers, let $\chi = \chi(\lambda)$ be error distribution over $\mathbb{Z}$ bounded by $B = B(\lambda)$, and let $q = q(\lambda) \geqslant 2$ be an integer modulus for any polynomial $p = p(\lambda)$ such that $q \geqslant 2^p \cdot B$. Then, sample a vector $\mathbf{s} \in \mathbb{Z}_q^{n \times 1}$ called the secret, the LWE distribution $\mathcal{B}_{\mathbf{s},\chi}$ over $\mathbb{Z}_q^n \times \mathbb{Z}_q$ is sampled by choosing $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ uniformly at random, choosing $\mathbf{e} \leftarrow \chi^{m \times 1}$, and outputting $\big(\mathbf{A}, \mathbf{b} = \mathbf{A} \cdot \mathbf{s} + \mathbf{e} \pmod q\big)$.

There are two main versions of the LWE assumption: search version and decision version. We define the decisional version as follows,

Definition 4 (Decision-$\mathsf{LWE}_{n,q,\chi,m}$)**.** Assume given an independent sample $(\mathbf{A}, \mathbf{b}) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^{m \times 1}$, where the sample is distributed according to either: (1) $\mathcal{A}_{\mathbf{s},\chi}$ for a uniform random $\mathbf{s} \in \mathbb{Z}_q^n$ (i.e., $\{(\mathbf{A}, \mathbf{b}) : \mathbf{A} \leftarrow \mathbb{Z}_q^{m \times n}, \mathbf{s} \leftarrow \mathbb{Z}_q^{n \times 1}, \mathbf{e} \leftarrow \chi^{m \times 1}, \mathbf{b} = \mathbf{A} \cdot \mathbf{s} + \mathbf{e} \pmod q\}$), or (2) the uniform distribution (i.e., $\{(\mathbf{A}, \mathbf{b}) : \mathbf{A} \leftarrow \mathbb{Z}_q^{m \times n}, \mathbf{b} \leftarrow \mathbb{Z}_q^{m \times 1}\}$). Then, the above two distributions are computationally indistinguishable.

Remark 2. Regev and others [12, 18–20] showed the reductions between the LWE assumption and approximating the shortest vector problem in lattices (for appropriate parameters). We omit the corollary of these schemes' results and refer to find further details from [12, 18–20].

The following lemma is from [21, 22].

Lemma 3. For any $N \geqslant m\lceil \log q \rceil$ there exists a computable gadget matrix $\mathbf{G} \in \mathbb{Z}_q^{m \times N}$ and an efficiently computable deterministic inverse (a.k.a., "short preimage") function $\mathbf{G}^{-1}(\cdot)$. The inverse function $\mathbf{G}^{-1}(\mathbf{M})$ takes as input a matrix $\mathbf{M} \in \mathbb{Z}_q^{m \times m'}$ for any m' and outputs a matrix $\mathbf{G}^{-1}(\mathbf{M}) \in \{0,1\}^{N \times m'}$ such that $\mathbf{G}\mathbf{G}^{-1}(\mathbf{M}) = \mathbf{M}$.

Remark 3. For future convenience, the gadget matrix $\mathbf{G}$ from Micciancio and Peikert [21] can be expressed by $\mathbf{G} = \mathbf{I}_m \otimes \mathbf{g} \in \mathbb{Z}_q^{m \times N}$ where $\mathbf{g} = (1, 2, 4, \ldots, 2^{l-1})^T$. Actually, we can regard the gadget matrix as the "powers-of-two" matrix, hence, for $\mathbf{v} \in \mathbb{Z}_q^m$ we have $\mathsf{PowerOf2}(\mathbf{v}) = \mathbf{v}^T\mathbf{G}$. For $\mathbf{v} \in \mathbb{Z}_q^N$ we have $\mathsf{BitDecomp}^{-1}(\mathbf{v}) = \mathbf{G}\mathbf{v}$. For $\mathbf{a} \in \mathbb{Z}_q^m$ the algorithm $\mathsf{BitDecomp}(\mathbf{a})$ can be renamed as $\mathbf{G}^{-1}(\mathbf{a})$.

Appendix B.4 Multi-Key Fully Homomorphic Encryption

A homomorphic encryption scheme is *multi-key* if it can evaluate circuits on ciphertexts encrypted under multiple (different) public keys. To decrypt an evaluated ciphertext, the algorithm uses the secret keys of all parties whose ciphertexts took part in the computation.

Definition 5 (Multi-Key (Leveled) FHE, [10, 12])**.** A multi-key (leveled) FHE is a tuple of probabilistic polynomial time (PPT) algorithms $\mathsf{MFHE} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Expand}, \mathsf{Eval}, \mathsf{Dec})$ described as follows:

– $\mathsf{params} \leftarrow \mathsf{Setup}(1^\lambda, 1^L)$: Given the security parameter λ and the circuit depth L, outputs the public parameters params. We assume that all the other algorithms take params as input implicitly.

– $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(\mathsf{params})$: Output secret key sk and public key pk.

– $c \leftarrow \mathsf{Enc}(\mathsf{pk}, \mu)$: Given the public key pk and a message $\mu \in \{0,1\}$, then output a ciphertext c.

– $\hat{c} \leftarrow \mathsf{Expand}((\mathsf{pk}_1, \cdots, \mathsf{pk}_N), i, c)$: Given a sequence of N public keys and a fresh ciphertext c under the i-th key pk_i, then it outputs an "expanded" ciphertext $\hat{c}$.

– $\mu := \mathsf{Dec}(\mathsf{params}, (\mathsf{sk}_1, \cdots, \mathsf{sk}_N), \hat{c})$: Given some ciphertext $\hat{c}$ and a sequence of N secret keys $(\mathsf{sk}_1, \cdots, \mathsf{sk}_N)$ to recover the message $\mu \in \{0,1\}$.

– $\hat{c} := \mathsf{Eval}(\mathsf{params}, C, (\hat{c}_1, \cdots, \hat{c}_l), (\mathsf{pk}_1, \cdots, \mathsf{pk}_N))$: Utilizing the sequence $(\mathsf{pk}_1, \cdots, \mathsf{pk}_N)$, apply a circuit $C \in :\{0,1\}^l \rightarrow \{0,1\}$ to ciphertexts $c_1, \cdots, c_l$, where each c_j is evaluated under a sequence of public keys $T_j \subset \{\mathsf{pk}_1, \cdots, \mathsf{pk}_N\}$ (we assume that T_j is implicit in c_j). Upon termination, outputs a ciphertext $\hat{c}$.

Remark 4. The homomorphic evaluation in [10, 12] and our schemes are given all the input ciphertexts and public keys from the start, i.e. before the homomorphic evaluation, all relevant keys must be known.

The following properties are security, correctness and compactness. In more detail:

Correctness: For any circuit $C \in \mathcal{C} : \{0,1\}^l \rightarrow \{0,1\}$ on N inputs, any l-tuple of input sequence $\mu_1, \cdots, \mu_l$ for circuit C and any sequences of N key pairs $\{(\mathsf{pk}_i, \mathsf{sk}_i) \leftarrow \mathsf{KeyGen}(\mathsf{params})\}_{i \in [N]}$, where params is generated by $\mathsf{Setup}(1^\lambda, 1^L)$. If generating l ciphertexts $\{c_i \leftarrow \mathsf{Enc}(\mathsf{pk}_i, \mu_i)\}_{i \in [l]}$ and the corresponding expanded ciphertexts $\hat{c}_i \leftarrow \mathsf{Expand}((\mathsf{pk}_1, \cdots, \mathsf{pk}_N), i, c_i)$ with $i \in [l]$, then we obtain $C(m_1, \cdots, m_l) = \mathsf{Dec}\big((\mathsf{sk}_1, \cdots, \mathsf{sk}_N), (\mathsf{Eval}(\mathsf{pk}, C, \hat{c}_1, \cdots, \hat{c}_l))\big)$ except with negligible probability (in λ) taken over the random of all these algorithms.

Compactness: There exists a polynomial $p(\lambda, L, N)$ satisfying that $|\hat{c}| < p(\lambda, L, N)$. Hence, the size of $|\hat{c}|$ should be independent of C and l, but can depend on λ, L and N.

