Secure and Efficient Multi-Key FHE Scheme Supporting Multi-bit Messages from LWE Preserving Non-Interactive Decryption

Chinmoy Biswas*, Ratna Dutta
Department of Mathematics, Indian Institute of Technology Kharagpur, Kharagpur -721302, India
Email:chinmoy88biswas@gmail.com, ratna@maths.iitkgp.ernet.in

Abstract

We consider *multi-key fully homomorphic encryption* (multi-key FHE) which is the richest variant of fully homomorphic encryption (FHE) that allows complex computation on encrypted data under different keys. Since its introduction by Lopez-Alt, Tromer and Vaikuntanathan in 2012, numerous proposals have been presented yielding various improvements in security and efficiency. However, most of these multi-key FHE schemes encrypt a single-bit message. Constructing a multi-key FHE scheme encrypting multi-bit messages have been notoriously difficult without loosing efficiency for homomorphic evaluation and ciphertext extension under additional keys. In this work, we study multi-key FHE that can encrypt multi-bit messages. Motivated by the goals of improving the efficiency, we propose a new construction with non-interactive decryption and security against chosen-plaintext attack (IND-CPA) from the standard *learning with errors* (LWE) assumption. We consider a binary matrix as plaintext instead of a single-bit. Our approach supports efficient homomorphic matrix addition and multiplication. Another interesting feature is that our technique of extending a ciphertext under additional keys yields significant reduction in the computational overhead. More interestingly, when contrasted with the previous multi-key FHE schemes for multi-bit messages, our candidates exhibits favorable results in the length of the secret key, public key and ciphertext preserving non-interactive decryption.
Keywords: lattice based cryptosystem, multi-key fully homomorphic encryption, learning with errors, multi-bit messages

1 Introduction

In the last few decades of technological advancement, we are fast approaching to a new digital era in which we store our data on powerful servers called the cloud and delegate our expensive computations to the servers. The data stored in the cloud could be vulnerable to interference by the cloud provider or even by other cloud clients. The data often contains sensitive information including personal conversations, medical information, organizational secrets and many more. It is desirable for users to encrypt their data before storing it in the cloud. Traditional public key encryption schemes are insufficient for such tasks, thus generating interests in the study of homomorphic encryption. Homomorphic encryption allows an untrusted server to perform computation

*Corresponding author. E-mail: chinmoy88biswas@gmail.com

on encrypted data without the data being compromised. This, in turn, facilitates outsourcing computation to untrusted servers.

1.1 Fully Homomorphic Encryption (FHE)

The emergences of FHE over the last decade has revolutionized cryptography. FHE makes it possible to compute an encryption of $f(m)$ for some arbitrary function f on the message m without knowing the private key and can only be decrypted by the party holding the private key. More concretely, a FHE scheme is a public key encryption scheme $(\mathsf{Keygen}, \mathsf{Enc}, \mathsf{Dec})$ with additional algorithm Eval that allows homomorphic operations on ciphertexts. For any $(pk, sk) \leftarrow \mathsf{Keygen}(1^\lambda)$, a function f and two ciphertexts $\mathbf{c} = \mathsf{Enc}(pk, m), \mathbf{c}^{'} = \mathsf{Enc}(pk, m^{'})$ encrypted under the same public key pk, the Eval algorithm takes $(pk, f, \mathbf{c}, \mathbf{c}^{'})$ as input and returns a homomorphically evaluated ciphertext $\mathbf{c}^*$ such that $\mathsf{Dec}(sk, \mathbf{c}^*) = f(m, m^{'}) = f(\mathsf{Dec}(sk, \mathbf{c}), \mathsf{Dec}(sk, \mathbf{c}^{'}))$ where sk is the secret key corresponding to the public key pk. This remarkable feature of computing arbitrary functions on encrypted data has enabled numerous application in cryptography over the years. Following the breakthrough result of Gentry et al. Gentry (2009), many improved variants Brakerski (2012), Brakerski et al. (2014), Brakerski and Vaikuntanathan (2011, 2014), Gentry et al. (2013), Smart and Vercauteren (2010) have appeared in the literature. However, all these constructions of FHE are focused on encrypting a single-bit message. Once we have an FHE for single-bit encryption, we can effortlessly compose it to obtain an FHE for multi-bit encryption. However, this results in huge overheads and weak performance. Recently, many exciting new constructions Brakerski et al. (2013), Hiromasa et al. (2016), Li et al. (2016) were proposed to design FHE schemes for encrypting multi-bit messages with various efficiency improvements. All the aforementioned works deal their security mainly from hard problems over lattices.

1.2 Multi-key FHE

An important limitation of FHE is that it requires all of the data to be encrypted under the same key in order to perform homomorphic evaluation. The notion of multi-key FHE was proposed as a generalization of FHE to the multi-party setting. Multi-key FHE is applicable to the setting where users upload their data in the cloud encrypted under different keys and wish the cloud server to process these encrypted data to compute an encryption of some joint function of their collected data without knowing the data. The users coordinate at the time of decryption and run a multi-party computation protocol to recover the joint function. Examples of such joint function include computing joint statistical information on user's databases, locating common files in their collections, running a computational factor to reach a decision based on their combine data without leaking anything except the final decision, and many more. More formally, each party in multi-key FHE schemes can individually choose a key pair by running the key generation algorithm and use it to encrypt its own private input. One of the appealing features of multi-key FHE compared to FHE is that it allows a public evaluator to perform homomorphic computation on ciphertexts encrypted even under disjoint sets of keys. Apart from Keygen, Enc, Eval, dec, a multi-key FHE has an additional ciphertext extension algorithm Ext. The extension algorithm Ext takes as input $pk_1, pk_2, \ldots, pk_l$, a ciphertext $\mathbf{C}_1 = \mathsf{Enc}(pk_1, m_1)$ and produces an extended ciphertext $\widehat{\mathbf{C}}_1 = \mathsf{Enc}(\mathsf{PK}, m_1)$ under the common public key $\mathsf{PK} = (pk_1, pk_2, \ldots, pk_l)$.

Naturally, decrypting the resulting multi-key ciphertext requires one to know all the secret keys for the involved parties. Let $\widehat{\mathbf{C}}_1 = \mathsf{Enc}(\mathsf{PK}, m_1)$ and $\widehat{\mathbf{C}}_2 = \mathsf{Enc}(\mathsf{PK}, m_2)$ be the extended (or evaluated) ciphertext of $\mathbf{C}_1 = \mathsf{Enc}(pk_1, m_1)$ and $\mathbf{C}_2 = \mathsf{Enc}(pk_2, m_2)$ respectively where $\mathsf{PK} = (pk_1, pk_2)$ is the common public key. The Eval algorithm takes as input a function f, extended (or evaluated) ciphertext $\widehat{\mathbf{C}}_1 = \mathsf{Enc}(\mathsf{PK}, m_1)$ and $\widehat{\mathbf{C}}_2 = \mathsf{Enc}(\mathsf{PK}, m_2)$ and returns a ciphertext $\mathbf{C}^*$ such that $\mathsf{Dec}(sk_1, sk_2, \mathbf{C}^*) = f(\mathsf{Dec}(sk_1, \mathbf{C}_1), \mathsf{Dec}(sk_2, \mathbf{C}_2))$ where sk_1, sk_2 are the secret keys corresponding to the public keys pk_1, pk_2 respectively. Thus decryption of the evaluated ciphertext $\mathbf{C}^*$ requires the secret keys sk_1, sk_2 of both the parties involved in the computation. Consequently, parties coordinate only at the time of decryption in multi-key FHE. The key efficiency requirement is that the size of the evaluated ciphertext must be independent of the size of the function. Over the past few years, there have been a wave of multi-key FHE constructions Brakerski and Perlman (2016), Clear and McGoldrick (2015), Li et al. (2018), López-Alt et al. (2012), Mukherjee and Wichs (2016), Peikert and Shiehian (2016). Multi-key FHE has interesting applications in privacy preserving scenarios apart from designing many other cryptographic primitives.

1.3 (Multi-identity) Identity-based FHE (IBFHE)

Given the rapidly expanding set of FHE applications, there has been a trend to design FHE in identity-based setting during the last few years. To provide a fundamental solution to the problem of too large public key size in FHE, IBFHE was introduced. However obtaining FHE in the identity-based setting seems quite a tricky problem in spite of its potential for comparative simplicity. Gentry, Sahai and Waters Gentry et al. (2013) constructed the first leveled IBFHE based on the learning with errors (LWE) assumption. The level restriction comes from the ability to evaluate certain bounded depth circuit. They built a compiler that can compile all lattice-based IBE schemes Agrawal et al. (2010a,b), Cash et al. (2012), Gentry et al. (2008) satisfying certain properties to leveled IBFHE. In Clear and McGoldrick (2014), Clear and McGoldrick proposed a non-leveled IBFHE scheme based on indistinguishable obfuscation ($i\mathcal{O}$). All IBFHE constructions mentioned above use the single identity based setting in the sense that homomorphic addition and multiplication can be done on the ciphertexts encrypted under the same identity. Clear and McGoldrick Clear and McGoldrick (2015) extends this to multi-identity setting and obtained the first leveled multi-identity IBFHE in the random oracle model under the hardness of LWE. Lately, Pal et al. Pal and Dutta (2020) propose a non-leveled multi-identity IBFHE without using $i\mathcal{O}$.

1.4 (Multi-attribute) Attribute-based FHE (ABFHE)

Another flavor of FHE that connect in attribute-based setting is ABFHE which is analogous to IBFHE where the underlying encryption scheme is attribute-based encryption (ABE) to realize fine grained access control. Gentry, Sahai and Waters Gentry et al. (2013) constructed the first leveled ABFHE scheme from the LWE problem that allows evaluating the circuits of limited depths. The only known way to obtain a non-leveled ABFHE is through $i\mathcal{O}$. Recently, Clear and McGoldrick Clear and McGoldrick (2016) proposed a non-leveled ABFHE without using $i\mathcal{O}$ with a limitation of supporting only a bounded number of users. The multi-attribute ABFHE scheme is an extension of ABFHE in multi-attribute setting. The first multi-attribute ABFHE was proposed by Clear et al. Clear and McGoldrick (2014) which uses $i\mathcal{O}$ and is non-leveled. Recently,

Pal et al. Pal and Dutta (2020) propose a multi-attribute ABFHE using witness pseudorandom function instead of $i\mathcal{O}$.

1.5 Our Result

The one round (or non-interactive) decryption ability of multi-key FHE has been thought of as a tool to construct two round multi-party computation protocols with low communication since the work of Mukherjee and Wichs Mukherjee and Wichs (2016). A line of recent work focuss on constructing multi-key FHE schemes offering improvements in security and efficiency Brakerski and Perlman (2016), Clear and McGoldrick (2015), Kim et al. (2018), Li et al. (2018), Peikert and Shiehian (2016). Out of these work, only the multi-key FHE of Li et al. (2018) supports multi-bit encryption. The challenge lies in the requirement of efficient homomorphic computation and ciphertext extension under additional keys. This somewhat unsatisfactory state-of-affairs motivates our search for a lattice-based instantiation of multi-key FHE for multi-bit messages. Inspired by the work of Mukherjee and Wichs (2016), we construct a new multi-key FHE for multi-bit messages with one round decryption in common reference string model. We set each entry of a plaintext matrix as a message slot and and directly encrypt the random binary matrix selected during the encryption process. As exhibited in TABLE 1 and TABLE 2, our construction offers several strong advantages over the existing approaches of designing FHE for multi-bit messages Brakerski et al. (2013), Hiromasa et al. (2016), Li et al. (2016, 2018), including small key size, ciphertext size. A bit more preciously, we show how to efficiently performs homomorphic evaluation and extend a ciphertext under an additional key in order to reduce the storage as well as computational overhead. Another interesting feature of our candidate is that it supports non-interactive decryption as decryption completes in single round. We support the conjectured security of our candidate by analysis and prove that the scheme achieves indistinguishability under chosen plaintext attack (IND-CPA) under a standard cryptograpic assumption namely LWE problem. Specifically, we prove the following theorems.

THEOREM 1. *(Informal) If the LWE problem is hard, then our multi-key FHE scheme for encrypting multi-bit messages is indistinguishable under chosen plaintext attack (IND-CPA).*

THEOREM 2. *(Informal) The extended ciphertext* $\widehat{\mathbf{C}}_i$ *of a ciphertext* $\mathbf{C}_i$ *satisfies* $\widehat{\mathsf{SK}}.\widehat{\mathbf{C}}_i = \lfloor \frac{q}{2} \rfloor \mathsf{Con}_k(\mathbf{M}_i) + \mathbf{E}_{\widehat{\mathbf{C}}_i} \in \mathbb{Z}_q^{kr \times kr}$ *with error* $|\mathbf{E}_{\widehat{\mathbf{C}}_i}|_\infty \leq r(1+l)\beta$ *where* $l = \lceil \log q \rceil$, β *is a bound of an error distribution,* $\widehat{\mathsf{SK}} \in \mathbb{Z}_q^{kr \times k(n+r)}$ *is the secret,* $\mathsf{Con}_k(\mathbf{M}_i) = (\mathbf{M}'_i, \mathbf{O}, \ldots, \mathbf{O})^T \in \{0,1\}^{kr \times kr}$ *with* $\mathbf{M}'_i = (\mathbf{M}_i, \mathbf{M}_i, \ldots, \mathbf{M}_i) \in \{0,1\}^{r \times kr}$ *and* $\mathbf{C}_i \in \mathbb{Z}_q^{(n+r) \times r}$ *is a* $r\beta$*-noisy encryption of a message matrix* $\mathbf{M}_i \in \{0,1\}^{r \times r}$.

Comparison to existing approaches: We briefly discuss our resulting multi-key FHE for multi-bit encryption in reference to existing approaches.

– The scheme of Li, Ma and Zhou Li et al. (2018) places r message bits in the diagonal of a $r \times r$ plaintext matrix and extends the multi-key multi-bit FHE plaintext size to $(n+r) \times (n+r)$ by adding rows of 0's and columns of 0's, featuring the ciphertext size $O((n+r)^2 l^2)$ bits where n is the lattice dimension, $l = \lceil \log q \rceil$ and q is integer modulus. On the contrary, to encrypt r message bits, we arrange each entry of a plaintext matrix of order $\sqrt{r} \times \sqrt{r}$ as a message slot and extend the plaintext

Table 1: Comparison table of LWE-based multi-bit FHE and multi-key FHE scheme for r message bits

Scheme	$\|pk\|$	$\|sk\|$	$\|ct\|$	FHE/multi-key FHE
Brakerski et al. (2013)	$O((n+r)rl^2)$	$O((n+r)rl^2)$	$O((n+r)l)$	FHE
Hiromasa et al. (2016)	$O((n+r)nl^3)$	$O((n+r)rl^2)$	$O((n+r)^2l)$	FHE
Li et al. (2016)	$O(n^2l^3)$	$O(nrl^3)$	$O(n(nl+r)l^2)$	FHE
Li et al. (2018)	$O((n+r)nl^2)$	$O((n+r)rl)$	$O((n+r)^2l^2)$	multi-key FHE
[Our]	$O((n+\sqrt{r})\sqrt{r}l)$	$O((n+\sqrt{r})\sqrt{r}l)$	$O((n+\sqrt{r})\sqrt{r}l)$	multi-key FHE

Here $|pk|, |sk|, |ct|$ indicate the size of the public key pk, secret key sk and ciphertex ct respectively, n is the underlying lattice dimension and $l = \lceil \log q \rceil$ where $\mathbb{Z}_q$ is the underlying field.

Table 2: Comparison table of multi-key FHE schemes for single bit message

Scheme	$\|pk\|$	$\|sk\|$	$\|ct\|$	$\|ct_{\mathsf{ext}}\|$
Clear and McGoldrick (2015)	$O(n^2l^2)$	$O(nl^3)$	$O(n^2l^4)$	$O(n^2k^2l^5)$
Mukherjee and Wichs (2016)	$O(n^2l^2)$	$O(nl)$	$O(n^2l^2)$	$O(n^2k^2l^2)$
Brakerski and Perlman (2016)	$O(n^3l^2)$	$O(nl)$	$O(nl)$	$O(n^2k^2l^2)$
Peikert and Shiehian (2016) Sch1	$O(nl^2)$	$O(nl)$	$O(n^3l^4)$	$O(n^3kl^4)$
Peikert and Shiehian (2016) Sch2	$O(n^4l^4)$	$O(nl)$	$O(n^2l^2)$	$O(n^2k^2l^2)$
Kim et al. (2018)	$O(n^2l^2)$	$O(nl)$	$O(n^2l^2)$	$O(n^2k^2l^2)$
Li et al. (2018)	$O(n^2l^2)$	$O(nl)$	$O(n^2l^2)$	$O(n^2k^2l^2)$
[our]	$O(nl)$	$O(nl)$	$O(nl)$	$O(nk^2l)$

Here $|pk|, |sk|, |ct|, |ct_{\mathsf{ext}}|$ indicate the size of public key pk, secret key sk, ciphertex ct and extended ciphertext ct_{ext} respectively, n is the underlying lattice dimension, $l = \lceil \log q \rceil$ where $\mathbb{Z}_q$ is the underlying field and k indicates the number of involved parties.

matrix to size $(n + \sqrt{r}) \times \sqrt{r}$ by adding rows of 0's, offering the ciphertext of size $O((n + \sqrt{r})\sqrt{r}l)$ bits. One of the appealing feature of our design compared to Li et al. (2018) is that it requires $O((n+\sqrt{r})\sqrt{r}l)$ bits for both the public key and secret key while the public key and secret key size in Li et al. (2018) are $O((n + r)nl^2)$ bits and $O((n + r)rl)$ bits respectively.