Semantic security: For any two messages μ_0 and μ_1 and any $L = L(\lambda)$, the following distributions $(\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, \mu_0))$ and $(\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, \mu_1))$ are indistinguishable, namely that $(\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, \mu_0)) \approx_c (\mathsf{pk}, \mathsf{Enc}(\mathsf{pk}, \mu_1))$, where $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(\mathsf{params})$. We stress that, the security for multi-bit MFHE is the same as that for single-bit MFHE . Since in the public key setting, IND-CPA security for encryption of a single message implies IND-CPA security for encryption of multiple messages. More details see [23]. Note that, when we set $N := 1$, we obtain single-key FHE (from the above MFHE).

Appendix C The Construction of mFHE Scheme

In this section, we present the construction of multi-bit Gentry-Sahai-Waters scheme (mFHE scheme).

Appendix C.1 Multi-Bit Gentry-Sahai-Waters Scheme (mFHE Scheme)

Given security parameter λ, we set t to be the number of secret key and the size of message. We now give the formal details. The variant of GSW scheme is defined similarly to the FHE proposed in [12, 22, 24–27].

Setup $\mathsf{params} \leftarrow \mathsf{mFHE.Setup}(1^\lambda, 1^L)$**:**

1. The $\mathsf{Setup}(\cdot)$ takes the parameters λ and L as input. We denote $\chi = \chi(\lambda)$, $n = n(\lambda)$, and $m = m(\lambda, L) = \mathcal{O}(n \log q)$ as above so that the (m, n, q, χ)-LWE assumption achieves at least 2^λ security against known attacks. Then chooses a parameter $t = \mathcal{O}(\log(n))$ (as the number of secret keys);
2. Moreover, let $l = \lfloor \log q \rfloor + 1$ and $N = (n + t) \cdot l$ and output $\mathsf{params} = (n, q, \chi, m)$.

Key Generation $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{mFHE.KeyGen}(\mathsf{params})$**:**

1. Sample $\mathbf{t}_i^T = (t_{i,1}, \cdots, t_{i,n}) \in \mathbb{Z}_q^{1\times n}$ and output $\mathsf{sk}_i := \mathbf{s}_i = (\mathbf{I}_i \mid -\mathbf{t}_i^T)^T \in \mathbb{Z}_q^{(n+t)\times 1}$. Here it is important to remark that $\mathbf{v}_i = \mathsf{PowerOf2}(\mathbf{s}_i)$. Most importantly, the secret key matrix $\mathsf{sk} := \mathbf{S} = [\mathsf{sk}_1, \cdots, \mathsf{sk}_t] = [\mathbf{s}_1, \cdots, \mathbf{s}_t] \in \mathbb{Z}_q^{(n+t)\times t}$;
2. Sample $\mathbf{B} \leftarrow \mathbb{Z}_q^{n\times m}$ and $\mathbf{e}_i \leftarrow \chi^{m\times 1}$ $i \in [t]$, then compute $\mathbf{b}_i = \mathbf{B}\cdot\mathbf{t}_i + \mathbf{e}_i \pmod q$, and output $\mathsf{pk} := \mathbf{A} = [\mathbf{b}_1 \mid \cdots \mid \mathbf{b}_t \mid \mathbf{B}] \in \mathbb{Z}_q^{m\times(n+t)}$, where the size of pk is $\mathcal{O}(nm\cdot\log^2 q)$. Finally, we observe that $\mathbf{A}\cdot\mathbf{s}_i = \mathbf{e}_i$ and $\mathbf{A}\cdot\mathbf{S} = [\mathbf{e}_1, \cdots, \mathbf{e}_t]$.

Encryption $\mathbf{C} \leftarrow \mathsf{mFHE.Enc}(\mathsf{params}, \mathsf{pk}, \mathbf{M})$**:**

1. Sample a uniform matrix $\mathbf{R} \leftarrow \{0,1\}^{m\times N}$, in order to encrypt t messages $u_i \in \{0,1\}$ for $i \in [t]$, we first embed the t bits into the *message* matrix $\mathbf{U} = \mathrm{diag}(\mu_1, \cdots, \mu_t)$, then create the *plaintext* matrix as follows
$$\mathbf{M} = \left(\begin{array}{c:c} \mathbf{U}_{t\times t} & \mathbf{0}_{t\times n} \\ \hdashline \mathbf{0}_{n\times t} & \mathbf{E}_{n\times n} \end{array}\right) \in \{0,1\}^{(n+t)\times(n+t)},$$
where the matrix $\mathbf{U} \in \mathbb{Z}_q^{t\times t}$ and $\mathbf{E} \in \{0,1\}^{n\times n}$ are diagonal matrices, i.e., $\mathbf{U} = \mathrm{diag}(\mu_1, \cdots, \mu_t)$ and $\mathbf{E} = \mathrm{diag}(1, \cdots, 1)$, they are also the two partitioned matrices of the *plaintext* matrix $\mathbf{M}$;
2. Compute and write $\mathbf{C} = \mathbf{M}\cdot\mathbf{G} + \mathbf{A}^T\cdot\mathbf{R} \pmod q \in \mathbb{Z}_q^{(n+t)\times N}$, we stress that $\mathbf{G} = \mathsf{BitDecomp}^{-1}(\mathbf{I}_{n+t}) = (g^T \otimes \mathbf{I}_{n+t}) \in \mathbb{Z}_q^{(n+t)\times(n+t)\cdot l}$, where $\mathbf{I}_{n+t}$ denotes the $(n+t)$-dimension identity matrix and thus $g^T = [2^0, 2^1, \cdots, 2^{l-1}] \in \mathbb{Z}_q^l, l = \lceil \log q \rceil = \lfloor \log q \rfloor + 1$, for $m \geqslant n\lceil \log q \rceil$, namely $m = \mathcal{O}(n(\log q))$.

Flexible Decryption $u_i \leftarrow \mathsf{mFHE.bitDec}(\mathsf{params}, \mathsf{sk}_i, \mathbf{C})$**:** we can use *"flexible decryption"* algorithm to decrypt certain specified bits of the underlying plaintexts and we don't need to decrypt the entire ciphertexts if our goal is to open a small portion of them. We call it as single-bit decryption bitDec.

1. Suppose we want to decrypt i-th row and j-th column bit $c_{i,j}$, thus we let $\mathsf{sk}_i = \mathbf{s}_i := (\mathbf{I}_i \mid -\mathbf{t}_i^T)^T \in \mathbb{Z}_q^{(n+t)\times 1}$, and then we define a vector $\mathbf{w} \in \mathbb{Z}_q^{1\times(n+t)}$ such that the j-th position is $\lceil q/2 \rceil$ and other positions are zero for $j \in [t]$, $\mathbf{w}_j^T = [\underbrace{0, \cdots, \lceil q/2 \rceil_j, \cdots, 0}_{t} \mid \underbrace{0, \cdots, 0}_{n}]$;
2. For $i, j = 1$ to t, compute $\langle \mathbf{s}_i, \mathbf{C} \rangle = \mathbf{s}_i^T\mathbf{A}\mathbf{R} + \mathbf{s}_i^T\mathbf{M}\mathbf{G} = \mathbf{e}_i^T\mathbf{R} + \mathbf{s}_i^T\mathbf{M}\mathbf{G}$ and set $v_{i,j} = \mathbf{s}_i^T\mathbf{C}\cdot\mathbf{G}^{-1}(\mathbf{w}_j^T)$;
3. Output the decryption message $u_{i,j} = \left| \lfloor \frac{v_{i,j}}{q/2} \rceil \right|$, where $\lfloor\cdot\rceil$ denotes the operation of rounding to the nearest integer. Hence, by construction we have that the output belongs to $\{0,1\}$.
4. Finally, repeat t^2 times, we can recover the whole messages.

Considering that the flexible (i.e., single-bit) decryption algorithm of mFHE scheme works as described above, we can get each bit of the message using the bitDec algorithm with the appropriate secret keys. We now present the one-time (i.e., multi-bit) decryption algorithm of mFHE scheme, which allows recovering all the bits of the message simultaneously.

One-Time Decryption $\mathbf{U} \leftarrow \mathsf{mFHE.Dec}(\mathsf{params}, \mathsf{sk}, \mathbf{C})$**:**

1. Denote a matrix
$$\mathbf{W}^T = \left(\begin{array}{ccc|c} \lceil q/2 \rceil, & \cdots, & 0 & \mathbf{0}^{1\times n} \\ \vdots & \ddots & \vdots & \vdots \\ 0, & \cdots, & \lceil q/2 \rceil & \mathbf{0}^{1\times n} \end{array}\right) \in \mathbb{Z}_q^{t\times(n+t)}$$
and construct the new secret key matrix $\mathbf{S} = (\mathbf{s}_1, \cdots, \mathbf{s}_t) = \left(\begin{array}{c} \mathbf{I} \\ \hline -\mathbf{t}_1, \cdots, -\mathbf{t}_t \end{array}\right) \in \{0,1\}^{(n+t)\times t}$;
2. Compute $\langle \mathbf{S}, \mathbf{C} \rangle = \mathbf{S}^T\mathbf{A}^T\mathbf{R} + \mathbf{S}^T\mathbf{M}\mathbf{G} = [\mathbf{e}_1, \cdots, \mathbf{e}_t]^T\mathbf{R} + \mathbf{S}^T\mathbf{M}\mathbf{G}$ and $\mathbf{V} = \mathbf{S}^T\mathbf{C}\cdot\mathbf{G}^{-1}(\mathbf{W}^T)$;
3. Output the decryption message $\mathbf{U} = \left| \lfloor \frac{\mathbf{V}_{i,j}}{q/2} \rceil \right|$.

Remark 5. In this setting, we stress that the matrix $\mathbf{I}$ is identity matrix, each $\mathbf{s}_i = [0, \cdots, 1, \cdots, 0 \mid -\mathbf{t}_i^T]^T \in \mathbb{Z}_q^{(n+t)\times 1}$. We have that $\mathbf{A}\cdot\mathbf{S} = [\mathbf{e}_1, \cdots, \mathbf{e}_t] \pmod q$.

Normally, we can choose different secret keys sk_i to decrypt the ciphertext $\mathbf{C}$ bit-by-bit and get the i-th bit message corresponding to i-th secret key. But actually we can use secret key matrix $\mathbf{S}$ to recover the whole messages using the one-time decryption algorithm described above. We compute $\mathbf{V}_{i,j} = \mathbf{S}^T\mathbf{C}\cdot\mathbf{G}^{-1}(\mathbf{W}^T)$ in order to get the result as follows:
$$\mathbf{V}_{i,j} = \mathbf{S}^T\mathbf{C}\cdot\mathbf{G}^{-1}(\mathbf{W}^T) = \lceil \frac{q}{2} \rceil \cdot \begin{pmatrix} u_{1,1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & u_{t,t} \end{pmatrix} + \begin{pmatrix} \mathbf{e}_1^T\mathbf{R} \\ \vdots \\ \mathbf{e}_t^T\mathbf{R} \end{pmatrix} \cdot \mathbf{G}^{-1}(\mathbf{W}^T);$$
It is straightforward to compute the magnitude of noise and verify that it grows linearly when compared to the single-bit decryption algorithm. Next, we will show the analysis in detail.

Homomorphic Evaluation $\mathsf{mFHE.Eval}(\mathsf{params}, \mathbf{C}_1, \cdots, \mathbf{C}_l)$**:** contains Add and Mult, in more detail:

- $\mathsf{mFHE.Add}(\mathbf{C}_1, \mathbf{C}_2)$: output

$$\mathbf{C}_1 + \mathbf{C}_2 = (\mathbf{M}_1 + \mathbf{M}_2)\mathbf{G} + \mathbf{A}^T(\mathbf{R}_1 + \mathbf{R}_2) \in \mathbb{Z}_q^{(n+t)\times N};$$

- $\mathsf{mFHE.Mult}(\mathbf{C}_1, \mathbf{C}_2)$: output the matrix product, because $\mathbf{C}_2 = \mathbf{M}_2 \cdot \mathbf{G} + \mathbf{A}^T \cdot \mathbf{R}_2$, then we get

$$\begin{aligned}\mathbf{C}_1\mathbf{G}^{-1}(\mathbf{C}_2) &= \left(\mathbf{M}_1 \cdot \mathbf{G} + \mathbf{A}^T \cdot \mathbf{R}_1\right) \cdot \mathbf{G}^{-1}(\mathbf{C}_2) = \mathbf{M}_1 \cdot \mathbf{C}_2 + \mathbf{A}^T \cdot \mathbf{R}_1 \cdot \mathbf{G}^{-1}(\mathbf{C}_2) \\ &= \mathbf{M}_1\mathbf{M}_2 \cdot \mathbf{G} + \mathbf{A}^T\mathbf{R}_1 \cdot \mathbf{G}^{-1}(\mathbf{C}_2) + \mathbf{M}_1\mathbf{A}^T\mathbf{R}_2 \in \mathbb{Z}_q^{(n+t)\times N}\end{aligned}$$

This also allows us to compute a homomorphic NAND gate by outputting $\mathbf{G} - \mathbf{C}_1\mathbf{G}^{-1}(\mathbf{C}_2)$.

Appendix C.2 Correctness and Homomorphic Operations Analysis of mFHE Scheme

Next we will analyze the scheme's correctness and homomorphic operations.

Definition 6. We say that a ciphertext $\mathbf{C}$ that is designed to encrypt $\mathbf{U} \in \mathbb{Z}_q^{t\times t}$, under t different secret keys $\mathbf{s}_i$, for $i \in [t]$, it has error vector $\mathbf{err} \in \mathbb{Z}_q^{1\times N}$, if $\mathbf{err}_i = \mathbf{s}_i^T\mathbf{C} - \mathbf{s}_i^T\mathbf{M}\mathbf{G} = \mathbf{s}_i^T\mathbf{A}^T\mathbf{R} = \mathbf{e}_i^T \cdot \mathbf{R} \pmod q$. Obviously, for secret key matrix $\mathbf{S} = [\mathbf{s}_1, \cdots, \mathbf{s}_t] \in \mathbb{Z}_q^{(n+t)\times t}$, it has error matrix $\mathbf{Err} = (\mathbf{err}_1, \cdots, \mathbf{err}_t)^T \in \mathbb{Z}_q^{t\times N}$, if the error matrix $\mathbf{Err} = \mathbf{S}^T \cdot \mathbf{C} - \mathbf{S}^T \cdot \mathbf{M} \cdot \mathbf{G} = \mathbf{S}^T \cdot \mathbf{A}^T \cdot \mathbf{R} \pmod q$.

In order to analyze correctness, we first define the following notion of *noisy ciphertexts*.

Lemma 4 (E-noisy ciphertext)**.** An E-noisy ciphertext, for a corresponding message $\mathbf{M}$ and secret key $\mathsf{sk} = \mathbf{s}_i \in \mathbb{Z}_q^{(n+t)\times 1}$, is a matrix $\mathbf{C} \in \mathbb{Z}_q^{(m+1)\times N}$ such that $\langle \mathbf{s}_i, \mathbf{C}\rangle = \mathbf{s}_i^T \cdot \mathbf{M} \cdot \mathbf{G} + \mathbf{e}_i^T \cdot \mathbf{R}$. Then, we set $\mathbf{err}_i = \mathbf{e}_i^T\mathbf{R}$ with $||\mathbf{err}_i|| \leqslant N \cdot mB \leqslant E$. Furthermore, if we use the multi-bit decryption secret key matrix $\mathbf{S} \in \mathbb{Z}_q^{(n+t)\times t}$, we obtain $\mathbf{Err} = \langle \mathbf{S}, \mathbf{C}\rangle - \mathbf{S}^T \cdot \mathbf{M}\mathbf{G} = (\mathbf{A}\mathbf{S})^T \cdot \mathbf{R} \pmod q$ with $||\mathbf{Err}|| \leqslant t \cdot ||\mathbf{err}_i|| \leqslant t \cdot E$.