– Although, LWE based FHE schemes of Brakerski et al. (2013), Hiromasa et al. (2016), Li et al. (2016) supports encryption of multi-bit messages, they are not in multi-key setting. We compare our construction with these schemes in TABLE 1 for encryption of r message bits which exhibits that our scheme works favorably better tan these schemes in terms of secret key, public key and ciphertext size. We furthermore compare the size of public key, secret key, ciphertext and expanded ciphertext with the existing multi-key FHE schemes for a single bit message. In all the said aspects, our scheme achieves significant improvement over the existing multi-key FHE schemes as demonstrated in TABLE 2.

– The security of our multi-key FHE for multi-bit message is based on the standard LWE assumption.

1.6 Applications of multi-key FHE

Multi-key FHE has enabled numerous powerful cryptographic applications including two round multi party computation (MPC) protocol Mukherjee and Wichs (2016), homomorphic secret sharing Boyle et al. (2016, 2017), spooky encryption Dodis et al. (2016), obfuscation and functional encryption combiners Ananth et al. (2016, 2017),

multi-party obfuscation Halevi et al. (2017), homomorphic time-lock puzzle Brakerski et al. (2019), Malavolta and Thyagarajan (2019) and ad-hoc multi-party functional encryption Agrawal et al. (2020). To illustrate the potential of our techniques, we consider two example use-cases of multi-key FHE for encrypting multi-bit messages.

• *Malware sharing:* Suppose a famous company wants to know which other companies have investigated the similar malware that they have recently detected without disclosing how many malware they have found so far. An ideal solution should allow each company to maintain a dataset of the malware they have detected and store it in encrypted form to the cloud server. Whenever a company detects a new malware, it encrypts the malware with its public key and sends it to the cloud server. The cloud server collects the encrypted dataset on a daily basis and evaluates a comparison function which scans the datasets for similar malware samples using some algorithm allowing its computational complexity to depend on the size of the function and sends the compact multi-key encrypted results to the companies. Then the companies get the intended result by interactive decryption.

• *Medical application:* In a medical system, there is a number of scenarios in which computation on patient data is desirable maintaining the patient's privacy. Multi-key FHE provides a potential solution for that. Here the patients are the data owner who upload their encrypted medical data under their respective public keys to the service provider in order to maintain privacy. There could be an expanding range of medical data including blood pressure, heart rate greater than 80 per minute, weight or blood sugar to predict the likelihood of certain physical condition occurring and many more. The service provider computes a function on these encrypted medical data and sends the evaluated result to the patients retaining the communication efficiency. The patients then engage in a multi-party computation protocol and can learn the result over the inputs of the patients. The main benefit of this is to make real-time health analysis based on data from various source while maintaining the privacy of each source.

1.7 Technical Overview

We now proceed to describe a high level overview of the techniques that we develop in order to achieve our result. Our multi-key FHE for multi-bit message construction $\mathsf{MMFHE} = (\mathsf{Setup}, \mathsf{Keygen}, \mathsf{Enc}, \mathsf{Ext}, \mathsf{Eval}, \mathsf{Dec})$ encrypts $\tilde{r} = r^2$ bits message. On input the security parameter, a trusted party picks a random matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, a β bounded error distribution χ over $\mathbb{Z}_q$ and set the common parameter as $params = \{n, q, \chi, \beta, r, N, \mathbf{A}\}$ where q is the modulus, n is the lattice dimension and $m = n\lceil \log q \rceil$. Each party utilizes $params$ to set its public key and secret key as $pk = \mathbf{B} = \begin{pmatrix} \mathbf{S}^{'}\mathbf{A} + \mathbf{E} \\ \mathbf{A} \end{pmatrix} \in \mathbb{Z}_q^{(n+r)\times r}$ and $sk = \mathbf{S} = \begin{pmatrix} \mathbf{I}_r | -\mathbf{S}^{'} \end{pmatrix} \in \mathbb{Z}_q^{r \times (n+r)}$ respectively where $\mathbf{S}^{'} \in \mathbb{Z}_q^{r \times n}$ is chosen randomly and $\mathbf{E} \in \chi^{r \times r}$ is an error matrix. Note that $pk = \mathbf{B}$ and $sk = \mathbf{S}$ satisfy the relation $\mathbf{SB} = \mathbf{E}$. To encrypt an $\tilde{r} = r^2$ bits message, an encrypter constructs a message matrix $\mathbf{M} \in \{0,1\}^{r \times r}$ by setting each message bit as an entry of $\mathbf{M}$ and encrypt $\mathbf{M}$ under the public key $pk = \mathbf{B}$ to generate the ciphertext $\mathbf{C} = \lceil \frac{q}{2} \rceil \begin{pmatrix} \mathbf{M} \\ \mathbf{O} \end{pmatrix} + \mathbf{BR} \in \mathbb{Z}_q^{(n+r)\times r}$ and auxiliary information $\mathsf{Aux} = \mathbf{RG} + \mathbf{E}^{'} \mod q \in \mathbb{Z}_q^{r \times rl}$ where $\mathbf{R} \in \{0,1\}^{r \times r}$, $\mathbf{G}$ is the r-th order standard gadget matrix and $\mathbf{E}^{'} \in \chi^{r \times rl}$ is an error matrix. Here, ciphertext $\mathbf{C}$ satisfies the relation $\mathbf{SC} = \lfloor \frac{q}{2} \rfloor \mathbf{M} + \mathbf{E_C}$ where $\mathbf{E_C}$ is an error matrix. The challenge comes in ad-

dressing the ciphertext extension and evaluation. As we discussed below, the technical road towards achieving these goals is not straightforward. To aid these operations, the auxiliary information Aux generated in the encryption procedure is utilized retaining its key virtue of communication. For simplicity, let us consider the extension and evaluation algorithm for two parties.

To extend a ciphertext $\mathbf{C}_1$ (encrypted using the public key pk_1) of the first party under an additional public key pk_2 of the second party, the server computes $\mathbf{X}_2 = [\mathsf{Aux}_1\mathbf{C}_1^{-1}(pk_2 - pk_1)^T]^T \in \mathbb{Z}_q^{(n+r)\times r}$ utilizing the auxiliary information Aux_1 of first party, pk_1, pk_2 . The server then sets the extended ciphertext $\widehat{\mathbf{C}}_1$ as $\widehat{\mathbf{C}}_1 = \begin{pmatrix} \mathbf{C}_1 & \mathbf{X}_2 \\ \mathbf{O} & \mathbf{C}_1 \end{pmatrix} \in \mathbb{Z}_q^{2(n+r)\times 2r}$. This extended ciphertext $\widehat{\mathbf{C}}_1$ satisfies $\widehat{\mathbf{S}}_1\widehat{\mathbf{C}}_1 = \lfloor \frac{q}{2} \rfloor \widehat{\mathbf{M}}_1 + \widehat{\mathbf{E}}_{\mathbf{C}}$ where $\widehat{\mathbf{M}}_1 = \begin{pmatrix} \mathbf{M}_1 & \mathbf{M}_1 \\ \mathbf{O} & \mathbf{O} \end{pmatrix}$ and $\widehat{\mathbf{S}}_1 = \begin{pmatrix} \mathbf{S}_1 & \mathbf{S}_2 \\ \mathbf{O} & \mathbf{O} \end{pmatrix}$ and $\widehat{\mathbf{E}}_{\mathbf{C}}$ is an error matrix. The message $\mathbf{M}_1$ can be recovered from $\widehat{\mathbf{M}}_1$.

To run the evaluation algorithm on two ciphertexts $\mathbf{C}_1 \in \mathbb{Z}_q^{(n+r)\times r}$ and $\mathbf{C}_2 \in \mathbb{Z}_q^{(n+r)\times r}$ under the public keys $pk_1 \in \mathbb{Z}_q^{(n+r)\times r}$ and $pk_2 \in \mathbb{Z}_q^{(n+r)\times r}$ respectively, the server extends the ciphertext $\mathbf{C}_1$ under pk_2 to produce the extended ciphertext $\widehat{\mathbf{C}}_1 \in \mathbb{Z}_q^{2(n+r)\times 2r}$ and $\mathbf{C}_2$ under pk_1 to produce the extended ciphertext $\widehat{\mathbf{C}}_2 \in \mathbb{Z}_q^{2(n+r)\times 2r}$. Observe that $\widehat{\mathbf{C}}_1$ and $\widehat{\mathbf{C}}_2$ are ciphertexts under the common public key $\mathsf{PK} = (pk_1, pk_2)$, corresponding common secret key $\mathsf{SK} = \begin{pmatrix} sk_1 & sk_2 \\ \mathbf{O} & \mathbf{O} \end{pmatrix}$ and satisfy the relation $(\mathsf{SK})\widehat{\mathbf{C}}_i = \lfloor \frac{q}{2} \rfloor \widehat{\mathbf{M}}_i + \widehat{\mathbf{E}}_i$. Running the subroutine to add two ciphertexts $\mathbf{C}_1$ and $\mathbf{C}_2$, the server first extends the ciphertexts $\mathbf{C}_1$ and $\mathbf{C}_2$ to $\widehat{\mathbf{C}}_1$ and $\widehat{\mathbf{C}}_2$ respectively and then computes $\mathbf{C}_{\mathsf{add}} = \widehat{\mathbf{C}}_1 + \widehat{\mathbf{C}}_2 \in \mathbb{Z}_q^{2(n+r)\times 2r}$. The ciphertext $\mathbf{C}_{\mathsf{add}}$ is a ciphertext under the common public key PK and satisfies the relation $(\mathsf{SK})\mathbf{C}_{\mathsf{add}} = \lfloor \frac{q}{2} \rfloor (\widehat{\mathbf{M}}_1 + \widehat{\mathbf{M}}_2) + \mathbf{E}_{\mathsf{add}}$ where $\mathbf{E}_{\mathsf{add}}$ is an error matrix. Similarly, executing the subroutine to multiply two ciphertexts $\mathbf{C}_1$ and $\mathbf{C}_2$, the server extends these ciphertexts to $\widehat{\mathbf{C}}_1$ and $\widehat{\mathbf{C}}_2$ respectively as above and computes $\mathbf{C}_{\mathsf{mult}} = (\widehat{\mathbf{C}}_1 \otimes \widehat{\mathbf{C}}_2)_{\mathsf{sc}}$ where $\otimes$ denotes the matrix tensor product and sc indicates shrink column operation whereby the number of columns of a matrix is reduced by the rearrangement of its columns. The ciphertext $\mathbf{C}_{\mathsf{mult}}$ is a ciphertext under the secret key $\mathbf{S}_{\mathsf{mult}} = (\mathbf{S}_1 \otimes \mathbf{S}_2)_{\mathsf{sr}}$ and satisfies $\mathbf{S}_{\mathsf{mult}}\mathbf{C}_{\mathsf{mult}} = \lfloor \frac{q}{2} \rceil \widehat{\mathbf{M}}_1\widehat{\mathbf{M}}_2 + \mathbf{E}_{\mathsf{mult}}$ where $\mathbf{E}_{\mathsf{mult}}$ is an error matrix and sr denotes shrink row operation whereby the number of rows of a matrix is reduced by the rearrangement of its rows. From $\widehat{\mathbf{M}}_1\widehat{\mathbf{M}}_2$, one can get $\mathbf{M}_1\mathbf{M}_2$. To decrypt a fresh ciphertext $\mathbf{C}$ under the secret key $\mathbf{S}$, the party computes $\lfloor \frac{q}{2} \rfloor \mathbf{SC}$ and recovers the message in the same way as in Regev Regev (2009). Moreover, to decrypt an evaluated (extended) ciphertext $\widehat{\mathbf{C}} \in \mathbb{Z}_q^{k(n+r)\times kr}$ under the public key $\mathsf{PK} = (pk_1, pk_2, \ldots, pk_k) \in \mathbb{Z}_q^{(n+r)\times kr}$, the k users with their respective secret keys $sk_1, sk_2, \ldots, sk_k$ run a threshold decryption protocol among themselves and recover the message. To conclude, we highlight that our scheme supports one round or non-interactive decryption, is proven to be secure under the LWE assumption and therefore enjoys post-quantum security.

2 Literature Survey on Multi-key FHE

The study of multi-key FHE is a recent but active research area. There is a long line of work which studied construction of multi-key FHE schemes and huge literature on them. We provide a brief overview of multi-key FHE schemes in chronological order to this work to illustrate the difference of the working procedures among the several variants of multi-key FHE schemes.

In 2012, **Lopez-Alt, Tromer and Vaikuntanathan** López-Alt et al. (2012) introduced the notion of multi-key FHE along with a candidate construction based on N-th degree Truncated polynomial Ring Units (NTRU) encryption. They considered the error distribution χ on a ring $R_q = R/qR$ where $R = \mathbb{Z}[x]/(x^n + 1)$ and n is a power of 2. The public and secret keys are $f = 2f^{'} + 1 \in \chi$ and $h = 2gf^{-1}$ respectively where $f^{'}$ is a short polynomial and $g \in \chi$. The ciphertext for message $m \in \{0, 1\}$ is $\mathbf{C} = hs + 2e + m$ for $s, e \in \chi$. For two different parties, addition and multiplication can be done by $\mathbf{C}_1 + \mathbf{C}_2$ and $\mathbf{C}_1.\mathbf{C}_2$ respectively which are decrypted through the joint secret key $f_1 f_2$. Similarly, to decrypt a ciphertext $\mathbf{C}_1^2\mathbf{C}_2$, the decrypter requires $f_1^2 f_2$. Thus the decrypter should have the knowledge of the evaluated circuit to correctly decrypt which is undesirable. To overcome this problem, the relinearization (or key switching) technique is used which is expensive. A somewhat homomorphic encryption (SWHE) scheme is first constructed under the *decisional small polynomial ratio* (DSPR) assumption and the ring LWE assumption which is converted to a multi-key leveled homomorphic encryption scheme that can evaluate circuit of maximum depth D with N keys provided $ND \approx \log q$. SWHE is a variant of homomorphic encryption where ciphertext length increases with every homomorphic operation. A multi-key FHE is designed from this multi-key leveled homomorphic encryption for a bounded number of parties using bootstrapping. The decryption requires interactive threshold decryption procedure that involved running a generic multi-party computation protocol.

Clear and McGoldrick Clear and McGoldrick (2015) in 2015 constructed the first multi-key FHE by designing a multi-identity FHE based on LWE. Their construction requires to fix the maximum number of independent keys supported for evaluation. The construction is instantiated by using the identity-based encryption of Gentry, Peikert and Vaikuntanathan (GPV-IBE)Gentry et al. (2008). The secret key for an identity id is set as $\mathbf{v}_{id} = \mathsf{PowersOf2}(\mathbf{s}_{id}) \in \mathbb{Z}_q^N$ where $\mathbf{s}_{id} \in \mathbb{Z}_q^{m^{'}}$ is chosen randomly, $N = m^{'}.\lceil \log q \rceil$ and $\mathsf{PowersOf2}(\mathbf{x}) = (\mathbf{x}, 2\mathbf{x}, 2^2\mathbf{x}, \ldots, 2^{\lfloor \log q \rfloor - 1}\mathbf{x})$. The ciphertext for message $\mu \in \{0, 1\}$ under the identity id is $\mathbf{C}_{id} \in \mathbb{Z}_q^{N \times N}$ which satisfies $\mathbf{C}_{id}\mathbf{v}_{id} = \mu\mathbf{v}_{id} + \mathbf{e}$ where $\mathbf{e}$ is an error vector. Let $\mathbf{C}_{id_1}$, $\mathbf{C}_{id_2} \in \mathbb{Z}_q^{N \times N}$ be two ciphertexts under the public keys pk_{id_1} and pk_{id_2} respectively. To extend the ciphertext $\mathbf{C}_{id_1}$ under the additional public key pk_{id_2}, the extended ciphertext $\widehat{\mathbf{C}}_{id_1}$ is constructed as $\widehat{\mathbf{C}}_{id_1} = \begin{pmatrix} \mathbf{C}_{id_1} & \mathbf{X} \\ \mathbf{O} & \mathbf{Y} \end{pmatrix}$ where the matrices $\mathbf{X}, \mathbf{Y} \in \{0, 1\}^{N \times N}$ satisfy $\mathbf{X}\mathbf{v}_{id_1} + \mathbf{Y}\mathbf{v}_{id_2} = \mu\mathbf{v}_{id_2} + \mathbf{e}$. A masking scheme is used to produce an auxiliary information $\mathcal{U} \in \{0, 1\}^*$ that enables the evaluator to generate the matrices $\mathbf{X}$ and $\mathbf{Y}$. The homomorphic addition and multiplication for two ciphertexts $\mathbf{C}_{id_1}$ and $\mathbf{C}_{id_2}$ are $\widehat{\mathbf{C}}_{id_1} + \widehat{\mathbf{C}}_{id_2}$ and $\widehat{\mathbf{C}}_{id_1}.\widehat{\mathbf{C}}_{id_2}$ respectively where $\widehat{\mathbf{C}}_{id_1}$ and $\widehat{\mathbf{C}}_{id_2}$ are the extended ciphertexts of $\mathbf{C}_{id_1}$ and $\mathbf{C}_{id_2}$ under the common public key $\mathsf{PK} = (pk_{id_1}, pk_{id_2})$.