Proof. For $\mathbf{e} \leftarrow \chi^m$ and $\forall e_i \leftarrow \chi$, $|x_i| \leqslant B$ (where $B \ll q$ is a bound on the values of χ). We set $\mathbf{err}_i = \mathbf{e}_i^T\mathbf{R}$ with $||\mathbf{err}_i|| \leqslant ||\mathbf{e}_i^T\mathbf{R}|| \leqslant ||\mathbf{e}_i^T||_2 \cdot ||\mathbf{R}||_\infty \leqslant N \cdot mB \leqslant E$. By Definition 6, the error matrix $\mathbf{Err} = (\mathbf{err}_1, \cdots, \mathbf{err}_t)^T \in \mathbb{Z}^{t\times N}$, clearly, $||\mathbf{Err}|| = [\mathbf{e}_1, \cdots, \mathbf{e}_t]^T \cdot \mathbf{R} = t \cdot (\mathbf{e}_i^T\mathbf{R}) \leqslant t \cdot ||\mathbf{err}_i|| \leqslant t \cdot E$.

Next we analyze the correctness of decryption.

Lemma 5. Let $\mathbf{C}$ be an E-noisy encryption of $\mathbf{M}$, if we want to decrypt i-th row and j-th ciphertext, then there exists a secret key $\mathbf{s}_i$ such that $\langle \mathbf{s}_i, \mathbf{C}_{i,j}\rangle = \mathbf{err}_i + \mathbf{s}_i^T \cdot \mathbf{M}_{i,j} \cdot \mathbf{G}$ with $||\mathbf{err}_i|| \leqslant E$ by Lemma 4. Then we obtain $v_{i,j} = \mathbf{s}_i^T\mathbf{C}_{i,j} \cdot \mathbf{G}^{-1}(\mathbf{w}^T) = \mathbf{err}_i^{dec} + \lceil \frac{q}{2} \rceil \cdot u_i \pmod q$ with $\|\mathbf{err}_i^{dec}\|_\infty \leqslant \|\mathbf{e}_i^T\mathbf{R}\| \cdot \|\mathbf{G}^{-1}(\mathbf{w}^T)\| \leqslant N \cdot E$. Further, if we want to decrypt ciphertext $\mathbf{C}$, then there exists a secret key matrix $\mathbf{S}$ such that $\langle \mathbf{S}, \mathbf{C}\rangle = \mathbf{Err} + \mathbf{S}^T \cdot \mathbf{M} \cdot \mathbf{G}$ with $||\mathbf{Err}|| \leqslant t \cdot E$, by Lemma 4. Then we obtain $\mathbf{V}_{i,j} = \mathbf{S}^T\mathbf{C} \cdot \mathbf{G}^{-1}(\mathbf{W}^T) = \mathbf{Err}^{dec} + \lceil \frac{q}{2} \rceil \cdot \mathbf{U}^T \pmod q$ with $\|\mathbf{Err}^{dec}\|_\infty \leqslant N \cdot tE$.

Proof. Clearly, we can easily prove Lemma 5 using Lemma 4. Hence, we have that

$$||\mathbf{err}_i^{dec}|| = ||\mathbf{e}_i^T\mathbf{R} \cdot \mathbf{G}^{-1}(\mathbf{w}^T)|| \leqslant ||\mathbf{e}_i^T\mathbf{R}|| \cdot ||\mathbf{G}^{-1}(\mathbf{w}^T)||_1 \leqslant N \cdot E.$$

$$||\mathbf{Err}^{dec}|| = ||[\mathbf{e}_1, \cdots, \mathbf{e}_t]^T\mathbf{R}|| \cdot ||\mathbf{G}^{-1}(\mathbf{W}^T)||_1 \leqslant N \cdot tE.$$

Now one can observe that decryption works correctly as long as $||\mathbf{Err}^{dec}||_\infty \leqslant \frac{q}{8}$, i.e. $E < \frac{q}{8tN}$. We call this value $E_{\max} = \frac{q}{8tN}$.

Let $\mathbf{C}_1$ or $\mathbf{C}_2$ be two ciphertexts which are E_1 or E_2 noisy encryption of $\mathbf{M}_1, \mathbf{M}_2 \in \{0,1\}^{(n+t)\times(n+t)}$ under the $\mathbf{s}$ respectively, such that $\langle \mathbf{s}, \mathbf{C}_1\rangle = \mathbf{s}^T \cdot \mathbf{M}_1 \cdot \mathbf{G} + \mathbf{err}_1$ and $\langle \mathbf{s}, \mathbf{C}_2\rangle = \mathbf{s}^T \cdot \mathbf{M}_2 \cdot \mathbf{G} + \mathbf{err}_2$ with $||\mathbf{err}_1||_\infty \leqslant E_1$ and $||\mathbf{err}_2||_\infty \leqslant E_2$ by Lemma 4. Furthermore, if $\mathbf{C}_1$ or $\mathbf{C}_2$ is under the $\mathbf{S}$ respectively, then $\langle \mathbf{S}, \mathbf{C}_1\rangle = \mathbf{S}^T \cdot \mathbf{M}_1 \cdot \mathbf{G} + \mathbf{Err}_1$ and $\langle \mathbf{S}, \mathbf{C}_2\rangle = \mathbf{S}^T \cdot \mathbf{M}_2 \cdot \mathbf{G} + \mathbf{Err}_2$ with $||\mathbf{Err}_1||_\infty \leqslant tE_1$ and $||\mathbf{Err}_2||_\infty \leqslant tE_2$ by Lemma 4.

Appendix C.2.1 *Homomorphic Addition Analysis.*

Below, we analyze the homomorphic addition. More details are as follows:

Lemma 6. If a ciphertext $\mathbf{C}$ is designed to encrypt message $\mathbf{M} \in \{0,1\}^{(n+t)\times(n+t)}$ under a secret key matrix $\mathbf{S}$, then ciphertext addition results in ciphertext $\mathbf{C}^{add} = \mathbf{C}_1 + \mathbf{C}_2$ such that $\langle \mathbf{S}, \mathbf{C}^{add}\rangle = \mathbf{Err}^{add} + \mathbf{S}^T \cdot (\mathbf{M}_1 + \mathbf{M}_2) \cdot \mathbf{G}$, where $\mathbf{C}_1$, $\mathbf{C}_2$ are respectively designed to encrypt $\mathbf{M}_1, \mathbf{M}_2 \in \{0,1\}^{(n+t)\times(n+t)}$, $\mathbf{M}^{add} = \mathbf{M}_1 + \mathbf{M}_2$ and $\mathbf{Err}^{add} = \mathbf{Err}_1 + \mathbf{Err}_2$. Clearly, it is $t \cdot (E_1 + E_2)$-noisy.

Proof. The above statements for $\mathbf{C}^{add}$ follow by straightforward calculation.

Appendix C.2.2 *Homomorphic Multiplication Analysis.*

Before analyzing homomorphic multiplication, we note that $u_{i,i}\mathbf{b}_i^T - \mathbf{t}_i^T\mathbf{B}^T$ for each $i \in [t]$ is bounded by $|t \cdot \mathbf{e}_i^T|$. Hence, we have that

$$\mathbf{S}^T \cdot \left(\mathbf{M}_1\mathbf{A}^T\right) = \begin{pmatrix} u_{1,1} \cdot \mathbf{b}_1^T - \mathbf{t}_1^T \cdot \mathbf{B}^T \\ \vdots \\ u_{t,t} \cdot \mathbf{b}_t^T - \mathbf{t}_t^T \cdot \mathbf{B}^T \end{pmatrix} \tag{C1}$$

then there exists $\|\mathbf{S}^T \cdot \left(\mathbf{M}_1\mathbf{A}^T\mathbf{R}_2\right)\| \leqslant t \cdot \|\mathbf{e}_i^T\mathbf{R}_2\| \leqslant t \cdot E$.

Lemma 7. Let $\mathbf{S} \in \mathbb{Z}^{(n+t)\times t}$ be a secret key matrix. Let $\mathbf{C}_1$, $\mathbf{C}_2 \in \mathbb{Z}_q^{(m+1)\times N}$ be ciphertexts that encrypt message $\mathbf{M}_1$, $\mathbf{M}_2 \in \{0,1\}^{(n+t)\times(n+t)}$, respectively. Thus ciphertext multiplication results in ciphertext $\mathbf{C}^{mult} = \mathbf{C}_1 \cdot \mathbf{G}^{-1}(\mathbf{C}_2)$ such that $\mathbf{C}^{mult} = \mathbf{M}_1 \cdot \mathbf{M}_2 \cdot \mathbf{G} + \left(\mathbf{A}^T\mathbf{R}_1\mathbf{G}^{-1}(\mathbf{C}_2) + \mathbf{M}_1\mathbf{A}^T\mathbf{R}_2\right) \in \mathbb{Z}_q^{(n+t)\times N}$, then the ciphertext $\mathbf{C}^{mult}$ is $\left((Nt+t) \cdot E\right)$-noisy.

Proof. By Eq (C1), we have that

$$\langle \mathbf{S}, \mathbf{C}^{mult} \rangle = \mathbf{S}^T \cdot \Big(\mathbf{C}_1 \mathbf{G}^{-1}(\mathbf{C}_2) \Big) = \mathbf{S}^T \cdot \Big(\mathbf{M}_1 \mathbf{M}_2 \mathbf{G} + \big(\mathbf{A}^T \mathbf{R}_1 \mathbf{G}^{-1}(\mathbf{C}_2) \big) + \big(\mathbf{M}_1 \mathbf{A}^T \mathbf{R}_2 \big) \Big)$$

where we set $\mathbf{Err}^{mult} = \mathbf{S}^T \cdot \mathbf{A}^T \big(\mathbf{R}_1 \mathbf{G}^{-1}(\mathbf{C}_2) \big) + \mathbf{S}^T \cdot \big(\mathbf{M}_1 \mathbf{A}^T \mathbf{R}_2 \big)$.

Clearly, $\|\mathbf{S}^T \cdot \mathbf{A}^T\| = \|[\mathbf{e}_1, \cdots, \mathbf{e}_t]^T \cdot \mathbf{R}_1\| \leqslant tE_1$ by Lemma 4 and $\|\mathbf{S}^T \cdot \big(\mathbf{M}_1 \mathbf{A}^T \mathbf{R}_2 \big)\| \leqslant tE_2$ by Eq.(C1). We have $\mathbf{C}_2$ is an $(n+t) \times N$-dimension binary matrix ($\mathbf{G}^{-1} \in \mathbb{Z}_q^{N \times (n+t)}$). Hence, we have that

$$\begin{aligned} ||\mathbf{Err}^{mult}||_\infty \ &\leqslant \ ||tE_1 \cdot \mathbf{G}^{-1}(\mathbf{C}_2)|| + ||t \cdot E_2|| \ \leqslant \ tE_1 \cdot ||\mathbf{G}^{-1}(\mathbf{C}_2)||_\infty + tE_2 \\ &\leqslant \big(N \cdot tE_1 + tE_2 \big) \leqslant (Nt + t) \cdot E \end{aligned}$$

and the ciphertext $\mathbf{C}^{mult}$ is $\big((Nt+t) \cdot E\big)$-noisy.

The same calculation holds for NAND gates. Consider the evaluation of a Boolean circuit of depth L consisting of NAND gates. It takes as input fresh ciphertexts, i.e. E-noisy ciphertexts, and at each level the noise is multiplied by a factor of at most $(Nt+t)$, i.e. the norm of error elements is increased by a factor of at most $(Nt+t)$. Therefore, the error elements of final ciphertext have norm bounded by

$$E_{\text{final}} = (Nt+t)^L \cdot E.$$

To ensure correctness of decryption, we have that condition $(Nt+t)^L \cdot E_P < \lfloor \frac{q}{2} \rfloor / 4$ must hold, which is guaranteed by our choice of parameters.

Appendix C.3 IND-CPA Security Analysis of mFHE Scheme

Below we use the main result to prove the security of multi-bit and single-key FHE scheme.

Theorem 1 ([24]). Let $m > n \in \mathbb{N}$, let $q \in \mathbb{N}$ and let χ be a discrete Gaussian distribution on $\mathbb{Z}$ such that the (n, q, χ, m)-LWE problem is hard. Let t be an integer such that $t = \mathcal{O}(\log(n))$. Define two distributions $\mathcal{X}$ and $\mathcal{Y}$ as follows.

– $\mathcal{X}$ is the distribution on $m \times (t+n)$ matrices $[\mathbf{b}_1 = \mathbf{B}\mathbf{t}_1 + \mathbf{e}_1 \pmod q | \cdots | \mathbf{b}_t = \mathbf{B}\mathbf{t}_t + \mathbf{e}_t \pmod q | \mathbf{B}]$ where $\mathbf{B} \in \mathbb{Z}_q^{m \times n}$ is chosen uniformly at random and where, for all $1 \leqslant i \leqslant t$, $\mathbf{t}_i$ is sampled uniformly from $\mathbb{Z}_q^n$ and $\mathbf{e}_i$ is sampled from a discrete Gaussian distribution χ.

– $\mathcal{Y}$ is the uniform distribution on $\mathbb{Z}_q^{m \times (t+n)}$.

Then the two distributions $\mathcal{X}$ and $\mathcal{Y}$ are computationally indistinguishable.

Proof. Let $\mathcal{D}$ be a PPT adversary that can distinguish $\mathcal{X}$ from $\mathcal{Y}$ with non-negligible advantage. For $1 \leqslant i \leqslant t+1$ we introduce intermediate distributions $\mathcal{X}_i$ given by $[\mathbf{b}'_1 | \cdots | \mathbf{b}'_{i-1} | \mathbf{b}_i | \cdots | \mathbf{b}_t | \mathbf{B}]$, where $\mathbf{b}_i$ is as above and $\mathbf{b}'_i$ is uniformly chosen from $\mathbb{Z}_q^m$. Hence $\mathcal{X}_1 = \mathcal{X}$ and $\mathcal{X}_{t+1} = \mathcal{Y}$.

By assumption, $\mathcal{D}$ can distinguish $\mathcal{X}_1$ from $\mathcal{X}_{t+1}$ with noticeable advantage ϵ and so, by a standard hybrid argument, there is some i such that $\mathcal{D}$ can distinguish $\mathcal{X}_i$ from $\mathcal{X}_{i+1}$ with some noticeable advantage at least ϵ/t. Apparently $\mathcal{D}$ gives an LWE distinguisher: Given an LWE challenge $(\mathbf{B}, \mathbf{y})$ one samples $\mathbf{b}'_1, \ldots, \mathbf{b}'_{i-1}$ uniformly and samples $\mathbf{b}_{i+1}, \ldots, \mathbf{b}_t$ as specified above (so they are from an LWE distribution, all for different choices of the secret vector) and then calls $\mathcal{D}$ on $[\mathbf{b}'_1 | \cdots | \mathbf{b}'_{i-1} | \mathbf{y} | \mathbf{b}_{i+1} | \cdots | \mathbf{b}_t | \mathbf{B}]$. Therefore, no such distinguisher exists. Hence the theorem is proved.

We show the scheme is IND-CPA-secure based on the LWE assumption by using Theorem 1 to show that the scheme is indistinguishable from the original GSW [9] scheme.

Theorem 2. Let $\mathsf{params} = (n, q, \chi, m, t)$ be such that the $\mathsf{LWE}_{n,q,\sigma,m}$ assumption holds and $m = \mathcal{O}(n \log q)$. Then the mFHE scheme is IND-CPA-secure.

Proof. Given the fact that $\mathsf{LWE}_{n,q,\chi,m}$ assumption holds, we have that any PPT distinguisher $\mathcal{D}$ can distinguish the following two cases at most with negligible probability δ:

- **(0)** the uniform distribution (i.e., $\{(\mathbf{b}'_1, \cdots, \mathbf{b}'_t, \mathbf{B}) : \mathbf{B} \leftarrow \mathbb{Z}_q^{m \times n}, \mathbf{b}'_i \leftarrow \mathbb{Z}_q^{m \times 1}\}$ for $i \in [t]$).
- **(1)** $\mathcal{B}_{\mathbf{t},\chi}$ for a uniform random $\mathbf{t} \in \mathbb{Z}_q^n$ (i.e., $\{(\mathbf{b}_1, \cdots, \mathbf{b}_t, \mathbf{B}) : \mathbf{B} \leftarrow \mathbb{Z}_q^{m \times n}, \mathbf{t}_i \leftarrow \mathbb{Z}_q^{n \times 1}, \mathbf{e}_i \leftarrow \chi^{m \times 1}, \mathbf{b}_i = \mathbf{B} \cdot \mathbf{t}_i + \mathbf{e}_i \pmod q\}$ for $i \in [t]$).