In 2016, **Mukherjee and Wichs** Mukherjee and Wichs (2016) further simplified the

scheme of Clear and McGoldrick Clear and McGoldrick (2015) to facilitate two round multi-party computation protocol based on the standard LWE assumption. Their construction requires initial setup process to sample common parameters that must be used by each party to compute its key. More concretely, the scheme sets a common random matrix $\mathbf{B} \in \mathbb{Z}_q^{(n-1)\times m}$, selects $\mathbf{s} \in \mathbf{Z}_q^{(n-1)}$ and sets $\mathbf{b} = \mathbf{sB}+\mathbf{e}$ where $\mathbf{e} \in \chi^m$ is an error vector. The secret key $\mathbf{t} = (-\mathbf{s}, 1) \in \mathbb{Z}_q^n$ and the public key $\mathbf{A} = \begin{pmatrix} \mathbf{B} \\ \mathbf{b} \end{pmatrix} \in \mathbb{Z}_q^{n\times m}$ satisfy $\mathbf{tA} = \mathbf{e}$. To encrypt a message $\mu \in \{0,1\}$ under the public key $\mathbf{A}$, the ciphertext is generated as $\mathbf{C} = \mathbf{AR} + \mu\mathbf{G} \in \mathbb{Z}_q^{n\times m}$ where $\mathbf{R} \in \{0,1\}^{m\times m}$ and $\mathbf{G} = \mathbf{I}_n \otimes \mathbf{g}$ is the gadget matrix with $\mathbf{g} = (1, 2, 2^2, \ldots, 2^{(l-1)})$ as the gadget vector, $l = \lceil \log q \rceil$. Here $\otimes$ indicates the Kronecker (or tensor) product for matrices (or vectors). The ciphertext $\mathbf{C}$ and secret key $\mathbf{t}$ satisfy the relation $\mathbf{tC} = \mathbf{e}^{'} + \mu\mathbf{tG}$ where $\mathbf{e}^{'} = \mathbf{eR}$. The addition and multiplication of two ciphertexts under the same public key is computed as $\mathbf{C}^{+} = \mathbf{C}_1 + \mathbf{C}_2 \in \mathbb{Z}_q^{n\times m}$ and $\mathbf{C}^{\times} = \mathbf{C}_1\mathbf{G}^{-1}(\mathbf{C}_2) \in \mathbb{Z}_q^{n\times m}$ respectively. To define the addition and multiplication of ciphertexts under different public keys, the ciphertexts are extended to new ciphertexts under a common public key using a masking scheme. The masking scheme produces $\mathbf{C}_l, \mathcal{U}_l$ for party l where $\mathbf{C}_l = \mathbf{A}_l\mathbf{R}_l + \mu_l\mathbf{G}$ and $\mathcal{U}_l$ consists of the encryption of all the entries in the matrix $\mathbf{R}_l \in \{0,1\}^{m\times m}$. Utilising this $\mathcal{U}_l$ and public key $pk_i (i \neq l)$, the party l can generate the auxiliary information to extend its ciphertext under the common public key. For decrypting the evaluated ciphertext, a one round non-interactive threshold decryption protocol is utilized.

Brakerski and Perlman Brakerski and Perlman (2016) in 2016, proposed a multi-key FHE for an unbounded number of parties using a common setup. Their construction is dynamic (or multi-hop) for keys as evaluator can use an evaluated ciphertext as an input to another homomorphic computation under additional key. A multi-key FHE is static (or single-hop) for keys when an evaluated ciphertext can't be utilized for further homomorphic computation under additional keys. A static multi-key homomorphic encryption scheme is constructed first. The secret and public keys are respectively $sk = \mathbf{t} = (-\mathbf{s}, 1) \in \mathbb{Z}_q^n$ and $pk = \mathbf{A} = \begin{bmatrix} \mathbf{B} \\ \mathbf{b} \end{bmatrix} \in \mathbb{Z}_q^{n\times m}$ where $\mathbf{s} \in \mathbb{Z}_q^{n-1}$, $\mathbf{B} \in \mathbb{Z}_q^{(n-1)\times m}$ is a common random matrix, $\mathbf{b} = \mathbf{sB} + \mathbf{e} \in \mathbb{Z}_q^m$, $\mathbf{e} \in \chi^m$ and χ is the Gaussian distribution. The ciphertext is generated as $\mathbf{C} = \mathbf{AR} + \mu\mathbf{G}$ where $\mathbf{R} \in \{0,1\}^{m\times m}, \mu \in \{0,1\}$ is a message and $\mathbf{G} = \mathbf{I}_n \otimes \mathbf{g}$ is the standard gadget matrix. The ciphertext extension algorithm is similar to that in Mukherjee and Wichs Mukherjee and Wichs (2016). The last column of the evaluated (or extended) ciphertext is utilized. To reduce the decryption complexity, a one round (or non-interactive) threshold decryption protocol is run on input $\hat{\mathbf{t}} = (\mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_N)$ and $\hat{\mathbf{c}} = \widehat{\mathbf{C}}\mathbf{G}^{-1}(2^{(l-1)}\mathbf{u}_{nN})$ where $\widehat{\mathbf{C}}$ is the evaluated ciphertext, $l = \lceil \log q \rceil$ and $\mathbf{u}_{nN}$ is the vector with nN-th place 1 and other places 0. This static multi-key homomorphic encryption scheme is then made bootstappable by considering a small fragment of the ciphertext as $\mathbf{c} = \mathbf{Ar} + \mu \cdot 2^{l-1} \cdot \mathbf{u}_n$ and public key $pk^{'} = (\mathbf{A}, \overrightarrow{\mathcal{S}})$ where $\overrightarrow{\mathcal{S}}[i]$ is the encryption of t_i for $1 \leq i \leq n$, $sk = \mathbf{t} = (t_1, \ldots, t_i, \ldots, t_n)$ and $\mathbf{r}$ is the last column of the matrix $\mathbf{R} \in \{0,1\}^{m\times m}$. For evaluation, the NAND circuit is converted to a branching program which is then evaluated homomorphically. Informally, a branching program P is a deterministic program and defined by a directed acyclic graph which takes $\mathbf{x} \in \{0,1\}^n$ as input and induces a computational path from a distinguished initial node to terminal node whose label determines the output $P(\mathbf{x})$. The scheme supports $O(N)$ ciphertext expansion and $O(N)$ space complexity for a homomorphic

operation where N is the number of parties in the computation.

Peikert and Shiehian Peikert and Shiehian (2016) in 2016 built two LWE-based leveled multi-key FHE in dynamic setting with one round decryption in common reference string model. The ciphertext length of their first construction is large. This construction uses the symmetric key variant of Gentry, Sahai and Waters (GSW) encryption. The ciphertext corresponding to a message $\mu \in \{0,1\}$ and a secret key $sk = \mathbf{t} = (-\bar{\mathbf{t}}, 1) \in \mathbb{Z}_q^n$ is $\mathbf{C} = \overline{\mathbf{C}} + \mu(\mathbf{I}_n \otimes \mathbf{g}) \in \mathbb{Z}_q^{n \times m}$ where $\mathbf{g} = (1, 2, \ldots, 2^{l-1})$ is the gadget vector, $l = \lceil \log q \rceil$ and $\overline{\mathbf{C}}$ satisfies $\mathbf{t}\overline{\mathbf{C}} = \mathbf{e}$ with $\mathbf{e}$ as an error vector. Additionally, the scheme uses a homomorphic commitment $\mathbf{F} = \mathbf{AR} + \mu(\mathbf{I}_n \otimes \mathbf{g}) \in \mathbb{Z}_q^{n \times m}$ to the message μ where $\mathbf{R} \in \{0,1\}^{m \times m}$ is a random matrix and a special encryption $\mathbf{D} = \overline{\mathbf{D}} + \mathbf{R} \otimes \mathbf{g}^T \otimes \mathbf{e}^T \in \mathbb{Z}_q^{nml \times nl}$ satisfying $(\mathbf{I}_{ml} \otimes \mathbf{t}) \otimes \overline{\mathbf{D}} = \mathbf{E}'$ with $\mathbf{E}'$ as an error matrix. Let $\mathbf{C} \in \mathbb{Z}_q^{kn \times km}$ be an extended (or evaluated) ciphertext under k secret keys $\mathbf{t}' = (\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_k) \in \mathbb{Z}_q^{kn}$. To extend $\mathbf{C} \in \mathbb{Z}_q^{kn \times km}$ under a new secret key $\mathbf{t}^* \in \mathbb{Z}_q^n$. the extended ciphertext is set as $\widehat{\mathbf{C}} = \begin{pmatrix} \mathbf{C} & \mathbf{X} \\ \mathbf{O} & \mathbf{F} \end{pmatrix} \in \mathbb{Z}_q^{(k+1)n \times (k+1)m}$ where $\mathbf{X}$ satisfies $\mathbf{t}'\mathbf{X} + \mathbf{t}^*\mathbf{F} = \mu(\mathbf{t}^* \otimes \mathbf{g}), \mathbf{t}' \in \mathbb{Z}_q^{kn}$. The extended cipertext $\widehat{\mathbf{C}}$ remains a GSW ciphertext satisfying the relation $\hat{\mathbf{t}}\widehat{\mathbf{C}} = \hat{\mathbf{e}} + \mu\hat{\mathbf{t}}\widehat{\mathbf{G}}$ where $\hat{\mathbf{t}} = (\mathbf{t}', \mathbf{t}^*)$ with $\hat{\mathbf{e}}$ as an error vector and $\widehat{\mathbf{G}}$ is the extended gadget matrix. The homomorphic operations on these ciphertexts follow similarly from the homomorphic operations on GSW ciphertext.

The ciphertext in their second construction comprises of only a GSW ciphertext without any homomorphic commitment or special encryption, yielding small ciphertext size and supports the standard GSW homomorphic operation. The public key, on the other hand, contains a commitment $\mathbf{P} \in \mathbb{Z}_q^{n \times n^2 l}$ to its secret key $\mathbf{t} \in \mathbb{Z}_q^n$ where $\mathbf{P} = \mathbf{AR} + \mathbf{I}_n \otimes \mathbf{t} \otimes \mathbf{g}$, $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ is a common random matrix and $\mathbf{R} \in \{0,1\}^{m \times n^2 l}$. Additionally, the public key contains $\mathbf{D} = \bar{\mathbf{D}} + \mathbf{R} \otimes \mathbf{g}^T \otimes \mathbf{e}^T \in \mathbb{Z}_q^{nml \times nl}$ satisfying $\mathbf{t}\bar{\mathbf{D}} = \mathbf{e}''$ which is an encryption of the matrix $\mathbf{R}$ under the secret key $\mathbf{t}$ and $\mathbf{e}''$ as an error vector. If $\mathbf{C} \in \mathbb{Z}_q^{nk \times mk}$ is an extended (or evaluated) ciphertext under k secret key $\mathbf{t}' = (\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_k)$, then $\mathbf{C}$ can be extended under an additional secret key $\mathbf{t}^* \in \mathbb{Z}_q^n$ by setting $\widehat{\mathbf{C}} = \begin{pmatrix} \mathbf{C} & \mathbf{X} \\ \mathbf{O} & \mathbf{X}^* \end{pmatrix} \in \mathbb{Z}_q^{(k+1)n \times (k+1)m}$ where $\mathbf{X}$ and $\mathbf{X}^*$ satisfy $\mathbf{t}'\mathbf{X} + \mathbf{t}^*\mathbf{X}^* \approx \mu(\mathbf{t}^* \otimes \mathbf{g})$.

In 2018, **Kim, Lee and Park** Kim et al. (2018) proposed a technique to improve the scheme of Mukherjee and Wichs Mukherjee and Wichs (2016) that avoids the use of initial setup retaining non-interactive decryption ability. In this construction, the secret key is $\mathbf{t} = (-\mathbf{s}, 1) \in \mathbb{Z}_q^n$ and public key is $\mathbf{A} = \begin{pmatrix} \mathbf{B} \\ \mathbf{b} \end{pmatrix}$ where $\mathbf{b} = \mathbf{sB} + \mathbf{e}$ with $\mathbf{e}$ as an error vector and $\mathbf{B}$ is chosen by the user independently. The ciphertext generation, homomorphic addition, multiplication and decryption follow the same procedures as in Mukherjee and Wichs Mukherjee and Wichs (2016). To extend a ciphertext under an additional key, an auxiliary information is generated utilizing the masking scheme of Mukherjee and Wichs (2016) with a modified form of homomorphic linear combination. This yields one more round in the multi-party computation protocol derived from this multi-key FHE in contrast with the multi-key FHE of Mukherjee and Wichs Mukherjee and Wichs (2016) results in a two round multi-party computation protocol handling the random matrix needed for encryption and producing auxiliary information during extension of the ciphertext.

Li, Ma and Jhou Li et al. (2018) in 2018 designed a multi-key FHE for encrypting multi-bit message from LWE with one round decryption in the common random string model. The public key is $pk = \mathbf{A} = [\mathbf{b}_1|\mathbf{b}_2|\cdots|\mathbf{b}_t|\mathbf{B}] \in \mathbb{Z}_q^{m\times n}$ where $\mathbf{B} \in \mathbb{Z}_q^{m\times(n-t)}$ is a common random matrix chosen by a trusted party, $\mathbf{b}_i = \mathbf{B}\mathbf{t}_i + \mathbf{e}_i \in \mathbb{Z}_q^m, \mathbf{t}_i = (t_{i,1}, t_{i,2}, \ldots, t_{i,(n-t)}) \in \mathbb{Z}_q^{(n-t)}$ and $\mathbf{e}_i$ is an error vector. The secret key is $sk = \mathbf{S} = [\mathbf{s}_1|\mathbf{s}_2|\ldots|\mathbf{s}_t] \in \mathbb{Z}_q^{n\times t}$ where $\mathbf{s}_j = [0, 0, \ldots, 0, 1, 0, \ldots, 0, 0|-\mathbf{t}_j^T] \in \mathbb{Z}_q^n$. The public key $pk = \mathbf{A}$ and the secret key $sk = \mathbf{S}$ satisfies the relation $\mathbf{AS} = \mathbf{E} = [\mathbf{e}_1|\mathbf{e}_2|\cdots|\mathbf{e}_t]$. To encrypt t messages $\mu_i \in \{0, 1\}$, set the message matrix $\mathbf{U} = \mathsf{diag}(\mu_1, \mu_2, \ldots, \mu_t) \in \{0, 1\}^{t\times t}$ and compute the plaintext as $\mathbf{M} = \begin{pmatrix} \mathbf{U}_{t\times t} & \vdots & \mathbf{O}_{t\times(n-t)} \\ \mathbf{O}_{n\times t} & \vdots & \mathbf{I}_{(n-t)} \end{pmatrix} \in \{0, 1\}^{n\times n}$. The ciphertext is $\mathbf{C} = \mathbf{MG} + \mathbf{A}^T\mathbf{R} \in \mathbb{Z}_q^{n\times m}$ and the encoded information is $\mathbf{C}_{en} = \mathbf{RG} \in \mathbb{Z}_q^{m\times ml}$ where $n = ml, \mathbf{G} = \mathbf{g}^T \otimes \mathbf{I}_n, \mathbf{g} = (1, 2, 2^2, \ldots, 2^{l-1}), l = \lfloor \log q \rfloor$ and $\mathbf{R} \in \{0, 1\}^{m\times m}$. The encoded information $\mathbf{C}_{en}$ is used to extend the ciphertext $\mathbf{C}$ under additional keys. The ciphertext $\mathbf{C}$ and the secret key $\mathbf{S}$ satisfy the relation $< \mathbf{S}, \mathbf{C} >= \mathbf{S}^T\mathbf{C} = \mathbf{S}^T\mathbf{MG} + (\mathbf{AS})^T\mathbf{R} = \mathbf{S}^T\mathbf{MG} + \mathbf{E}'$ where $\mathbf{E}' = \mathbf{ER}$ is the error matrix. Let $\mathbf{C}_i$ be a ciphertext under the public key $pk_i = \mathbf{A}_i = (\mathbf{b}_{i,1}|\mathbf{b}_{i,2}|\cdots|\mathbf{b}_{i,t}|\mathbf{B})$ for a message $\mathbf{M}_i$ and $\mathbf{C}_{en_i} = \mathbf{R}_i\mathbf{G}(\mod q) \in \mathbb{Z}_q^{m\times n}$ be the encoded information. Let the corresponding secret key be $sk_i = \mathbf{S}_i = [\mathbf{s}_{i,1}|\mathbf{s}_{i,2}|\cdots|\mathbf{s}_{i,t}] \in \mathbb{Z}_q^{n\times t}$. The ciphertext $\mathbf{C}_i$ is extended to $\widehat{\mathbf{C}}_i$ under an additional public key $pk_j = \mathbf{A}_j = (\mathbf{b}_{j,1}|\mathbf{b}_{j,2}|\cdots|\mathbf{b}_{j,t}|\mathbf{B})$, by setting $\widehat{\mathbf{C}}_i$ as $\widehat{\mathbf{C}}_i = \begin{pmatrix} \mathbf{C}_i & \mathbf{X}^{(j)} \\ \mathbf{O} & \mathbf{C}_i \end{pmatrix}$ where $\mathbf{X}^{(j)} = \mathbf{A}_j + \mathbf{C}_{en_i}.\mathbf{G}^{-1}(pk_i - pk_j)$. Then $\widehat{\mathbf{S}}_i = (\mathbf{S}_i, \mathbf{S}_j)$ and $\widehat{\mathbf{C}}_i$ satisfies $< \widehat{\mathbf{S}}_i, \widehat{\mathbf{C}}_i >= \widehat{\mathbf{S}}_i^T\widehat{\mathbf{C}}_i = \widehat{\mathbf{E}}_i + \widehat{\mathbf{S}}_i^T.\mathbf{M}_i.\widehat{\mathbf{G}}$ where $sk_i = \mathbf{S}_i, sk_j = \mathbf{S}_j$, $\mathbf{M}_i$ is the message corresponding to the ciphertext $\mathbf{C}_i$, $\widehat{\mathbf{G}} = \begin{pmatrix} \mathbf{G} & \mathbf{O} \\ \mathbf{O} & \mathbf{G} \end{pmatrix}$ and $\widehat{\mathbf{E}}_i$ is the error matrix.