Our goal here is to show that any PPT adversary $\mathcal{A}$ can break the IND-CPA-security of mFHE scheme at most negligible advantage. Next, we show the security via the following hybrids games.

– Game H0, this is a real IND-CPA game with adversary $\mathcal{A}$: More concretely, **i).** run the $\mathsf{KeyGen}(\cdot)$ to obtain the key pairs and send the public key pk to adversary $\mathcal{A}$; here the public key $\mathbf{A} = [\mathbf{b}_1, \cdots, \mathbf{b}_t, \mathbf{B}] \in \mathbb{Z}_q^{m \times (n+t)}$, where $\mathbf{b}_i = \mathbf{B}\mathbf{t}_i + \mathbf{e}_i \pmod q$ with $i \in [t]$; **ii).** then the adversary $\mathcal{A}$ returns a pair of messages $\mathbf{M}_0$ and $\mathbf{M}_1$; **iii).** after that, choose a uniform bit $b \in \{0, 1\}$, and send the adversary $\mathcal{A}$ the challenge ciphertext $\mathbf{C}_b \leftarrow \mathsf{Enc}(\mathsf{pk}, \mathbf{M}_b)$; note that here the ciphertext $\mathbf{C}_b = \mathbf{M}_b \mathbf{G} + \mathbf{A}^T \mathbf{R} \in \mathbb{Z}_q^{(n+t) \times N}$ for random matrix $\mathbf{R} \in \{0, 1\}^{(n+t) \times N}$; **iv).** finally, the adversary $\mathcal{A}$ returns a bit b'. The game outputs 1 if $b' = b$ and 0 otherwise. If $b' = b$ we say that $\mathcal{A}$ succeeds. We define advantage $\mathsf{Adv}_{\mathcal{A},\mathsf{H0}} = \Pr[b' = b]$.

– Game H1, this is a game with adversary $\mathcal{A}_1$. Note that game H1 is identical to game H0 except that we use $\mathbf{b}'_i$ which is sampled from $\mathbb{Z}_q^{m \times 1}$ uniformly to replace $\mathbf{b}_i = \mathbf{B}\mathbf{t}_i + \mathbf{e}_i \pmod q$, $i \in [t]$. Utilizing Theorem 1, we can see that the matrix $\mathbf{A} = [\mathbf{b}_1, \cdots, \mathbf{b}_t, \mathbf{B}] \in \mathbb{Z}_q^{m \times (n+t)}$ is computationally indistinguishable from a randomly chosen matrix $\widetilde{\mathbf{A}} = [\mathbf{b}'_1, \cdots, \mathbf{b}'_t, \mathbf{B}] \in \mathbb{Z}_q^{m \times (n+t)}$; the computational distance is $\delta \cdot t$, where δ is negligible and t is polynomial in λ. Now we can have the advantage $\mathsf{Adv}_{\mathcal{A},\mathsf{H0}} \leqslant \mathsf{Adv}_{\mathcal{A}_1,\mathsf{H1}} + \delta \cdot t$.

– Game H2, this is a game with adversary $\mathcal{A}_2$. Note that game H2 is identical to game H1 except that we use a uniform random $\mathbf{U} \in \mathbb{Z}_q^{(n+t)\times N}$ to replace the matrix $\widetilde{\mathbf{A}}^T\mathbf{R}$. Using the leftover hash lemma, we can see that $\widetilde{\mathbf{A}}^T\mathbf{R}$ is indistinguishable from uniform. The statistical distance is δ', where δ' is negligible in λ. Now we can have the advantage $\mathsf{Adv}_{\mathcal{A}_1,\mathsf{H1}} \leqslant \mathsf{Adv}_{\mathcal{A}_2,\mathsf{H2}} + \delta_1$.

We note that, in Game H2, all the elements of both the public key and the ciphertext are uniformly random and independent of the message. Hence, $\mathsf{Adv}_{\mathcal{A}_2,\mathsf{H2}} = \frac{1}{2}$.

In summary, we have $\mathsf{Adv}_{\mathcal{A},\mathsf{H0}} \leqslant 1/2 + \delta \cdot t + \delta' = 1/2 + \mathsf{negl}(\lambda)$. That means, adversary $\mathcal{A}$ can break the IND-CPA-security of mFHE scheme with at most negligible advantage. This completes the proof.

Appendix D Correctness Analysis of iLCP

In this section, we give an example to present our main idea. Suppose there exist two parties User 1 and User 2, then we denote the secret key matrix $\mathbf{S}^{(1)}$ and $\mathbf{S}^{(2)}$ respectively, in more detail: User 1's $\mathsf{pk}^{(1)} = \mathbf{A}^{(1)}$, $\mathsf{sk}^{(1)} = \mathbf{S}^{(1)}$ and $\mathbf{b}_i^{(1)} = \mathbf{B}\mathbf{t}_i^{(1)} + \mathbf{e}_i^{(1)}$, $i = 1, 2$

$$\mathbf{S}^{(1)} = [\mathbf{s}_1^{(1)}, \mathbf{s}_2^{(1)}]^T = [(1, 0 \mid -\mathbf{t}_1^1), (0, 1 \mid -\mathbf{t}_2^{(1)})]^T;\ \mathbf{A}^{(1)} = [\mathbf{b}_1^{(1)} \mid \mathbf{b}_2^{(1)} \mid \mathbf{B}];$$

User 2's $\mathsf{pk}^{(2)} = \mathbf{A}^{(2)}$, $\mathsf{sk}^{(2)} = \mathbf{S}^{(2)}$ and $\mathbf{b}_i^{(2)} = \mathbf{B}\mathbf{t}_i^{(2)} + \mathbf{e}_i^{(2)}$, $i = 1, 2$

$$\mathbf{S}^{(2)} = [\mathbf{s}_1^{(2)}, \mathbf{s}_2^{(2)}]^T = [(1, 0 \mid -\mathbf{t}_1^{(2)}), (0, 1 \mid -\mathbf{t}_2^{(2)})]^T; \mathbf{A}^{(2)} = [\mathbf{b}_1^{(2)} \mid \mathbf{b}_2^{(2)} \mid \mathbf{B}];$$

For the ciphertext $\mathbf{C}_{\#}^{(1)} = \mathbf{M}\widetilde{\mathbf{G}} + {\mathbf{A}^{(1)}}^T\widetilde{\mathbf{R}}$, where the plaintext matrix $\mathbf{M}$ is

$$\mathbf{M} = \left(\begin{array}{c:c} m_{1,1}, 0 & 0, 0 \\ \hdashline 0, m_{2,2} & 0, 0 \\ \hdashline 0, 0 & 1, 0 \\ 0, 0 & 0, 1 \end{array}\right) \in \{0,1\}^{n\times n},$$

Suppose we want to use $\mathbf{S}^{(2)}$ to decrypt the ciphertext $\mathbf{C}_{\#}^{(1)}$, we have that:

$$\begin{aligned} \langle \mathbf{S}^{(2)}, \mathbf{C}_{\#}^{(1)} \rangle &= \left(\mathbf{S}^{(2)}\right)^T \cdot \left(\mathbf{A}^{(1)}\right)^T \cdot \widetilde{\mathbf{R}} + \left(\mathbf{S}^{(2)}\right)^T \cdot \mathbf{M}\widetilde{\mathbf{G}} \\ &= \left(\begin{array}{c} \mathbf{b}_1^{(1)} - \mathbf{b}_1^{(2)} \\ \hdashline \mathbf{b}_2^{(1)} - \mathbf{b}_2^{(2)} \end{array}\right)^T \cdot \widetilde{\mathbf{R}} + \left(\begin{array}{c} \mathbf{e}_1^{(2)} \\ \hdashline \mathbf{e}_2^{(2)} \end{array}\right) \cdot \widetilde{\mathbf{R}} + \left(\begin{array}{c:c} 1, 0 & -\mathbf{t}_1^{(2)} \\ \hdashline 0, 1 & -\mathbf{t}_2^{(2)} \end{array}\right) \cdot \mathbf{M}\widetilde{\mathbf{G}} \end{aligned} \tag{D1}$$

Consider the inner product $\mathbf{S}^{(1)}$ and $\mathbf{X}$, then we have that

$$\langle \mathbf{S}^{(1)}, \mathbf{X}^T \rangle = \left(\begin{array}{c} \mathbf{b}_1^{(2)} - \mathbf{b}_1^{(1)} \\ \hdashline \mathbf{b}_2^{(2)} - \mathbf{b}_2^{(1)} \end{array}\right)^T \cdot \widetilde{\mathbf{R}} + \mathcal{T} \pmod q, \tag{D2}$$

where $\mathcal{T} := [\mathbf{e}_1, \cdots, \mathbf{e}_t]$. Putting Eq.(D1)and Eq.(D2) together, if there exists the equation

$$\langle \mathbf{S}^{(2)}, \mathbf{C}_{\#}^{(1)} \rangle + \langle \mathbf{S}^{(1)}, \mathbf{X} \rangle = (\mathbf{S}^{(2)})^T \cdot \mathbf{M}\widetilde{\mathbf{G}} + \underline{\mathcal{T} + \left(\begin{array}{c} \mathbf{e}_1^{(2)} \\ \hdashline \mathbf{e}_2^{(2)} \end{array}\right) \cdot \widetilde{\mathbf{R}}} = (\mathbf{S}^{(2)})^T \cdot \mathbf{M}\widetilde{\mathbf{G}} + \mathbf{Noise} \pmod q$$

then the bound of **Noise** is $\|\mathbf{Noise}\| \leqslant \big(tn + t\big) \cdot E$ (in above described case, the bound of **Noise** is $(2n + 2) \cdot E$).

Remark 6. Below, we use an example to explain our multi-key scheme. We define the "combined secret key matrix" as $\widehat{\mathbf{S}} = [\mathbf{S}^{(1)}; \mathbf{S}^{(2)}] \in \mathbb{Z}_q^{2n\times t}$. Hence, the expanded ciphertext is as follows:

$$\widehat{\mathbf{C}} = \left(\begin{array}{c:c} \mathbf{C}_{\#} & \mathbf{X} \\ \hdashline \mathbf{0} & \mathbf{C}_{\#} \end{array}\right) \in \mathbb{Z}_q^{2n\times 2t}$$

Hence, we have that $\langle \widehat{\mathbf{S}}, \widehat{\mathbf{C}} \rangle = [\mathbf{S}^{(1)} \cdot \mathbf{C}, \mathbf{S}^{(1)} \cdot \mathbf{X} + \mathbf{S}^{(2)} \cdot \mathbf{C}] \approx [\mathbf{M}\mathbf{S}^{(1)}\widetilde{\mathbf{G}}, \mathbf{M}\mathbf{S}^{(2)}\widetilde{\mathbf{G}}] = \mathbf{M} \cdot \widehat{\mathbf{S}} \cdot \widetilde{\mathbf{G}}$ by Lemma 5 and Eq. (D1), (D2).

Appendix E Correctness Analysis of mMFHE Scheme

Appendix E.1 Correctness of Expansion:

Suppose there exist N-parties, then we consider N-pairs key $\big((\mathsf{pk}^{(1)}, \mathsf{sk}^{(1)} = \mathbf{S}^{(1)}), \cdots, (\mathsf{pk}^{(N)}, \mathsf{sk}^{(N)} = \mathbf{S}^{(N)})\big)$ such that $\{(\mathsf{pk}^{(i)}, \mathsf{sk}^{(i)}) \leftarrow \mathsf{mMFHE.KeyGen(params)}\}_{i\in[N]}$. For any length message $\mathbf{M}$ and any $i \in [N]$ we have a ciphertext $\mathbf{C} \leftarrow \mathsf{mMFHE.Enc}(\mathsf{pk}^{(i)}, \mathbf{M})$ under the i-th key and the corresponding expanded ciphertext $\widehat{\mathbf{C}} \leftarrow \mathsf{mMFHE.Expand}((\mathsf{pk}^{(1)}, \cdots, \mathsf{pk}^{(N)}), i, \mathbf{C})$. Let $\widehat{\mathbf{S}} = [\mathbf{S}^{(1)}, \cdots, \mathbf{S}^{(N)}]$, then

$$\begin{aligned} \widehat{\mathbf{S}} \cdot \widehat{\mathbf{C}} &= [(\mathbf{S}^{(1)}\mathbf{X}_1 + \mathbf{S}^{(1)}\mathbf{C}), \cdots, \mathbf{S}^{(i)}\mathbf{C}, \cdots, (\mathbf{S}^{(N)}\mathbf{X}_N + \mathbf{S}^{(N)}\mathbf{C})] \\ &= \widehat{\mathbf{S}}\widehat{\mathbf{M}}\mathbf{G} + [\mathbf{Noise}^{(1)}, \cdots, t(\mathbf{e}_1^{(i)}, \cdots, \mathbf{e}_t^{(i)}) \cdot \widetilde{\mathbf{R}}, \cdots, \mathbf{Noise}^{(N)}], \end{aligned}$$

Hence, $\|\mathbf{Noise}^{(1)}, \cdots, \mathbf{Noise}^{(N)}\|_\infty \leqslant \|\mathbf{Noise}^{(i)}\| \leqslant ((tn + t) \cdot E)$.

Appendix E.2 Correctness of Evaluation:

Let $\widehat{\mathbf{C}}_1, \cdots, \widehat{\mathbf{C}}_t$ be expanded ciphertexts corresponding to bit $\mu_1, \cdots, \mu_t$ so that, $\widehat{\mathbf{S}}^T\widehat{\mathbf{C}} = \widehat{\mathbf{S}}^T\widehat{\mathbf{M}}\widetilde{\mathbf{G}}+\mathbf{Noise}$, where $\|\mathbf{Noise}^{(i)}\| \leqslant ((nt+t)\cdot E)$. If $\widehat{\mathbf{C}}$ is the result of homomorphic operation, then for L-depth circuit we have $\|\mathbf{Noise}\| \leqslant (tN+1)^L \cdot (nt+t)\cdot E$.

Appendix E.3 Correctness of Threshold Decryption:

There is no difference with the correctness of mFHE scheme except that mMFHE with N-pairs key.

Lemma 8. If $\widehat{\mathbf{C}}$ is an evaluated ciphertext encrypting multi-bit $\mathbf{U}$ and the secret keys are $\widehat{\mathbf{S}} = [\mathbf{S}^1, \cdots, \mathbf{S}^N]$, by the analysis used for non-threshold correctness, then we have that

$$\langle \widehat{\mathbf{S}}, \widehat{\mathbf{C}} \rangle = \sum_{i\in[N]} (\mathbf{S}^{(i)})^T \cdot \widehat{\mathbf{C}}^{(i)} = \widehat{\mathbf{S}}^T \cdot \widehat{\mathbf{M}}\widehat{\mathbf{G}} + [\mathbf{Noise}^{(1)}, \cdots, \mathbf{Noise}^{(N)}],$$

where each $\mathbf{Noise}$ has that $\|\mathbf{Noise}\|_\infty \leqslant ((nt+t)\cdot E)$, and each level multiplies the noise by a factor of at most $(tN+1)$, then for L-depth circuit we have $\|\mathbf{Noise}\| \leqslant (tN+1)^L \cdot (nt+t)\cdot E$.