More recently, Ananth, Jain, Jin and Maralta Ananth et al. (2020) in 2020 built a multi-key FHE with one round decryption in the plain model i.e. without a trusted setup from DSPR, LWE, Ring-LWE assumptions against semi-honest adversary. This scheme has compact decryption and with no depth dependency assuming circular security. Briefly, an encryption scheme is circular secure if it can securely encrypt a function of its secret key under its public key. There have only a few schemes related to multi-bit FHE Brakerski et al. (2013), Hiromasa et al. (2016), Li et al. (2016) prior the work of Li, Ma and Zhou Li et al. (2018). However, none of these schemes are in the multi-key setting. In 2013, Brakerski, Gentry and Halevi Brakerski et al. (2013) introduced a simple method to encrypt a multi-bit plaintext into a packed ciphertext and compute efficient homomorphic operations on it. Hiromosa, Abe and Okamato Hiromasa et al. (2016) in 2016 proposed an FHE scheme which can encrypt a matrix and support homomorphic addition and multiplication. The security of both these schemes is based on LWE assumption. Li, Ma, Morais and Du Li et al. (2016) came up with a multi-bit homomorphic encryption based on the encryption scheme of Gentry, Peikert and Vaikuntanathan Gentry et al. (2008). The scheme is secure under the hardness of LWE assumption.

2.1 Structure of this paper

The remainder of this paper are organized as follows. In Section 3, we first give definitions required for our construction. We then describe our protocol in Section 4. In

Section 5, we provide complete security analysis for our proposed scheme. Finally, we conclude in Section 6.

3 Preliminaries

3.1 Notations

By $\mathbf{H}^T$ we denote the transpose of a matrix $\mathbf{H}$. We write $\mathbf{H}_i$ and $\mathbf{H}_i^T$ to denote the i-th row and i-th column of the matrix $\mathbf{H}$. We use the symbol $\mathbf{A}|\mathbf{B}$ for concatenation of $\mathbf{A}$ and $\mathbf{B}$. The set $\{1, 2, \ldots, n\}$ is denoted by $[n]$. If $\mathbf{v} = (v_1, v_2, \ldots, v_n)$ then we represent l_1 norm of $\mathbf{v}$ as $l_1(\mathbf{v}) = \sum_{i=1}^{n} |v_i|$ and l_∞ norm of $\mathbf{v}$ as $l_\infty(\mathbf{v}) = \max\limits_{1 \le i \le n} |v_i|$. We denote respectively the l_1 norm and l_∞ norm of matrix $\mathbf{H}$ by $l_1(\mathbf{H})$ and $l_\infty(\mathbf{H})$ where $l_1(\mathbf{H}) = \max\limits_{1 \le i \le k} l_1(\mathbf{H}_i^T)$ and $l_\infty(\mathbf{H}) = \max\limits_{1 \le i \le k} l_\infty(\mathbf{H}_i^T)$, k being the number of column of the matrix $\mathbf{H}$. A function $\mathsf{negl} : \mathbb{N} \to \mathbb{R}$ is said to be negligible if it approaches to zero faster than the reciprocal of any polynomial i.e. for every $c \in \mathbb{N}$ there is an integer n_c such that $\mathsf{negl}(n) < n^{-c}$ for all $n \ge n_c$. By $\mathbf{a} \leftarrow \mathcal{D}$, we mean that $\mathbf{a}$ is sampled randomly from the probability distribution (or the set) $\mathcal{D}$. For any two matrices $\mathbf{A}$ of order $m \times n$ and $\mathbf{B}$ of order $s \times t$, the *Kronecker* or *tensor product* is denoted by $\mathbf{A} \otimes \mathbf{B}$ and defined as $\mathbf{A} \otimes \mathbf{B} = (a_{i,j}\mathbf{B})_{i \in [m], j \in [n]}$. Moreover, $\mathbf{A} \otimes' \mathbf{B}$ indicates a variant of *tensor product* and is represented as $\mathbf{A} \otimes' \mathbf{B} = ((a_{i,j}\mathbf{B}_k^T)_{i \in [m], j \in [n]})_{k \in [t]}$.

3.2 Row and Column Stretch and Gadget matrix

DEFINITION 1. *(**Stretching by row or by column** Wang et al. (2017)). For a matrix $\mathbf{H}$ of dimension $m^2 \times n^2$, the stretching matrix of $\mathbf{H}$ by row, denoted by $\mathbf{H}_{\mathsf{sr}}$, is a matrix of order $m \times mn^2$ defined as*

$$\mathbf{H}_{\mathsf{sr}} = \begin{pmatrix} \mathbf{H}_1 & \mathbf{H}_2 & \cdots & \mathbf{H}_m \\ \mathbf{H}_{m+1} & \mathbf{H}_{m+2} & \cdots & \mathbf{H}_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ \mathbf{H}_{m(m-1)+1} & \mathbf{H}_{m(m-1)+2} & \cdots & \mathbf{H}_{m^2} \end{pmatrix}$$

where $\mathbf{H}_i$ is the i-th row of $\mathbf{H}$, $1 \le i \le m^2$. Similarly, the stretching matrix of $\mathbf{H}$ by column, denoted by $\mathbf{H}_{\mathsf{sc}}$, is the matrix of order $m^2 n \times n$ defined as

$$\mathbf{H}_{\mathsf{sc}} = \begin{pmatrix} \mathbf{H}_1^T & \mathbf{H}_{n+1}^T & \cdots & \mathbf{H}_{n(n-1)+1}^T \\ \mathbf{H}_2^T & \mathbf{H}_{n+2}^T & \cdots & \mathbf{H}_{n(n-1)+2}^T \\ \cdots & \cdots & \cdots & \cdots \\ \mathbf{H}_n^T & \mathbf{H}_{2n}^T & \cdots & \mathbf{H}_{n^2}^T \end{pmatrix}$$

where $\mathbf{H}_i^T$ is the i-th column of $\mathbf{H}$, $1 \le i \le n^2$.

THEOREM 3. *Wang et al. (2017) For any matrices $\mathbf{A}$ and $\mathbf{B} \in \mathbb{Z}_q^{n \times m}$ and $\mathbf{S} \in \mathbb{Z}_q^{m \times n}$, we have $(\mathbf{SA}) \cdot (\mathbf{SB}) = (\mathbf{S} \otimes \mathbf{S})_{\mathsf{sr}} \cdot (\mathbf{A} \otimes' \mathbf{B})_{\mathsf{sc}} \bmod q$.*

DEFINITION 2. *(**Gadget matrix and bit decomposition operator**). We use the standard* gadget vector $\mathbf{g} = \left(1, 2, \ldots, 2^{l-1}\right) \in \mathbb{Z}_q^l$ *where $l = \lceil \log\ q \rceil$. The* bit decomposition function $\mathbf{g}^{-1} : \mathbb{Z}_q \longrightarrow \{0,1\}^l$ *outputs a binary column vector consisting of the binary representation of its argument and satisfies the identity $\mathbf{g}\mathbf{g}^{-1}(a) = a$. More*

generally, the matrix $\mathbf{G} = (\mathbf{I}_n \otimes \mathbf{g})$ *is the* n*-row* gadget matrix. *We define the* bit-decomposition operator *on height-*n *vectors (or matrices)* $\mathbf{G}^{-1}(\cdot) = \left(\mathbf{I}_n \otimes \mathbf{g}^{-1}\right)(\cdot)$ *which applies* $\mathbf{g}^{-1}$ *entrywise to a height-*n *vector (or matrix) say* $\mathbf{A}$ *and thereby produces a height-*nl *binary vector/matrix* $\mathbf{G}^{-1}(\mathbf{A})$ *as output that satisfies the identity* $\mathbf{G}\mathbf{G}^{-1}(\mathbf{A}) = (\mathbf{I}_n \otimes \mathbf{g})\left(\mathbf{I}_n \otimes \mathbf{g}^{-1}\right)(\mathbf{A}) = \mathbf{A}$.

3.3 Preliminaries on Learning with Errors

DEFINITION 3. (**Discrete Gaussian Distribution**). *For* $\mathbf{L} \subseteq \mathbb{Z}^m$, $\mathbf{x} \in \mathbf{L}$ *and* $\mathbf{c} \in \mathbb{R}^m$ *and a positive parameter* $\sigma \in \mathbb{R}$, *define the Gaussian function* $\rho_{\sigma,\mu}(\mathbf{x})$ *and the probability density function* $\rho_{\sigma,\mu}(\mathbf{L})$ *as follows:*

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

For any $\mathbf{y} \in \mathbf{L}$, *there exists the discrete Gaussian distribution* $D_{\mathbf{L},\mu,\sigma}(\mathbf{y}) = \frac{\rho_{\sigma,\mu}(\mathbf{y})}{\rho_{\sigma,\mu}(\mathbf{L})}$ *over* $\mathbf{L}$ *with center* μ *and parameter* σ. *When* $\mu = 0$, *we write* $D^m_\sigma(\mathbf{y})$ *instead of* $D_{\mathbf{L},0,\sigma}(\mathbf{y})$.

LEMMA 1. *Li et al. (2018) The following results hold*

(i) $\Pr[|\mathbf{z}| > \omega(\sigma\sqrt{\log m}) : \mathbf{z} \leftarrow D^1_\sigma] = 2^{-\omega(\log m)}$,

(ii) $\Pr[||\mathbf{z}|| > 2\sigma\sqrt{m}, \mathbf{z} \leftarrow D^m_\sigma] < 2^{-m}$.

DEFINITION 4. (B-**Bounded Distribution**). *A distribution ensemble* $\{D_n\}_{n\in\mathbb{N}}$ *supported over the integers is said to be* B*-bounded if* $\Pr_{e\leftarrow D_n}[|e| > B] = \mathsf{negl}(n)$ *where* $\mathsf{negl}(n)$ *is a negligible function in* n.

DEFINITION 5. (**Decisional** $\mathsf{LWE}_{n,q,\chi}$ **Problem**). *Let* λ *be a security parameter and* $n = n(\lambda)$, $q = q(\lambda) \geq 2$ *are integers and* $\chi = \chi(\lambda)$ *be a probability distribution on* $\mathbb{Z}$. *The decisional* $\mathsf{LWE}_{n,q,\chi}$ *problem decide whether two distributions* Dist_0 *and* Dist_1 *are distinguishable or not.*

$$\mathsf{Dist}_0 = \{(\mathbf{A},\mathbf{b}) \in \mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^m : \mathbf{A} \xleftarrow{R} \mathbb{Z}_q^{n\times m}, \mathbf{b} \xleftarrow{R} \mathbb{Z}_q^m\}$$
$$\mathsf{Dist}_1 = \{(\mathbf{A},\mathbf{b}) \in \mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^m : \mathbf{s} \xleftarrow{R} \mathbb{Z}_q^n, \mathbf{e} \xleftarrow{R} \chi^m, \mathbf{A} \xleftarrow{R} \mathbb{Z}_q^{n\times m}, \mathbf{b} = \mathbf{s}\mathbf{A} + \mathbf{e} \in \mathbb{Z}_q^m\}.$$

DEFINITION 6. (**Decisional** $\mathsf{LWE}_{n,q,\chi}$ **assumption**). *The decisional* $\mathsf{LWE}_{n,q,\chi}$ *assumption holds if*

$$|\Pr[\mathcal{A}(\mathbf{A},\mathbf{b}) = 1 : (\mathbf{A},\mathbf{b}) \leftarrow \mathsf{Dist}_0] - \Pr[\mathcal{A}(\mathbf{A},\mathbf{b}) = 1 : (\mathbf{A},\mathbf{b}) \leftarrow \mathsf{Dist}_1]| = \mathsf{negl}(\lambda)$$

for every PPT adversary $\mathcal{A}$ *where* $\mathsf{negl}(\lambda)$ *is a negligible function in* λ.

DEFINITION 7. (*GapSVP_γ*). *An instance of* GapSVP_γ *is given by an* n *dimensional lattice* $\mathbf{L}$, *a real number* $d > 0$ *and* $\lambda_1(\mathbf{L})$ *where* $\lambda_1(\mathbf{L})$ *denotes the length of the shortest non zero vector in* $\mathbf{L}$. *In YES instances,* $\lambda_1(\mathbf{L}) \leq d$ *and in NO instances* $\lambda_1(\mathbf{L}) > \gamma d$ *where real number* γ *is an approximation factor.*

THEOREM 4. *Regev (2009) Let $B \geq \sqrt{n}(\log\ n)$ and $q = q(n) \in \mathbb{N}$ be either a prime power or a product of distinct primes each of size $\mathsf{poly}(n)$ where $\mathsf{poly}(n)$ is polynomial function in n. Then there exists an efficient samplable B-bounded distribution χ such that the existence of an efficient algorithm that solves the decisional $\mathsf{LWE}_{n,q,\chi}$ problem implies the followings:*

- *there is an efficient quantum algorithm that solves $\mathsf{GapSVP}_{\tilde{O}(nq/B)}$ on any n-dimensional lattice,*
- *there is an efficient classical algorithm for $\mathsf{GapSVP}_{\tilde{O}(nq/B)}$ on any n-dimensional lattice if $q > \tilde{O}(2^{n/2})$. Here $\tilde{O}$ notation hides logarithmic factors in n, q and B.*

3.4 Defining Leveled Multi-key Fully homomorphic Encryption

DEFINITION 8. *(**(leveled) Multi-Key FHE** López-Alt et al. (2012)). A* (leveled) multi-key FHE *scheme* $\mathsf{MMFHE} = (\mathsf{Setup}, \mathsf{Keygen}, \mathsf{Enc}, \mathsf{Ext}, \mathsf{Eval}, \mathsf{Dec})$ *with message space $\mathcal{M}$ consists of the following probabilistic polynomial time (PPT) algorithms:*

• $\mathsf{MMFHE.Setup}(1^\lambda, 1^d) \rightarrow params$*: On input the security parameter λ and maximum circuit depth d, a trusted authority outputs the system parameters $params$.*

• $\mathsf{MMFHE.Keygen}(params) \rightarrow (pk, sk)$*: This algorithm is run by a user who generates a public-secret key pair (pk, sk). The user publishes pk as his public key and keeps sk secret to himself.*

• $\mathsf{MMFHE.Enc}(params, pk, \mathbf{M}) \rightarrow (\mathbf{C}, \mathsf{Aux})$*: A user encrypts the message $\mathbf{M} \in \mathcal{M}$ under his own public key pk and outputs a fresh ciphertext $\mathbf{C}$ and an auxiliary information* Aux.

• $\mathsf{MMFHE.Ext}(params, \mathsf{PK}, pk_{j_i}, \mathbf{C}_i, \mathsf{Aux}_i) \rightarrow \widehat{\mathbf{C}}_i$*: Let $\mathsf{PK} = (pk_{j_1}, pk_{j_2}, \ldots, pk_{j_k})$ and $1 \leq i \leq k$. On input* PK *consisting of k public keys $pk_{j_1}, pk_{j_2}, \ldots, pk_{j_k}$ and a fresh ciphertext $\mathbf{C}_i$ generated using the public key pk_{j_i} together with the auxiliary information Aux_i associated with $\mathbf{C}_i$, a server outputs an extended ciphertext $\widehat{\mathbf{C}}_i$ without explicitly knowing the secret keys $sk_{j_1}, sk_{j_2}, \ldots, sk_{j_k}$. If $(\mathbf{C}_i, \mathsf{Aux}_i) = \mathsf{MMFHE.Enc}(params, pk_{j_i}, \mathbf{M}_i)$ then by $\widehat{\mathbf{C}}_i = \mathsf{MMFHE.Enc}(params, \mathsf{PK}, \mathsf{Con}_k(\mathbf{M}_i))$ it means $\widehat{\mathbf{C}}_i$ is an encryption of $\mathsf{Con}_k(\mathbf{M}_i)$ under the public key* PK *where $\mathsf{Con}_k(\mathbf{M}_i)$ is a $k \times k$ block matrix with $\mathbf{M}_i \in \mathcal{M}$ and $\mathbf{O}$ having with same dimension. Note that when $k = 1$, we have $\mathsf{PK} = pk_{j_i}$ and $\widehat{\mathbf{C}}_i = \mathbf{C}_i = \mathsf{MMFHE.Enc}(params, pk_{j_i}, \mathbf{M}_i)$ is a fresh ciphertext which is an encryption of message $\mathsf{Con}_1(\mathbf{M}_i) = \mathbf{M}_i$ under the public key pk_{j_i}. Observe that additionally, Aux_i is output during the generation of fresh ciphertext $\mathbf{C}_i$ for the message $\mathbf{M}_i$ under pk_{j_i}.*

• $\mathsf{MMFHE.Eval}(params, \mathsf{PK}, \mathcal{C}, \mathbf{C}_1^*, \mathbf{C}_2^*, \ldots, \mathbf{C}_s^*) \rightarrow \mathbf{C}'$*: Let $\mathsf{PK} = (pk_{j_1}, pk_{j_2}, \ldots, pk_{j_k})$ and $\mathbf{C}_i^* = \mathsf{MMFHE.Enc}(params, \mathsf{PK}, \mathbf{M}_i^*), 1 \leq i \leq s$ where $\mathbf{M}_i^* = \mathsf{Con}_k(\mathbf{M}_i)$. Given a circuit $\mathcal{C}$ of depth $\leq d$ along with s ciphertexts $\mathbf{C}_1^*, \mathbf{C}_2^*, \ldots, \mathbf{C}_s^*$, a server uses* PK *to output an evaluated ciphertext $\mathbf{C}'$ which is an encryption of $\mathcal{C}(\mathbf{M}_1^*, \mathbf{M}_2^*, \ldots, \mathbf{M}_s^*) \in \mathcal{M}$ under the public key* PK*. Note that, all the input ciphertexts $\mathbf{C}_i^*, 1 \leq i \leq s$, must be generated under the common public key* PK*.*