Therefore if the partial decryptions $\mathbf{p}_i$ are computed as specified we have:

$$\begin{aligned}
\sum_{i\in[N]} \mathbf{P}^{(i)} &= \sum_{i\in[N]} \mathcal{R}^{(i)} + \sum_{i\in[N]} (\mathbf{e}^{(i)})^{sm} = (\sum_{i\in[N]} (\mathbf{S}^{(i)})^T \cdot \widehat{\mathbf{C}}^{(i)}) \cdot \widehat{\mathbf{G}}^{-1}(\widehat{\mathbf{W}}^T) + \mathbf{e}^{sm} \\
&= (\widehat{\mathbf{S}}^T\widehat{\mathbf{M}}\widehat{\mathbf{G}} + [\mathbf{Noise}^{(1)}, \cdots, \mathbf{Noise}^{(N)}]) \cdot \widehat{\mathbf{G}}^{-1}(\widehat{\mathbf{W}}^T) + \mathbf{e}^{sm} \\
&= \lceil \frac{q}{2} \rceil \cdot \sum_{i\in[N]} \mathbf{U}^{(i)} + [err^{(1)}, \cdots, err^{(N)}] + \mathbf{e}^{sm},
\end{aligned}$$

Hence, output $\mathbf{U} = \frac{1}{N} \cdot \left\| \left\lceil \frac{\mathbf{P}}{q/2} \right\rfloor \right\| \approx \left\| \left\lceil \frac{\frac{1}{N}\cdot\lceil \frac{q}{2} \rceil \cdot \sum_{i\in[N]} \mathbf{U}^{(i)}}{q/2} \right\rfloor \right\|$. For the sake of readability, we set $\mathbf{err} = [err^{(1)}, \cdots, err^{(N)}]$ and each $err = \mathbf{Noise} \cdot \widehat{\mathbf{G}}^{-1}(\widehat{\mathbf{W}}^T)$ has norm $|err| \leqslant |\mathbf{Noise}| \cdot |\widehat{\mathbf{G}}^{-1}(\widehat{\mathbf{W}}^T)| \leqslant ((nt+t)\cdot E)$. Hence, in this setting, since $q = 2^{\omega(d\lambda \log \lambda)} \cdot B$ we have that $|\mathbf{err} + \mathbf{e}^{sm}| < \frac{q}{4}$ i.e. each $|err + e^{sm}| < \frac{q}{4N}$. We stress that, $\mathbf{e}^{sm} = \sum_{i\in[N]} \sum_{j\in[t]} \sum_{k\in[t]} (e_j^{(i)})^{sm} = \sum_{i\in[N]} (\mathbf{e}^{(i)})^{sm}$ has norm $|\mathbf{e}^{sm}| \leqslant tN \cdot B_{smdg}^{dec} = 2^{O(d\lambda \log \lambda)} B_\chi$ where $(\mathbf{e}^{(i)})^{sm} = \sum_{k=1}^{t} (e_j^{(i)})^{sm} = [(e_1^{(i)})^{sm}, \cdots, (e_t^{(i)})^{sm}]$.

Remark 7. For $\mathbf{S}^{(i)} \in \mathbb{Z}_q^{n\times t}$, and $\widehat{\mathbf{C}}^{(i)} \in \mathbb{Z}_q^{n\times mN}$, and $\widehat{\mathbf{G}}^{(i)} \in \mathbb{Z}_q^{n\times mN}$, there exists: $\langle \mathbf{S}^{(i)}, \widehat{\mathbf{C}}^{(i)} \rangle = (\mathbf{S}^{(i)})^T \cdot \widehat{\mathbf{C}}^{(i)} = (\mathbf{S}^{(i)})^T \cdot (\mathbf{M} \cdot \widehat{\mathbf{G}}^{(i)}) + \mathbf{Noise}^{(i)}$.

References

1 Ronald L Rivest, Len Adleman, and Michael L Dertouzos. On data banks and privacy homomorphisms. *Foundations of secure computation*, 1978, 4(11):169–180.

2 Craig Gentry. Fully homomorphic encryption using ideal lattices. *In: Proceedings of ACM STOC 2009*, 2009. 169–178.

3 Zvika Brakerski and Vinod Vaikuntanathan. Fully homomorphic encryption from ring-LWE and security for key dependent messages. *In: Proceedings of CRYPTO 2011*, 2011, 6841: 505–524.

4 Zvika Brakerski and Vinod Vaikuntanathan. Efficient fully homomorphic encryption from (standard) LWE. *In: Proceedings of IEEE FOCS 2011*, 2011. 97–106.

5 Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. (Leveled) fully homomorphic encryption without bootstrapping. *In: Proceedings of ACM ITCS 2012*, 2012. 309–325.

6 Shai Halevi and Victor Shoup. Algorithms in HElib. *In: Proceedings of CRYPTO 2014, Part I*, 2014, 8616:554–571.

7 Shai Halevi and Victor Shoup. Bootstrapping for HElib. *In: Proceedings of EUROCRYPT 2015, Part I*, 2015, 9056: 641–670.

8 Zvika Brakerski. Fully homomorphic encryption without modulus switching from classical GapSVP. *In: Proceedings of CRYPTO 2012*, 2012,7417:868–886.

9 Craig Gentry, Amit Sahai, and Brent Waters. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. *In: Proceedings of CRYPTO 2013, Part I*, 2013, 8042:75–92.

10 Adriana López-Alt, Eran Tromer, and Vinod Vaikuntanathan. On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption. *In: Proceedings of ACM STOC 2012*, 2012. 1219–1234.

11 Michael Clear and Ciaran McGoldrick. Multi-identity and multi-key leveled FHE from learning with errors. *In: Proceedings of CRYPTO 2015, Part II*, 2015, 9216:630–656.

12 Pratyay Mukherjee and Daniel Wichs. Two round multiparty computation via multi-key FHE. *In: Proceedings of EUROCRYPT 2016, Part II*, 2016, 9666: 735–763.

13 Zvika Brakerski and Renen Perlman. Lattice-based fully dynamic multi-key FHE with short ciphertexts. *In: Proceedings of CRYPTO 2016, Part I*, 2016, 9814: 190–213.

14 Chris Peikert and Sina Shiehian. Multi-key FHE from LWE, revisited. *In: Proceedings of TCC 2016*, 2016. 217–238.

15 Ryo Hiromasa, Masayuki Abe, and Tatsuaki Okamoto. Packing messages and optimizing bootstrapping in GSW-FHE. *In: Proceedings of PKC 2015*, 2015, 9020:699–715.

16 Vadim Lyubashevsky. Lattice signatures without trapdoors. *In: Proceedings of EUROCRYPT 2012*, 2012, 7237: 738–755.

17 Gilad Asharov, Abhishek Jain, Adriana López-Alt, Eran Tromer, Vinod Vaikuntanathan, and Daniel Wichs. Multiparty computation with low communication, computation and interaction via threshold FHE. *In: Proceedings of EUROCRYPT 2012*, 2012, 7237:483–501.

18 Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *In: Proceedings of ACM STOC 2005*, 2005. 84–93.

19 Chris Peikert. Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. *In: Proceedings of ACM STOC 2009*, 2009. 333–342.

20 Chris Peikert and Brent Waters. Lossy trapdoor functions and their applications. *In: Proceedings of ACM STOC 2008*, 2008. 187–196.

21 Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. *In: Proceedings of EUROCRYPT 2012*, 2012, 7237: 700–718.

22 Jacob Alperin-Sheriff and Chris Peikert. Faster bootstrapping with polynomial error. *In: Proceedings of CRYPTO 2014, Part I*, 2014, 8616: 297–314.

23 Jonathan Katz and Yehuda Lindell. *Introduction to modern cryptography*. CRC press, 2014.

24 Zengpeng Li, Steven D. Galbraith, and Chunguang Ma. Preventing adaptive key recovery attacks on the GSW levelled homomorphic encryption scheme. *In: Proceedings of Provsec 2016*, 2016. 373–383.

25 Zengpeng Li, Chunguang Ma, Eduardo Morais, and Gang Du. Multi-bit leveled homomorphic encryption via dual.lwe-based. *In: Proceedings of Inscrypt 2016*, 2016. 221–242.

26 Zvika Brakerski and Vinod Vaikuntanathan. Lattice-based FHE as secure as PKE. *In: Proceedings of ACM ITCS 2014*, 2014. 1–12.

27 Léo Ducas and Daniele Micciancio. FHEW: Bootstrapping homomorphic encryption in less than a second. *In: Proceedings of EUROCRYPT 2015, Part I*, 2015, 9056: 617–640.