• $\mathsf{MMFHE.Dec}(params, \mathsf{SK}, \mathbf{C}') \rightarrow \mathbf{M}'$*: Ciphertexts are decrypted in the following ways:*

Case-I: *Let* $\mathbf{C}^{'}$ *be a fresh ciphertext i.e.* $\mathbf{C}^{'} = (\mathbf{C}, \mathsf{Aux}) \leftarrow \mathsf{MMFHE.Enc}(params, pk, \mathbf{M})$. *The decrypter uses the secret key* sk *corresponding to* pk *and recovers the message* $\mathbf{M} \in \mathcal{M}$.
Case-II: *Let* $\mathbf{C}^{'}$ *be an extended ciphertext i.e.* $\mathbf{C}^{'} = \mathbf{C}_i^* = \mathsf{MMFHE.Ext}(params, \mathsf{PK}, pk_{j_i}, \mathbf{C}_i, \mathsf{Aux}_i)$ *under the public key* $\mathsf{PK} = (pk_{j_1}, pk_{j_2}, \ldots, pk_{j_k})$ *corresponding to the secret key* $\mathsf{SK} = (sk_{j_1}, sk_{j_2}, \ldots, sk_{j_k})$. *Then the* k *parties* $j_1, j_2, \ldots, j_k$ *run a multi-party decryption protocol and recover the message.*
Case-III: *Let* $\mathbf{C}^{'}$ *be an evaluated ciphertext i.e.* $\mathbf{C}^{'} = \mathsf{MMFHE.Eval}(params, \mathsf{PK}, \mathcal{C}, \mathbf{C}_1^*, \mathbf{C}_2^*, \ldots, \mathbf{C}_s^*)$ *where for* $1 \leq i \leq s$, $\mathbf{C}_i^*$ *are ciphertext of the message* $\mathbf{M}^* = \mathsf{Con}_k(\mathbf{M}_i)$ *under the public key* $\mathsf{PK} = (pk_{j_1}, pk_{j_2}, \ldots, pk_{j_k})$ *corresponding to the secret key* $\mathsf{SK} = (sk_{j_1}, sk_{j_2}, \ldots, sk_{j_k})$. *Then the parties* $j_1, j_2, \ldots, j_k$ *run a multi-party decryption protocol and get the evaluated message.*

MMFHE must satisfies the correctness and compactness:

DEFINITION 9. **(Correctness)**. *A* MMFHE *scheme with matrix message space* $\mathcal{M}$ *is correct if:*

- $\mathsf{MMFHE.Dec}(params, \mathsf{SK}, \mathbf{C}_i) = \mathbf{M}_i$ *for each fresh ciphertext* $\mathbf{C}_i$ *satisfying* $\mathbf{C}_i = \mathsf{MMFHE.Enc}(params, pk_{j_i}, \mathbf{M}_i)$ *where* $\mathsf{SK} = sk_{j_i}$.
- $\mathsf{MMFHE.Dec}(params, \mathsf{SK}, \widehat{\mathbf{C}}_i) = \mathsf{Con}_k(\mathbf{M}_i)$ *for each extended ciphertext* $\widehat{\mathbf{C}}_i$ *satisfying* $\widehat{\mathbf{C}}_i = \mathsf{MMFHE.Enc}(params, \mathsf{PK}, \mathsf{Con}_k(\mathbf{M}_i))$ *where* $\mathsf{SK} = (sk_{j_1}, sk_{j_2}, \ldots, sk_{j_k})$ *and* $\mathsf{PK} = (pk_{j_1}, pk_{j_2}, \ldots, pk_{j_k})$.
- $\mathsf{MMFHE.Dec}(params, \mathsf{SK}, \mathbf{C}^{'}) = \mathcal{C}(\mathbf{M}_1^*, \mathbf{M}_2^*, \ldots, \mathbf{M}_s^*)$ *for each evaluated ciphertext* $\mathbf{C}^{'}$ *satisfying* $\mathbf{C}^{'} = \mathsf{MMFHE.Enc}(params, \mathsf{PK}, \mathcal{C}(\mathbf{M}_1^*, \mathbf{M}_2^*, \ldots, \mathbf{M}_s^*))$ *where* $\mathsf{SK} = (sk_{j_1}, sk_{j_2}, \ldots, sk_{j_k})$, $\mathsf{PK} = (pk_{j_1}, pk_{j_2}, \ldots, pk_{j_k})$ *and* $\mathbf{M}_i^* = \mathsf{Con}_k(\mathbf{M}_i)$, $\mathbf{M}_i \in \mathcal{M}$.

DEFINITION 10. *(***Compactness***)*. *A* MMFHE *scheme is compact if* $|\mathbf{C}^{'}| \leq p(\lambda, N, d)$ *where* λ *is the security parameter,* d *is the depth of the circuit and* $N = N(\lambda, d)$ *is a designated upper bound on the number of parties. More formally, if* $\mathbf{C}^{'} = \mathsf{MMFHE.Eval}(params, \mathsf{PK}, \mathcal{C}, \mathbf{C}_1^*, \mathbf{C}_2^*, \ldots, \mathbf{C}_s^*)$ *with* $\mathbf{C}_i^* = \mathsf{MMFHE.Enc}(params, \mathsf{PK}, \mathbf{M}_i^*), 1 \leq i \leq s$, *then* $\mathbf{C}^{'} = \mathsf{MMFHE.Enc}(params, \mathsf{PK}, \mathcal{C}(\mathbf{M}_1^*, \mathbf{M}_2^*, \ldots, \mathbf{M}_s^*))$ *and* $\mathbf{C}^{'}$ *is independent of the size and the number of input wires of the circuit* $\mathcal{C}$ *although it can depend polynomially on* $\lambda, N,$ *and* d.

Note that the algorithms MMFHE.Eval and MMFHE.Ext are run by a public server, the indistinguishability under chosen-plaintext attack (IND-CPA) security of a multi-key FHE scheme $\mathsf{MMFHE} = (\mathsf{Setup}, \mathsf{Keygen}, \mathsf{Enc}, \mathsf{Ext}, \mathsf{Eval}, \mathsf{Dec})$ relies on the IND-CPA security of the underlying basic encryption scheme $\mathsf{MMFHE.basic} = (\mathsf{Setup}, \mathsf{Keygen}, \mathsf{Enc}, \mathsf{Dec})$ which has the same Setup, Keygen, Enc and Dec algorithms as that of MMFHE. Therefore, we define below the IND-CPA security of MMFHE.basic.

DEFINITION 11. **(Indistinguishability under Chosen-Plaintext Attack (IND-CPA))**. *An encryption scheme* $\mathsf{MMFHE.basic} = (\mathsf{Setup}, \mathsf{Keygen}, \mathsf{Enc}, \mathsf{Dec})$ *is IND-CPA secure if*

$$\mathsf{Adv}^{\mathsf{IND-CPA}}_{\mathcal{A},\mathsf{MMFHE.basic}} = |\Pr[\mathsf{Expt}^{\mathsf{MMFHE.basic}}_{\mathcal{A}}(1^\lambda, b) = 1] - \frac{1}{2}| \leq p(\lambda)$$

for any $\lambda \in \mathbb{N}$ and every PPT distinguisher $\mathcal{A}$ in the experiments $\mathsf{Expt}_{\mathcal{A}}^{\mathsf{MMFHE.basic}}(1^\lambda, b)$ described in FIGURE 1 where $b \in \{0,1\}$ and p is negligible function of λ. Here, $\mathsf{Adv}^{\mathsf{IND-CPA}}_{\mathcal{A},\mathsf{MMFHE.basic}}$ is the advantage of $\mathcal{A}$ in breaking the scheme MMFHE.basic.

1. The challenger runs

$$\mathsf{MMFHEbasic.Setup}(1^\lambda, 1^d) \rightarrow params$$
$$\mathsf{MMFHEbasic.Keygen}(params) \rightarrow (pk, sk)$$

   and makes pk public and keeps sk secret.

2. The adversary $\mathcal{A}$ selects $\mathbf{M}_0, \mathbf{M}_1 \in \mathcal{M}$ of equal length and sends $(\mathbf{M}_0, \mathbf{M}_1)$ to the challenger.

3. The challenger chooses a random bit $b \in \{0,1\}$ and computes a challenge ciphertext $\mathbf{C}_b$ in the following way and sends it to $\mathcal{A}$:

   – The ciphertext $\mathbf{C}_b$ for the message $\mathbf{M}_b$ under the public key pk generated as

$$\mathbf{C}_b \leftarrow \mathsf{MMFHEbasic.Enc}(params, pk, \mathbf{M}_b)$$

4. Finally, the adversary $\mathcal{A}$ outputs a guess bit $b^{'}$ for b.

Figure 1: $\mathsf{Expt}_{\mathcal{A}}^{\mathsf{MMFHEbasic}}(1^\lambda, b)$: The IND-CPA security game of MMFHEbasic.

DEFINITION 12. *(**Statistical distance**). The statistical distance between two distributions $\mathcal{X}$ and $\mathcal{Y}$ over a set A is defined as*

$$\delta(\mathcal{X}, \mathcal{Y}) = \frac{1}{2} \sum_{x \in A} |\Pr\{x \leftarrow \mathcal{X}\} - \Pr\{x \leftarrow \mathcal{X}\}|$$

Two distributions $\mathcal{X}$ and $\mathcal{Y}$ are said to be statistically close if $\delta(\mathcal{X}, \mathcal{Y}) = \mathsf{negl}(\lambda)$ where $\mathsf{negl}(\lambda)$ is a negligible function in the security parameter λ.

DEFINITION 13. *(**Leftover hash lemma** Micciancio (2018)). For any odd integer q, integers k and $n > k \log q$, if $\mathbf{A} \leftarrow \mathbb{Z}_q^{k \times n}$, $\mathbf{z} \leftarrow \{0,1\}^n$ and $\mathbf{y} \leftarrow \mathbf{Z}_q^k$ are chosen uniformly at random, then $(\mathbf{A}, \mathbf{Az})$ is statistically close to $(\mathbf{A}, \mathbf{y})$.*

4 Protocol

Our LWE based (leveled) multi-key FHE construction $\mathsf{MMFHE} = (\mathsf{Setup}, \mathsf{Keygen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval}, \mathsf{Ext})$ builds on the matrix variant of Regev's basic public key encryption Regev (2009). In this description, we consider the message space $\mathcal{M} = \bigcup_{i=1}^{N} \{0,1\}^{ir \times ir}$ where r is an integer and $N = N(\lambda, d)$ is a designated upper bound on the number of

parties in the system.

• $\mathsf{MMFHE.Setup}(1^\lambda, 1^d) \rightarrow params$: On input the security parameter λ and maximum circuit depth d, a trusted third party chooses a lattice dimension $n = n(\lambda, d)$, sets integers r, $q = q(\lambda, d) \geq 2$, selects a β-bounded error distribution (discrete Gaussian distribution) $\chi = \chi(\lambda, d)$ over $\mathbb{Z}_q$, picks a common random matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times r}$ and outputs the public parameter $params = (n, q, \chi, \beta, r, N, \mathbf{A})$.

• $\mathsf{MMFHE.Keygen}(params) \rightarrow (pk, sk)$: A user uses $params$, samples $\mathbf{S}' \in \mathbb{Z}_q^{r \times n}$, chooses error matrix $\mathbf{E} \in \chi^{r \times r}$ (i.e. $||\mathbf{E}||_\infty \leq \beta$) and sets

$$pk = \mathbf{B} = \begin{pmatrix} \mathbf{S}'\mathbf{A} + \mathbf{E} \\ \mathbf{A} \end{pmatrix} \in \mathbb{Z}_q^{(n+r) \times r}$$

$$sk = \mathbf{S} = (\mathbf{I}_r | -\mathbf{S}') \in \mathbb{Z}_q^{r \times (n+r)}$$

where $\mathbf{I}_r$ is the $r \times r$ identity matrix.

• $\mathsf{MMFHE.Enc}(params, pk = \mathbf{B}, \mathbf{M}) \rightarrow (\mathbf{C}, \mathsf{Aux})$: To encrypt a message $\mathbf{M} \in \{0,1\}^{r \times r}$, a user with public key $pk = \mathbf{B}$ chooses a symmetric matrix $\mathbf{R} \in \{0,1\}^{r \times r}$, $\mathbf{E}' \in \chi^{r \times rl}$ randomly and computes

$$\mathbf{C} = \left\lfloor \frac{q}{2} \right\rfloor \begin{pmatrix} \mathbf{M} \\ \mathbf{O} \end{pmatrix} + \mathbf{BR} \in \mathbb{Z}_q^{(n+r) \times r}$$

$$\mathsf{Aux} = \mathbf{RG} + \mathbf{E}' \mod q \in \mathbb{Z}_q^{r \times rl}$$

where $\mathbf{G} = \mathbf{I}_r \otimes \mathbf{g} \in \mathbb{Z}_q^{r \times rl}$ is the gadget matrix with $\mathbf{g} = (1, 2, \ldots, 2^{l-1})$, $l = \lceil \log q \rceil$ and $\mathbf{O}$ is the zero matrix of order $n \times r$. The user publishes $\mathbf{C}$ as the fresh ciphertext together with the auxiliary information Aux.

Observe that the public-secret key pair $(pk = \mathbf{B}, sk = \mathbf{S})$ satisfies the relation $sk.pk = \mathbf{E}$ as

$$sk.pk = \mathbf{SB} = (\mathbf{I}_r | -\mathbf{S}') \begin{pmatrix} \mathbf{S}'\mathbf{A} + \mathbf{E} \\ \mathbf{A} \end{pmatrix} = \mathbf{E}.$$

Hence,

$$\begin{aligned} sk.\mathbf{C} = \mathbf{SC} &= \mathbf{S}\left(\lfloor \frac{q}{2} \rfloor \begin{pmatrix} \mathbf{M} \\ \mathbf{O} \end{pmatrix} \right) + \mathbf{SBR} \\ &= \lfloor \frac{q}{2} \rfloor \mathbf{M} + \mathbf{ER} = \lfloor \frac{q}{2} \rfloor \mathbf{M} + \mathbf{E}_\mathbf{C}. \end{aligned}$$

Thus the ciphertext $\mathbf{C}$ for the plaintext matrix $\mathbf{M} \in \{0,1\}^{r \times r}$ satisfies $sk.\mathbf{C} = \mathbf{SC} = \lfloor \frac{q}{2} \rfloor \mathbf{M} + \mathbf{E}_\mathbf{C} \in \mathbb{Z}_q^{r \times r}$ with error $||\mathbf{E}_\mathbf{C}||_\infty = ||\mathbf{ER}||_\infty \leq r\beta$. We say that $\mathbf{C}$ is $r\beta$-noisy ciphertext.

• $\mathsf{MMFHE.Ext}(params, \mathsf{PK}, pk_{j_i}, \mathbf{C}_i, \mathsf{Aux}_i) \rightarrow \widehat{\mathbf{C}}_i$: Let $\mathsf{PK} = (pk_{j_1}, pk_{j_2}, \ldots, pk_{j_k})$ and $1 \leq i \leq k$. A server runs the following steps taking PK as input along with the pair $(\mathbf{C}_i, \mathsf{Aux}_i)$ where $\mathbf{C}_i$ denotes a fresh ciphertext for the plaintext matrix $\mathbf{M}_i \in \{0,1\}^{r \times r}$ under the public key pk_{j_i} for some $i, 1 \leq i \leq k$ and Aux_i is the auxiliary information associated with $\mathbf{C}_i$.

– Runs the procedure pairwiseExt described below in FIGURE 2 and computes $\mathbf{X}_a \leftarrow \mathsf{pairwiseExt}(params, (pk_{j_i}, pk_{j_a}), pk_{j_i}, \mathbf{C}_i, \mathsf{Aux}_i)$ for $1 \leq a \leq k$.

Let

$$pk_{j_i} = \mathbf{B}_i = \begin{pmatrix} \mathbf{S}_i'\mathbf{A} + \mathbf{E}_i \\ \mathbf{A} \end{pmatrix} \in \mathbb{Z}_q^{(n+r)\times r}$$
$$pk_{j_a} = \mathbf{B}_a = \begin{pmatrix} \mathbf{S}_a'\mathbf{A} + \mathbf{E}_a \\ \mathbf{A} \end{pmatrix} \in \mathbb{Z}_q^{(n+r)\times r}$$

where

$$sk_{j_i} = \mathbf{S}_i = (\mathbf{I}_r | -\mathbf{S}_i') \in \mathbb{Z}_q^{r\times(n+r)}$$
$$sk_{j_a} = \mathbf{S}_a = (\mathbf{I}_r | -\mathbf{S}_a') \in \mathbb{Z}_q^{r\times(n+r)}$$

with $\mathbf{S}_i\mathbf{B}_i = \mathbf{E}_i$ and $\mathbf{S}_a\mathbf{B}_a = \mathbf{E}_a$.

• Let ciphertext corresponding to message $\mathbf{M}_i$ be $(\mathbf{C}_i, \mathsf{Aux}_i) \leftarrow \mathsf{MMFHE.Enc}(params, pk_{j_i}, \mathbf{M}_i)$ with

$$\mathbf{C}_i = \lfloor \frac{q}{2} \rfloor \begin{pmatrix} \mathbf{M}_i \\ \mathbf{O} \end{pmatrix} + \mathbf{B}_i\mathbf{R}_i \in \mathbb{Z}_q^{(n+r)\times r}$$
$$\mathsf{Aux}_i = \mathbf{R}_i\mathbf{G} + \mathbf{E}_i^{'} \mod q \in \mathbb{Z}_q^{r\times rl}$$

which implicitly satisfies the equation

$$sk_{j_i}.\mathbf{C}_i = \lfloor \frac{q}{2} \rfloor \mathbf{M}_i + \mathbf{E}_{\mathbf{C}_i},$$

$\mathbf{R}_i \in \{0,1\}^{r\times r}$ being a randomly selected symmetric matrix and error $||\mathbf{E}_{\mathbf{C}_i}||_\infty = ||\mathbf{E}_i\mathbf{R}_i||_\infty \leq r\beta$.

• The server works as follows:

– Computes

$$pk_{j_a} - pk_{j_i} = \mathbf{B}_a - \mathbf{B}_i = \begin{pmatrix} \mathbf{S}_a'\mathbf{A} + \mathbf{E}_a - \mathbf{S}_i'\mathbf{A} - \mathbf{E}_i \\ \mathbf{O} \end{pmatrix} \in \mathbb{Z}_q^{(n+r)\times r}$$

where $\mathbf{O}$ is the zero matrix of order $n \times r$.

– Outputs $\mathbf{X}_a$ as

$$\mathbf{X}_a = [\mathsf{Aux}_i\mathbf{G}^{-1}((\mathbf{B}_a - \mathbf{B}_i)^T)]^T$$

where $\mathbf{G}^{-1}(.) = \mathbf{I}_r \otimes g^{-1}(.)$ is the bit decomposition operator applies on r-height matrices or vectors and satisfies $\mathbf{G}.\mathbf{G}^{-1}(\mathbf{H}) = \mathbf{H}$ for any r-height matrix $\mathbf{H}$. As $\mathbf{R}_i^T = \mathbf{R}_i$ whereas $\mathbf{X}_a = (\mathbf{B}_a - \mathbf{B}_i)\mathbf{R}_i + \mathbf{E}_a^{''}$ with $\mathbf{E}_a^{''} = [\mathbf{G}^{-1}((\mathbf{B}_a - \mathbf{B}_i)^T)]^T(\mathbf{E}_i^{'})^T)$

$$\begin{aligned} \mathbf{X}_a &= [\mathsf{Aux}_i\mathbf{G}^{-1}((\mathbf{B}_a - \mathbf{B}_i)^T)]^T \\ &= [(\mathbf{R}_i\mathbf{G} + \mathbf{E}_i^{'})\mathbf{G}^{-1}((\mathbf{B}_a - \mathbf{B}_i)^T)]^T \\ &= [\mathbf{R}_i(\mathbf{B}_a - \mathbf{B}_i)^T]^T + [\mathbf{E}_i^{'}\mathbf{G}^{-1}((\mathbf{B}_a - \mathbf{B}_i)^T)]^T \\ &= (\mathbf{B}_a - \mathbf{B}_i)\mathbf{R}_i^T + [\mathbf{G}^{-1}((\mathbf{B}_a - \mathbf{B}_i)^T)]^T(\mathbf{E}_i^{'})^T \\ &= (\mathbf{B}_a - \mathbf{B}_i)\mathbf{R}_i + \mathbf{E}_a^{''} \end{aligned}$$

Figure 2: The procedure $\mathsf{pairwiseExt}(params, (pk_{j_i}, pk_{j_a}), pk_{j_i}, \mathbf{C}_i, \mathsf{Aux}_i) \rightarrow \mathbf{X}_a$ procedure.

– Sets the extended ciphertext $\widehat{\mathbf{C}}_i = [\mathbf{C}_{a,b}]_{a,b\in[k]} \in \mathbb{Z}_q^{k(n+r)\times kr}$ where

$$\mathbf{C}_{a,b} = \begin{cases} \mathbf{C}_i & \text{if } a = b \\ \mathbf{X}^j & \text{if } b = j \neq i \text{ and } a = i \\ \mathbf{O} & \text{otherwise} \end{cases}$$

More preciously, $\widehat{\mathbf{C}}_i$ is equal to the following matrix.

$$\begin{array}{r} \\ \text{row } 1 \\ \vdots \\ \text{row } (i-1) \\ \text{row } i \\ \text{row } (i+1) \\ \vdots \\ \text{row } k \end{array} \begin{array}{c} \begin{array}{ccccccc} \text{col } 1 & \cdots & \text{col } (i-1) & \text{col } i & \text{col } (i+1) & \cdots & \text{col } k \end{array} \\ \begin{pmatrix} \mathbf{C}_i & \dots & \mathbf{O} & \mathbf{O} & \mathbf{O} & \dots & \mathbf{O} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \mathbf{O} & \dots & \mathbf{C}_i & \mathbf{O} & \mathbf{O} & \dots & \mathbf{O} \\ \mathbf{X}_1 & \dots & \mathbf{X}_{i-1} & \mathbf{C}_i & \mathbf{X}_{i+1} & \dots & \mathbf{X}_k \\ \mathbf{O} & \dots & \mathbf{O} & \mathbf{O} & \mathbf{C}_i & \dots & \mathbf{O} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \mathbf{O} & \dots & \mathbf{O} & \mathbf{O} & \mathbf{O} & \dots & \mathbf{C}_i \end{pmatrix} \end{array}$$

Note that $\widehat{\mathbf{C}}_i$ is generated without explicitly knowing the secret keys $sk_{j_1}, sk_{j_2}, \ldots, sk_{j_k}$. We now claim that $\widehat{\mathbf{C}}_i = \mathsf{MMFHE.Enc}(params, \mathsf{PK}, \mathsf{Con}_k(\mathbf{M}_i))$ is an extended ciphertext for the message

$$\begin{aligned} \mathsf{Con}_k(\mathbf{M}_i) &= (\mathbf{M}_i', \mathbf{O}, \ldots, \mathbf{O})^T \\ &= \begin{pmatrix} \mathbf{M}_i & \mathbf{M}_i & \cdots & \mathbf{M}_i \\ \mathbf{O} & \mathbf{O} & \cdots & \mathbf{O} \\ \cdots & \cdots & \cdots & \cdots \\ \mathbf{O} & \mathbf{O} & \cdots & \mathbf{O} \end{pmatrix} \in \{0,1\}^{kr\times kr} \end{aligned}$$

under the public key $\mathsf{PK} = (pk_{j_1}, pk_{j_2}, \ldots, pk_{j_k})$ where $\mathbf{M}_i' = (\mathbf{M}_i, \mathbf{M}_i, \cdots, \mathbf{M}_i)$ which corresponds to the secret key $\widehat{\mathsf{SK}} = (\mathsf{SK}, \mathbf{O}, \ldots, \mathbf{O})^T \in \mathbb{Z}_q^{kr\times k(n+r)}$ with $\mathsf{SK} = (sk_{j_1}, sk_{j_2}, \cdots, sk_{j_k})$. We show that $\widehat{\mathbf{C}}_i$ preserves the relation with $\widehat{\mathsf{SK}}$ similar to that preserved by the fresh ciphertext $\mathbf{C}_i$ with sk_{j_i}. THEOREM 5 below establishes the fact that if $\mathbf{C}_i$ is an $r\beta$-noisy encryption of $\mathbf{M}_i$ under pk_{j_i} then $\widehat{\mathbf{C}}_i$ provides an $r(1+l)\beta$-noisy encryption of $\mathsf{Con}_k(\mathbf{M}_i)$ under $\mathsf{PK} = (pk_{j_1}, pk_{j_2}, \ldots, pk_{j_k})$.

THEOREM 5. *The extended ciphertext $\widehat{\mathbf{C}}_i$ satisfies $\widehat{\mathsf{SK}}.\widehat{\mathbf{C}}_i = \lfloor \frac{q}{2} \rfloor \mathsf{Con}_k(\mathbf{M}_i) + \mathbf{E}_{\widehat{\mathbf{C}}_i} \in \{0,1\}^{kr\times kr}$ with error $|\mathbf{E}_{\widehat{\mathbf{C}}_i}|_\infty \leq r(1+l)\beta$ where $l = \lceil \log q \rceil$ and β is the bound of the error distribution.*

Proof: For $1 \leq a \leq k$, we note that

$$\begin{aligned} \mathbf{S}_a\mathbf{C}_i &= (\mathbf{I}_r | -\mathbf{S}_a') \left[\lfloor \frac{q}{2} \rfloor \begin{pmatrix} \mathbf{M}_i \\ \mathbf{O} \end{pmatrix} + \mathbf{B}_i\mathbf{R}_i \right] \\ &= \lfloor \frac{q}{2} \rfloor \mathbf{M}_i + (\mathbf{I}_r | -\mathbf{S}_a') \begin{pmatrix} \mathbf{S}_i'\mathbf{A} + \mathbf{E}_i \\ \mathbf{A} \end{pmatrix} . \mathbf{R}_i \\ &= \lfloor \frac{q}{2} \rfloor \mathbf{M}_i + (\mathbf{S}_i'\mathbf{A} + \mathbf{E}_i - \mathbf{S}_a'\mathbf{A})\mathbf{R}_i \\ &= \lfloor \frac{q}{2} \rfloor \mathbf{M}_i + (\mathbf{S}_i'\mathbf{A} + \mathbf{E}_i - \mathbf{S}_a'\mathbf{A} - \mathbf{E}_a + \mathbf{E}_a)\mathbf{R}_i \\ &= \lfloor \frac{q}{2} \rfloor \mathbf{M}_i + (\mathbf{S}_i'\mathbf{A} + \mathbf{E}_i - \mathbf{S}_a'\mathbf{A} - \mathbf{E}_a)\mathbf{R}_i + \mathbf{E}_a\mathbf{R}_i \end{aligned}$$

$$\begin{aligned}\mathbf{S}_i\mathbf{X}_a &= \mathbf{S}_i[(\mathbf{B}_a - \mathbf{B}_i)\mathbf{R}_i + \mathbf{E}_a^{''}] \\ &= (\mathbf{I}_r|-\mathbf{S}_i')\begin{pmatrix} \mathbf{S}_a'\mathbf{A} + \mathbf{E}_a - \mathbf{S}_i'\mathbf{A} - \mathbf{E}_i \\ \mathbf{O} \end{pmatrix}\mathbf{R}_i + \mathbf{S}_i\mathbf{E}_a^{''} \\ &= (\mathbf{S}_a'\mathbf{A} + \mathbf{E}_a - \mathbf{S}_i'\mathbf{A} - \mathbf{E}_i)\mathbf{R}_i + \mathbf{S}_i\mathbf{E}_a^{''}\end{aligned}$$

Hence,

$$\mathbf{S}_a\mathbf{C}_i = \left\lfloor \frac{q}{2} \right\rfloor \mathbf{M}_i - \mathbf{S}_i\mathbf{X}_a + \mathbf{E}_a\mathbf{R}_i \tag{1}$$

$$\mathbf{S}_a\mathbf{C}_i + \mathbf{S}_i\mathbf{X}_a = \left\lfloor \frac{q}{2} \right\rfloor \mathbf{M}_i + \mathbf{E}_a\mathbf{R}_i + \mathbf{S}_i\mathbf{E}_a^{''} \tag{2}$$

Let

$$\Gamma_\alpha = \begin{cases} [\mathbf{S}_1, \ldots, \mathbf{S}_{i-1}, \mathbf{S}_i, \mathbf{S}_{i+1}, \ldots, \mathbf{S}_k] & \text{if } \alpha = 1 \\ [\mathbf{O}, \ldots, \mathbf{O}, \mathbf{O}, \mathbf{O}, \ldots, \mathbf{O}] & \text{if } \alpha \neq 1 \end{cases}$$

and

$$\Lambda_\alpha = \begin{cases} [\mathbf{O}, \ldots, \mathbf{C}_i, \ldots, \mathbf{X}_i, \ldots, \mathbf{O}]^T & \text{if } \alpha \neq i \\ [\mathbf{O}, \ldots, \mathbf{O}, \ldots, \mathbf{C}_i, \ldots, \mathbf{O}]^T & \text{if } \alpha = i \end{cases}$$

Then $\widehat{\mathsf{SK}} = (\Gamma_1, \ldots, \Gamma_{i-1}, \Gamma_i, \Gamma_{i+1}, \ldots, \Gamma_k)$ and $\widehat{\mathbf{C}} = (\Lambda_1, \ldots, \Lambda_{i-1}, \Lambda_i, \Lambda_{i+1}, \ldots, \Lambda_k)$ and consequently, $\widehat{\mathsf{SK}}\widehat{\mathbf{C}} = (\Delta_1, \ldots, \Delta_{i-1}, \Delta_i, \Delta_{i+1}, \ldots, \Delta_k)$ where

$$\Delta_\alpha = \begin{cases} [\Psi_1, \ldots, \Psi_i, \ldots, \Psi_k] & \text{if } \alpha = 1 \\ [\mathbf{O}, \ldots, \mathbf{O}, \ldots, \mathbf{O}] & \text{if } \alpha \neq 1 \end{cases}$$

and

$$\Psi_\alpha = \begin{cases} \lfloor \frac{q}{2} \rfloor \mathbf{M}_i + \mathbf{E}_\alpha\mathbf{R}_i + \mathbf{S}_i\mathbf{E}_\alpha^{''} & \text{if } \alpha \neq i \\ \lfloor \frac{q}{2} \rfloor \mathbf{M}_i + \mathbf{E}_i\mathbf{R}_i & \text{if } \alpha = i \end{cases}$$

We get $\widehat{\mathsf{SK}}.\widehat{\mathbf{C}}_i = \lfloor \frac{q}{2} \rfloor \mathsf{Con}_k(\mathbf{M}_i) + \mathbf{E}_{\widehat{\mathbf{C}}_i}$ where

$$\mathsf{Con}_k(\mathbf{M}_i) = \lfloor \frac{q}{2} \rfloor \begin{pmatrix} \mathbf{M}_i & \cdots & \mathbf{M}_i & \cdots & \mathbf{M}_i \\ \mathbf{O} & \cdots & \mathbf{O} & \cdots & \mathbf{O} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mathbf{O} & \cdots & \mathbf{O} & \cdots & \mathbf{O} \\ \mathbf{O} & \cdots & \mathbf{O} & \cdots & \mathbf{O} \\ \mathbf{O} & \cdots & \mathbf{O} & \cdots & \mathbf{O} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mathbf{O} & \cdots & \mathbf{O} & \cdots & \mathbf{O} \end{pmatrix}$$

and

$$\mathbf{E}_{\widehat{\mathbf{C}}_i} = \begin{pmatrix} \Omega_1 & \cdots & \Omega_i & \cdots & \Omega_k \\ \mathbf{O} & \cdots & \mathbf{O} & \cdots & \mathbf{O} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mathbf{O} & \cdots & \mathbf{O} & \cdots & \mathbf{O} \\ \mathbf{O} & \cdots & \mathbf{O} & \cdots & \mathbf{O} \\ \mathbf{O} & \cdots & \mathbf{O} & \cdots & \mathbf{O} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mathbf{O} & \cdots & \mathbf{O} & \cdots & \mathbf{O} \end{pmatrix}$$

where

$$\Omega_\alpha = \begin{cases} \mathbf{E}_\alpha \mathbf{R}_i + \mathbf{S}_i \mathbf{E}_\alpha^{''} & \text{if } \alpha \neq i \\ \mathbf{E}_i \mathbf{R}_i & \text{if } \alpha = i \end{cases}$$

Thus the extended ciphertext $\widehat{\mathbf{C}}_i$ satisfies $\widehat{\mathsf{SK}}.\widehat{\mathbf{C}}_i = \lfloor \frac{q}{2} \rfloor \mathsf{Con}_k(\mathbf{M}_i) + \mathbf{E}_{\widehat{\mathbf{C}}_i}$ with error

$$\begin{aligned} ||\mathbf{E}_{\widehat{\mathbf{C}}_i}||_\infty = \max\{&||\mathbf{E}_1 \mathbf{R}_i + \mathbf{S}_i \mathbf{E}_1^{''}||_\infty, \ldots, ||\mathbf{E}_i \mathbf{R}_i||_\infty, \ldots, \\ &||\mathbf{E}_k \mathbf{R}_i + \mathbf{S}_i \mathbf{E}_k^{''}||_\infty\} \leq r\beta + rl\beta = r(1+l)\beta = r^{'}\beta. \end{aligned}$$

Since,

$$\begin{aligned} \mathbf{S}_i \mathbf{E}_1^{''} &= \mathbf{S}_i [\mathbf{G}^{-1}((\mathbf{B}_1 - \mathbf{B}_i)^T)]^T (\mathbf{E}_1^{'})^T \\ &= \mathbf{S}_i [\mathbf{G}^{-1} \begin{pmatrix} \mathbf{S}_1' \mathbf{A} + \mathbf{E}_1 - \mathbf{S}_i' \mathbf{A} - \mathbf{E}_i \\ \mathbf{O}_{n \times r} \end{pmatrix}^T]^T (\mathbf{E}_1^{'})^T \\ &= (\mathbf{I}_r | -\mathbf{S}_i') \, [\mathbf{G}^{-1}(\mathbf{K}_{r \times r}^T | \mathbf{O}_{r \times n})]^T (\mathbf{E}_1^{'})^T \end{aligned}$$

when $\mathbf{K} = \mathbf{S}_1' \mathbf{A} + \mathbf{E}_1 - \mathbf{S}_i' \mathbf{A} - \mathbf{E}_i)$ and the last $n \times rl$ part of the binary matrix $[\mathbf{G}^{-1}(\mathbf{K}_{r \times r}^T | \mathbf{O}_{r \times n})]^T$ is zero, we have $||\mathbf{S}_i \mathbf{E}_1^{''}||_\infty \leq rl\beta$. Hence, the extended ciphertext $\widehat{\mathbf{C}}_i$ is $r(1+l)\beta$-noisy extended ciphertext when $\mathbf{C}_i$ is a $r\beta$- noisy fresh ciphertext. ■ (of THEOREM 5).

• $\mathsf{MMFHE.Eval}(params, \mathsf{PK}, \mathcal{C}, \mathbf{C}_1^*, \mathbf{C}_2^*, \ldots, \mathbf{C}_s^*) \rightarrow \mathbf{C}'$: Let $\mathsf{PK} = (pk_{j_1}, pk_{j_2}, \ldots, pk_{j_k}) \in \mathbb{Z}_q^{(n+r) \times rk}$ and $\mathbf{C}_i^* = \mathsf{MMFHE.Enc}(params, \mathsf{PK}, \mathbf{M}_i^*) \in \mathbb{Z}_q^{(n+r)k \times rk}$ where $\mathbf{M}_i^* = \mathsf{Con}_k(\mathbf{M}_i) \in \{0,1\}^{rk \times rk}$, $1 \leq i \leq s$. Given a circuit $\mathcal{C}$ of depth $\leq d$ along with s ciphertexts $\mathbf{C}_1^*, \mathbf{C}_2^*, \ldots, \mathbf{C}_s^* \in \mathbb{Z}_q^{(n+r)k \times rk}$, a server uses PK to output an evaluated ciphertext $\mathbf{C}^{'} = \mathsf{MMFHE.Enc}(params, \mathsf{PK}, \mathcal{C}(\mathbf{M}_1^*, \mathbf{M}_2^*, \ldots, \mathbf{M}_s^*)) \in \mathbb{Z}_q^{(n+r)k \times rk}$ by repeated invocation of the procedure Add and Mult described below in FIGURE 3 and FIGURE 4 when

$$\begin{aligned} \mathbf{C}_1^* &= \mathsf{MMFHE.Enc}(params, \mathsf{PK}, \mathbf{M}_1^*) \in \mathbb{Z}_q^{(n+r)k \times rk}, \\ \mathbf{C}_2^* &= \mathsf{MMFHE.Enc}(params, \mathsf{PK}, \mathbf{M}_2^*) \in \mathbb{Z}_q^{(n+r)k \times rk}, \end{aligned}$$

$\mathsf{PK} = (pk_{j_1}, pk_{j_2}, \ldots, pk_{j_k}) \in \mathbb{Z}_q^{(n+r) \times rk}$ and the corresponding secret key

$$\mathsf{SK}^* = (\mathsf{SK}, \mathbf{O}, \ldots, \mathbf{O})^T \in \mathbb{Z}_q^{kr \times k(n+r)}.$$

• $\mathsf{Add}(\mathbf{C}_1^*, \mathbf{C}_2^*) \rightarrow \mathbf{C}_{\mathsf{add}} = \mathbf{C}_1^* + \mathbf{C}_2^*$: On input two $r'\beta$-noisy ciphertexts $\mathbf{C}_1^*$, $\mathbf{C}_2^*$, the server runs the $\mathsf{Add}(\mathbf{C}_1^*, \mathbf{C}_2^*)$ and outputs $2r'\beta$-noisy ciphertext $\mathbf{C}_{\mathsf{add}} = \mathbf{C}_1^* + \mathbf{C}_2^* \in \mathbb{Z}_q^{(n+r)k \times rk}$ for the message $\mathbf{M}_1^* + \mathbf{M}_2^*$ under PK. Note that $\mathbf{C}_{\mathsf{add}}$ satisfies

$$\begin{aligned}\mathsf{SK}^*.\mathbf{C}_{\mathsf{add}} &= \mathbf{S}^*(\mathbf{C}_1^* + \mathbf{C}_2^*) \\ &= \mathbf{S}^*\mathbf{C}_1^* + \mathbf{S}^*\mathbf{C}_2^* \\ &= \lfloor \tfrac{q}{2} \rfloor \mathbf{M}_1^* + \mathbf{E}_{\mathbf{C}_1^*} + \lfloor \tfrac{q}{2} \rfloor \mathbf{M}_2^* + \mathbf{E}_{\mathbf{C}_2^*} \\ &= \lfloor \tfrac{q}{2} \rfloor (\mathbf{M}_1^* + \mathbf{M}_2^*) + \mathbf{E}_{\mathbf{C}_{\mathsf{add}}}\end{aligned}$$

with error $||\mathbf{E}_{\mathbf{C}_{\mathsf{add}}}||_\infty = ||\mathbf{E}_{\mathbf{C}_1^*} + \mathbf{E}_{\mathbf{C}_2^*}||_\infty \leq 2r'\beta$.

Figure 3: Procedure Add

• $\mathsf{Mult}(\mathbf{C}_1^*, \mathbf{C}_2^*) \rightarrow \mathbf{C}_{\mathsf{mult}} = (\mathbf{C}_1^* \otimes' \mathbf{C}_2^*)_{\mathsf{sc}}$: On input two $r'\beta$-noisy ciphertexts $\mathbf{C}_1^*$, $\mathbf{C}_2^*$, the server runs the $\mathsf{Mult}(\mathbf{C}_1^*, \mathbf{C}_2^*)$ and outputs $\mathbf{C}_{\mathsf{mult}} = (\mathbf{C}_1^* \otimes' \mathbf{C}_2^*)_{\mathsf{sc}}$. The ciphertext $\mathbf{C}_{\mathsf{mult}}$ is a $2r^2k(1+l)\beta + \frac{2}{q}r^2(1+l)^2\beta^2$-noisy ciphertext for the message $\mathbf{M}_1^*\mathbf{M}_2^*$ under PK as by THEOREM 3 in Section 3, we get

$$(\mathsf{SK}^* \otimes \mathsf{SK}^*)_{\mathsf{sr}}(\mathbf{C}_1^* \otimes' \mathbf{C}_2^*)_{\mathsf{sc}} = (\mathsf{SK}^*\mathbf{C}_1^*) \cdot (\mathsf{SK}^*\mathbf{C}_2^*) \bmod q$$

$$\begin{aligned}&\lfloor \tfrac{2}{q}(\mathsf{SK}^*\mathbf{C}_1^*) \cdot (\mathsf{SK}^*\mathbf{C}_2^*) \rceil \\ &= \lfloor \tfrac{2}{q}(\lfloor \tfrac{q}{2} \rfloor \mathbf{M}_1^* + \mathbf{E}_{\mathbf{C}_1^*})(\lfloor \tfrac{q}{2} \rfloor \mathbf{M}_2^* + \mathbf{E}_{\mathbf{C}_2^*}) \rceil \\ &= \lfloor \tfrac{2}{q}(\lfloor \tfrac{q}{2} \rfloor \cdot \lfloor \tfrac{q}{2} \rfloor \mathbf{M}_1^*\mathbf{M}_2^* + \lfloor \tfrac{q}{2} \rfloor \cdot \mathbf{M}_1^*\mathbf{E}_{\mathbf{C}_2^*} \\ &\qquad + \lfloor \tfrac{q}{2} \rfloor \mathbf{M}_2^*\mathbf{E}_{\mathbf{C}_1^*} + \mathbf{E}_{\mathbf{C}_1^*} \cdot \mathbf{E}_{\mathbf{C}_2^*}) \rceil \\ &= \lfloor \tfrac{q}{2} \rfloor \mathbf{M}_1^*\mathbf{M}_2^* + (\mathbf{M}_1^*\mathbf{E}_{\mathbf{C}_2^*} + \mathbf{M}_2^*\mathbf{E}_{\mathbf{C}_1^*} + \tfrac{2}{q}\mathbf{E}_{\mathbf{C}_1^*}\mathbf{E}_{\mathbf{C}_2^*}) \\ &= \lfloor \tfrac{q}{2} \rfloor \mathbf{M}_1^*\mathbf{M}_2^* + \mathbf{E}_{\mathbf{C}_{\mathsf{mult}}}\end{aligned}$$

with error

$$\begin{aligned}&||\mathbf{E}_{\mathbf{C}_{\mathsf{mult}}}||_\infty \\ &\leq ||\mathbf{M}_1^*\mathbf{E}_{\mathbf{C}_2^*} + \mathbf{M}_2^*\mathbf{E}_{\mathbf{C}_1^*} + \tfrac{2}{q}\mathbf{E}_{\mathbf{C}_1^*}\mathbf{E}_{\mathbf{C}_2^*}||_\infty \\ &\leq ||\mathbf{M}_1^*\mathbf{E}_{\mathbf{C}_2^*}||_\infty + ||\mathbf{M}_2^*\mathbf{E}_{\mathbf{C}_1^*}||_\infty + ||\tfrac{2}{q}\mathbf{E}_{\mathbf{C}_1^*}\mathbf{E}_{\mathbf{C}_2^*}||_\infty \\ &\leq 2rk.r'\beta + \tfrac{2}{q}r'^2\beta^2 \\ &= 2r^2k(1+l)\beta + \tfrac{2}{q}r^2(1+l)^2\beta^2 \approx \mathsf{poly}(n,k,q)\beta.\end{aligned}$$

where poly indicates polynomial function. For evaluating d depth circuit, the final error will be $\mathsf{poly}(n,k,q)^d\beta$. Decryption succeeds as long as $\mathsf{poly}(n,k,q)^d\beta < \frac{q}{4}$ Regev (2009).

Figure 4: Procedure Mult

Then

$$\mathsf{SK}^*.\mathbf{C}_1^* = \mathbf{S}^*.\mathbf{C}_1^* = \lfloor \frac{q}{2} \rfloor \mathbf{M}_1^* + \mathbf{E}_{\mathbf{C}_1^*} \in \mathbb{Z}_q^{rk \times rk}$$

$$\mathsf{SK}^*.\mathbf{C}_2^* = \mathbf{S}^*\mathbf{C}_2^* = \lfloor \frac{q}{2} \rfloor \mathbf{M}_2^* + \mathbf{E}_{\mathbf{C}_2^*} \in \mathbb{Z}_q^{rk \times rk}$$

with $||\mathbf{E}_{\mathbf{C}_1^*}||_\infty \leq r'\beta$, $||\mathbf{E}_{\mathbf{C}_2^*}||_\infty \leq r^{'}\beta$ where $r' = r(1+l)$. The procedure Add performs homomorphic addition while the procedure Mult evaluates homomorphic multiplication.

• $\mathsf{MMFHE.Dec}(params, \mathsf{SK}, \mathbf{C}') \rightarrow \mathbf{M}'$: To decrypt a ciphertext $\mathbf{C}'$, the decrypter proceeds as follows:

• $\mathsf{MultProt.PartDec}(params, \mathsf{PK}, sk_{j_i}, \widehat{\mathbf{C}}) \rightarrow \mathbf{P}_i$: On input an extended ciphertext $\widehat{\mathbf{C}} \in \mathbb{Z}_q^{k(n+r)\times kr}$ under the extended public key $\mathsf{PK} = (pk_{j_1}, pk_{j_2}, \ldots, pk_{j_k})$, the i-th user with secret key $sk_{j_i} \in \mathbb{Z}_q^{r\times(n+r)}$ do the following:

– Parses $\widehat{\mathbf{C}}$ into k sub matrices $\widehat{\mathbf{C}}^{(i)} \in \mathbb{Z}_q^{(n+r)\times kr}$ such that

$$\widehat{\mathbf{C}} = \begin{pmatrix} \widehat{\mathbf{C}}^{(1)} \\ \widehat{\mathbf{C}}^{(2)} \\ \vdots \\ \widehat{\mathbf{C}}^{(k)} \end{pmatrix}$$

– Sets $\widehat{sk}_{j_i} = (sk_{j_i}, \mathbf{O}, \ldots, \mathbf{O})^T \in \mathbb{Z}_q^{kr\times(n+r)}$ where $\mathbf{O}$ is a zero matrix of order $r \times (n+r)$.

– Computes $\boldsymbol{\tau}_i = \widehat{sk}_{j_i}.\widehat{\mathbf{C}}^{(i)} \in \mathbb{Z}_q^{kr\times kr}$ and generate partial decryption share $\mathbf{P}_i = \boldsymbol{\tau}_i + \mathbf{E}_i^{sm} \in \chi^{kr\times kr}$ and send it to the other participants involved in the multiparty protocol where the random noise $\mathbf{E}_i^{sm} = (\boldsymbol{\xi}_i, \mathbf{O}, \ldots, \mathbf{O})^T \in \mathbb{Z}_q^{kr\times kr}$ and $\boldsymbol{\xi}_i = (\mathbf{O}, \ldots, \mathbf{O}, \boldsymbol{\zeta}_i, \mathbf{O}, \ldots, \mathbf{O})$ with $\boldsymbol{\zeta}_i \in \chi^{r\times r}$ is the error in the i-th position of $\boldsymbol{\xi}_i$. Here the noise $\mathbf{E}_i^{sm}$ (with designated upper bound) is used to protect information leakage of the secret key from partial decryption $\mathbf{P}_i$.

• $\mathsf{MultProt.FinDec}(\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_k) \rightarrow \widehat{\mathbf{M}} = \mathsf{Con}_k(\mathbf{M})$: On receiving the partial decryption share $\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_k$, each participant a with secret key $sk_{j_a}, 1 \leq a \leq k$ computes the sum $\mathbf{P} = \sum_{i=1}^{k} \mathbf{P}_i$, and recover the message $\widehat{\mathbf{M}} = \mathsf{Con}_k(\mathbf{M})$ by computing $\lceil \frac{2}{q}(\mathbf{P} \mod q) \rfloor \mod 2$.

Figure 5: Procedure $\mathsf{MultProt}(params, (pk_{j_1}, pk_{j_2}, \ldots, pk_{j_k}), (sk_{j_1}, sk_{j_2}, \ldots, sk_{j_k}), \widehat{\mathbf{C}})$ that consist of PartDec for partial decryption and FinDec for the final decryption.

<u>*Case 1*</u>: Let $\mathbf{C}^{'}$ be a *fresh ciphertext* for the message $\mathbf{M}$ under the public key $\mathsf{PK} = pk$ i.e.

$$\mathbf{C}^{'} = (\mathbf{C}, \mathsf{Aux}) \leftarrow \mathsf{MMFHE.Enc}(params, pk, \mathbf{M}) \in \mathbb{Z}_q^{(n+r)\times r}.$$

Let $\mathsf{SK} = sk = \mathbf{S} \in \mathbb{Z}_q^{r\times(n+r)}$ be the secret key corresponding to the public key $pk = \mathbf{B}$. Then Aux and $\mathbf{C}$ satisfy

$$sk\mathbf{C} = \mathbf{SC} = \lfloor \frac{q}{2} \rfloor \mathbf{M} + \mathbf{E}_{\mathbf{C}} \in \mathbb{Z}_q^{r\times r}$$

with $||\mathbf{E}_{\mathbf{C}}||_\infty \leq r\beta$. The decrypter uses his secret key $sk = \mathbf{S}$ and recovers the message

$\mathbf{M}$ by computing $\lceil \frac{2}{q}(\mathbf{SC} \mod q) \rfloor \mod 2$. Note that

$$\begin{aligned} &\lceil \tfrac{2}{q}(\mathbf{SC} \mod q) \rfloor \mod 2 \\ &\simeq \mathbf{M} + \lfloor \tfrac{q}{2} \rfloor \mathbf{E_C} \mod q \mod 2 \\ &\simeq \mathbf{M} \end{aligned}$$

provided $||\mathbf{E_C}||_\infty < \frac{q}{4}$ following the work of Regev Regev (2009), yielding $m\beta < \frac{q}{4}$ i.e. $q > 4m\beta$.

Case 2: Let $\mathbf{C}^{'}$ be an *extended ciphertext* under the public key $\mathsf{PK} = (pk_{j_1}, pk_{j_2}, \ldots, pk_{j_k})$ i.e.

$$\mathbf{C}^{'} = \widehat{\mathbf{C}}_i = \mathsf{MMFHE.Ext}(params, \mathsf{PK}, pk_{j_i}, \mathbf{C}_i, \mathsf{Aux}_i)$$

where $\mathbf{C}_i$ is a fresh ciphertext generated as

$$(\mathbf{C}_i, \mathsf{Aux}_i) \leftarrow \mathsf{MMFHE.Enc}(params, pk_{j_i}, \mathbf{M}_i).$$

Then $\widehat{\mathbf{C}}_i = \mathsf{MMFHE.Enc}(params, \mathsf{PK}, \mathsf{Con}_k(\mathbf{M}_i))$. Let $\widehat{\mathsf{SK}} = (\mathsf{SK}, \mathbf{O}, \ldots, \mathbf{O})^T \in \mathbb{Z}_q^{kr \times k(n+r)}$ where $\mathsf{SK} = (sk_{j_1}, sk_{j_2}, \ldots, sk_{j_k})$ is the secret key corresponding to PK. To decrypt $\mathbf{C}^{'}$, k users with their respective secret keys $sk_{j_1}, sk_{j_2}, \ldots, sk_{j_k}$ apply the threshold decryption technique by running the multi-party protocol described as procedure $\mathsf{MultProt} = (\mathsf{PartDec}, \mathsf{FinDec})$ in FIGURE 5 among themselves and recover the message $\mathbf{M}_i \in \{0,1\}^{r \times r}$.

Case 3: Let $\mathbf{C}^{'}$ be an *evaluated ciphertext* under the public key $\mathsf{PK} = (pk_{j_1}, pk_{j_2}, \ldots, pk_{j_k})$,

$$\mathbf{C}^{'} \leftarrow \mathsf{MMFHE.Eval}(params, \mathsf{PK}, \mathcal{C}, \mathbf{C}_1^*, \mathbf{C}_2^*, \ldots, \mathbf{C}_s^*)$$

where

$$\mathbf{C}_i^* = \mathsf{MMFHE.Enc}(params, \mathsf{PK}, \mathbf{M}_i^*),$$
$$\mathbf{M}_i^* = \mathsf{Con}_k(\mathbf{M}_i) \in \mathcal{M}, 1 \leq i \leq s$$

and $\mathcal{C}$ is a circuit of depth at most d. Let

$$\mathsf{SK}^* = (\mathsf{SK}, \mathbf{O}, \ldots, \mathbf{O})^T \in \mathbb{Z}_q^{kr \times k(n+r)}$$

be the secret key corresponding to PK. To decrypt $\mathbf{C}^{'}$, k users with their respective secret keys $sk_{j_1}, sk_{j_2}, \ldots, sk_{j_k}$ invoke the multi-party protocol $\mathsf{MultProt} = (\mathsf{PartDec}, \mathsf{FinDec})$ as described in FIGURE 5 among themselves and recover the message $\mathbf{M}' = \mathcal{C}(\mathbf{M}_1^*, \mathbf{M}_2^*, \ldots, \mathbf{M}_s^*)$.

5 Security Analysis

THEOREM 6. *Our scheme* $\mathsf{MMFHE} = (\mathsf{Setup}, \mathsf{Keygen}, \mathsf{Enc}, \mathsf{Ext}, \mathsf{Eval}, \mathsf{Dec})$ *described in Section 6 is IND-CPA secure as per DEFINITION 11 under the decisional LWE assumption.*

Proof: Since the public algorithms $\mathsf{MMFHE.Eval}$ and $\mathsf{MMFHE.Ext}$ can be run by the adversary, the security of our scheme $\mathsf{MMFHE} = (\mathsf{Setup}, \mathsf{Keygen}, \mathsf{Enc}, \mathsf{Ext}, \mathsf{Eval}, \mathsf{Dec})$ depends on the security of the underlying basic encryption scheme $\mathsf{MMFHEbasic} =$

(Setup, Keygen, Enc, Dec) where MMFHEbasic.Setup= MMFHE.Setup, MMFHEbasic.Keygen = MMFHE.Keygen, MMFHEbasic.Enc = MMFHE.Enc and MMFHEbasic.Dec = MMFHE.Dec It is secure under the hardness of decisional LWE assumption. Our goal is to show that the advantage of any PPT adversary $\mathcal{A}$ to break the IND-CPA-security (see $\mathsf{Expt}_{\mathcal{A}}^{\mathsf{MMFHEbasic}}(1^{\lambda}, b)$ defined in the Section 3) of our scheme MMFHEbasic is negligible. We proceed by considering the following hybrid experiments:

$\mathbf{H}_0$: This is the real IND-CPA game $\mathsf{Expt}_{\mathcal{A}}^{\mathsf{MMFHEbasic}}(1^{\lambda}, b)$ played between a challenger Ch and an adversary $\mathcal{A}$. More concretely, we have the following

1. The challenger Ch generates $\mathsf{MMFHEbasic.Setup}(1^{\lambda}, 1^{d}) \rightarrow params$ and runs $\mathsf{MMFHEbasic.Keygen}(params)$ to obtain key pairs (pk, sk) where $params = (n, q, \chi, \beta, r, N, \mathbf{A})$, integer n represents lattice dimension, integer r is the number of message bits, integer q is modulus, $\chi = \chi(\lambda, d)$ is a β-bounded error distribution (discrete Gaussian distribution) over $\mathbb{Z}_q$ and a common random matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times r}$ then the public key

$$pk = \mathbf{B} = \begin{pmatrix} \mathbf{S}'\mathbf{A} + \mathbf{E} \\ \mathbf{A} \end{pmatrix} \in \mathbb{Z}_q^{(n+r)\times r},$$

the secret key

$$sk = \mathbf{S} = (\mathbf{I}_r | -\mathbf{S}') \in \mathbb{Z}_q^{r \times (n+r)},$$

$\mathbf{E} \in \chi^{r \times r}$ is the error matrix and $\mathbf{S}' \in \mathbb{Z}_q^{r \times n}$ is sampled randomly. The challenger Ch sends pk to the adversary $\mathcal{A}$ and keeps sk secret to itself.

2. The adversary $\mathcal{A}$ returns a pair of messages $\mathbf{M}_0, \mathbf{M}_1 \in \{0,1\}^{r \times r}$ to Ch.

3. The challenger Ch chooses a random bit $b \in \{0,1\}$ and sends the challenge ciphertext $\mathbf{C}_b$ to $\mathcal{A}$ where $\mathbf{C}_b$ is constructed as follows:

   – To encrypt a message $\mathbf{M}_b \in \{0,1\}^{r \times r}$, Ch chooses a binary symmetric matrix $\mathbf{R}_b \in \{0,1\}^{r \times r}$, error matrix $\mathbf{E}_b^{'} \in \chi^{r \times rl}$ and computes

   $$(\mathbf{C}_b, \mathsf{Aux}_b) = \mathsf{MMFHEbasic.Enc}(params, pk, \mathbf{M}_b)$$

   when

   $$\mathbf{C}_b = \lfloor \frac{q}{2} \rfloor \begin{pmatrix} \mathbf{M}_b \\ \mathbf{O} \end{pmatrix} + \mathbf{B}\mathbf{R}_b) \text{ and}$$

   $$\mathsf{Aux}_b = (\mathbf{R}_b\mathbf{G} + \mathbf{E}_b^{'} \mod q)$$

4. Finally, $\mathcal{A}$ returns to Ch a guess bit $b^{'}$ for b. The game $\mathsf{Expt}_{\mathcal{A}}^{\mathsf{MMFHEbasic}}(1^{\lambda}, b)$ outputs 1 if $b^{'} = b$ and 0 otherwise.

Let us define $\mathbf{T}_0$ to be the event that $b^{'} = b$ in the hybrid experiment $\mathbf{H_0}$. To express $\mathcal{A}$'s advantage $\mathsf{Adv}_{\mathcal{A},\mathsf{MMFHEbasic}}^{\mathsf{IND-CPA}}$ for the game $\mathbf{H}_i$ where $i = 0, 1, 2$ according to the DEFINITION 11, we use the notation $\mathsf{Adv}_{\mathcal{A},\mathsf{MMFHEbasic},\mathbf{H_i}}^{\mathsf{IND-CPA}}$. Therefore, $\mathcal{A}$'s advantage from $\mathbf{H_0}$ is $\mathsf{Adv}_{\mathcal{A},\mathsf{MMFHEbasic},\mathbf{H_0}}^{\mathsf{IND-CPA}} = |\Pr[\mathbf{T}_0] - \frac{1}{2}|$.

$\mathbf{H}_1$: This hybrid experiment between a challenger Ch and an adversary $\mathcal{A}$ is identical to that in hybrid $\mathbf{H_0}$ except the following fact:

- To encrypt the message $\mathbf{M}_b$, the challenger Ch replaces the public key $\mathbf{B} = \begin{pmatrix} \mathbf{S'A} + \mathbf{E} \\ \mathbf{A} \end{pmatrix} \in \mathbb{Z}_q^{(n+r)\times r}$ by $\overline{\mathbf{B}} = \begin{pmatrix} \mathbf{U} \\ \mathbf{A} \end{pmatrix} \in \mathbb{Z}_q^{(n+r)\times r}$ where the matrix $\mathbf{U}$ is uniformly chosen from $\mathbb{Z}_q^{r\times r}$. Then the challenge ciphertext

$$\mathbf{C}_b = \lfloor \frac{q}{2} \rfloor \begin{pmatrix} \mathbf{M}_b \\ \mathbf{O} \end{pmatrix} + \overline{\mathbf{B}}\mathbf{R}_b) \text{ and}$$

$$\mathsf{Aux}_b = (\mathbf{R}_b\mathbf{G} + \mathbf{E}_b^{'} \mod q)$$

We claim that $\mathbf{B'} = \mathbf{S'A} + \mathbf{E} \in \mathbb{Z}_q^{r\times r}$ and $\mathbf{U} \in \mathbb{Z}_q^{r\times r}$ are computationally indistinguishable under decisional LWE assumption as proved in the CLAIM 1 below making $\mathbf{B}$ and $\overline{\mathbf{B}}$ are computationally indistinguishable.
Let $\mathbf{T}_1$ be the event that $b' = b$ in the hybrid experiment $\mathbf{H}_1$. Utilizing CLAIM 1, $\mathbf{B}$ and $\overline{\mathbf{B}}$ are computationally indistinguishable under the decisional LWE assumption. Let, this computational distance is δ where δ is negligible in the security parameter λ. Therefore, $|\Pr[\mathbf{T}_0] - \Pr[\mathbf{T}_1]| \leq \delta$. Consequently, we get the advantage $\mathsf{Adv}^{\mathsf{IND-CPA}}_{\mathcal{A},\mathsf{MMFHEbasic},\mathbf{H_0}} \leq \mathsf{Adv}^{\mathsf{IND-CPA}}_{\mathcal{A},\mathsf{MMFHEbasic},\mathbf{H_1}} + \delta$.

$\mathbf{H}_2$: This hybrid experiment between a challenger Ch and an adversary $\mathcal{A}$ is identical to that in the hybrid $\mathbf{H}_1$ except the following change in the challenge ciphertext $\mathbf{C}_b$:

- To encrypt the message $\mathbf{M}_b$, the challenger Ch uses uniformly random matrix $\mathbf{U}_b \in \mathbb{Z}_q^{(n+r)\times r}$ and replaces the matrix $\overline{\mathbf{B}}\mathbf{R}_b$ in $\mathbf{C}_b$ by $\mathbf{U}_b$. Thus $\mathbf{C}_b = \lfloor \frac{q}{2} \rfloor \begin{pmatrix} \mathbf{M}_b \\ \mathbf{O} \end{pmatrix} + \mathbf{U}_b \in \mathbb{Z}_q^{(n+r)\times r}$.

Let $\mathbf{T}_2$ to be the event that $b = b^{'}$ in the hybrid experiment $\mathbf{H_2}$. By left over hash lemma, $\overline{\mathbf{B}}\mathbf{R}_b$ is indistinguishable from uniformly random matrix $\mathbf{U}_b$. Let this statistical distance is δ', where δ' is negligible in λ. Therefore, $|\Pr[\mathbf{T}_1] - \Pr[\mathbf{T}_2]| \leq \delta'$. Accordingly, $\mathsf{Adv}^{\mathsf{IND-CPA}}_{\mathcal{A},\mathsf{MMFHEbasic},\mathbf{H_1}} \leq \mathsf{Adv}^{\mathsf{IND-CPA}}_{\mathcal{A},\mathsf{MMFHEbasic},\mathbf{H_2}} + \delta'$.

We note that, in the hybrid experiment $\mathbf{H}_2$, both the public key and the ciphertext are uniformly random and independent of the message. Hence, $\Pr[\mathbf{T}_2] = \frac{1}{2}$. Therefore, $\mathsf{Adv}^{\mathsf{IND-CPA}}_{\mathcal{A},\mathsf{MMFHEbasic},\mathbf{H}_2} = |\Pr[\mathbf{T}_2] - \frac{1}{2}| = 0$. Summarizing all the above results, we get

$$\begin{aligned}
\mathsf{Adv}^{\mathsf{IND-CPA}}_{\mathcal{A},\mathsf{MMFHEbasic},\mathbf{H}_0} &= |\Pr[\mathbf{T}_0] - \frac{1}{2}| \\
&= |\Pr[\mathbf{T}_0] - \Pr[\mathbf{T}_2]| \\
&= |\Pr[\mathbf{T}_0] - \Pr[\mathbf{T}_1] + \Pr[\mathbf{T}_1] - \Pr[\mathbf{T}_2]| \\
&\leq |\Pr[\mathbf{T}_0] - \Pr[\mathbf{T}_1]| + |\Pr[\mathbf{T}_1] - \Pr[\mathbf{T}_2]| \\
&\leq \delta + \delta' = \mathsf{negl}(\lambda)
\end{aligned}$$

Hence, adversary $\mathcal{A}$ can break the IND-CPA security of MMFHEbasic with at most negligible advantage. This completes the proof. ■(of THEOREM 6)

CLAIM 1. *Let $\mathbf{A} \in \mathbb{Z}_q^{n\times r}$ be a common public matrix, $\mathbf{S'} \in \mathbb{Z}_q^{r\times n}$ be a secret matrix and $\mathbf{E} \in \chi^{r\times r}$ is an error matrix chosen from the Gaussian distribution χ. Then $\mathbf{B'} = \mathbf{S'A} + \mathbf{E} \in \mathbb{Z}_q^{r\times r}$ and $\mathbf{U} \in \mathbb{Z}_q^{r\times r}$ are computationally indistinguishable under the decisional LWE assumption.*

Proof: Let $\mathbf{S}' = (\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_r)^T \in \mathbb{Z}_q^{r \times n}$ where each $\mathbf{s}_i$ is a row vector and $\mathbf{s}_i \in \mathbb{Z}_q^n$. Then $\mathbf{B}' = \mathbf{S}'\mathbf{A} + \mathbf{E} = (\mathbf{s}_1\mathbf{A} + \mathbf{e}_1, \mathbf{s}_2\mathbf{A} + \mathbf{e}_2, \ldots, \mathbf{s}_r\mathbf{A} + \mathbf{e}_r)^T = (\mathbf{b}_1, \mathbf{b}_2, \ldots \mathbf{b}_r)^T$ where $\mathbf{E} = (\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_r)^T \in \chi^{r \times r}$ and $\mathbf{e}_i$ is a row vector. We now define two distributions as follows:

– $\mathbf{X}$ is a distribution on $r \times r$ matrix $(\mathbf{b}_1, \mathbf{b}_2, \cdots, \mathbf{b}_r)^T \in \mathbb{Z}_q^{r \times r}$ where $\mathbf{b}_i = \mathbf{s}_i\mathbf{A} + \mathbf{e}_i, \mathbf{s}_i \in \mathbb{Z}_q^r$ and $\mathbf{e}_i \in \chi^r$ are row vectors $\forall i \in [r]$.

– $\mathbf{Y}$ is uniform distribution on $\mathbb{Z}_q^{r \times r}$

We show that $\mathbf{X}$ and $\mathbf{Y}$ are computationally indistinguishable under the decisional LWE assumption which represents that $\mathbf{B}'$ and $\mathbf{U}$ are computationally indistinguishable.
If possible let $\mathcal{A}$ be a PPT adversary that can distinguish $\mathbf{X}$ from $\mathbf{Y}$ with non-negligible advantage. Let $1 \leq i \leq r$, we introduce intermediate distribution $\mathbf{X}_i$ given by $\left(\mathbf{b}'_1, \cdots, \mathbf{b}'_{i-1}, \mathbf{b}_i, \cdots, \mathbf{b}_r\right)^T$ where $\mathbf{b}_j = \mathbf{s}_j\mathbf{A} + \mathbf{e}_j, i \leq j \leq r$ and $\mathbf{b}'_k$ are uniformly chosen from $\mathbb{Z}_q^r$ for $1 \leq k \leq i-1$. Hence $\mathbf{X}_1 = \mathbf{X}$ and $\mathbf{X}_{r+1} = \mathbf{Y}$. By assumption, let $\mathcal{A}$ can distinguish $\mathbf{X}$ from $\mathbf{Y}$ with non-negligible advantage δ. Then by standard hybrid argument, $\mathcal{A}$ can distinguish $\mathbf{X}_i$ from $\mathbf{X}_{i+1}$ for $1 \leq i \leq r$ with non-negligible advantage $\frac{\delta}{r}$. Consequently, $\mathcal{A}$ will produce an LWE distinguisher: Given the decisional LWE instances $(\mathbf{A}, \mathbf{y})$, the distinguisher $\mathcal{D}$ samples $\mathbf{b}'_1, \mathbf{b}'_2, \ldots, \mathbf{b}'_{i-1}$ uniformly from $\mathbb{Z}_q^r$ and $\mathbf{b}_{i+1}, \mathbf{b}_{i+2}, \ldots, \mathbf{b}_r$ are as sampled in distribution $\mathbf{X}$. Then $\mathcal{D}$ calls $\mathcal{A}$ on $(\mathbf{b}'_1, \ldots, \mathbf{b}'_{i-1}, \mathbf{y}, \ldots, \mathbf{b}_r)^T$. If $\mathcal{A}$ can distinguish it, then by using $\mathcal{A}$ as a subroutine $\mathcal{D}$ breaks the decisional LWE. As decisional LWE assumption holds so there does not exist any such distinguisher $\mathcal{D}$ that can distinguish the distribution $\mathbf{X}$ from the distribution $\mathbf{Y}$. Therefore, $\mathbf{X}$ and $\mathbf{Y}$ are computationally indistinguishable under the decisional LWE assumption. Consequently, $\mathbf{B}'$ and $\mathbf{U}$ are also computationally indistinguishable. ■ (of CLAIM 1)

6 Conclusion

In this paper, we have presented a multi-key FHE for multi-bit messages based on the hardness of decisional LWE assumption. Our construction enhances the efficiency of computation by setting each entry of the plaintext matrix as message slot. To extend a ciphertext under an additional key, we have employed the efficient homomorphic linear combination algorithm and encrypts directly a random matrix used for encryption to reduce the computational overhead of the extension algorithm. Furthermore, we have shown that the size of the secret key, public key and ciphertext of our design are favorably better than the existing similar schemes.

References

Agrawal, S., Boneh, D., and Boyen, X. (2010a). Efficient lattice (H) IBE in the standard model. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 553–572. Springer.

Agrawal, S., Boneh, D., and Boyen, X. (2010b). Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical IBE. In *Annual Cryptology Conference*, pages 98–115. Springer.

Agrawal, S., Clear, M., Frieder, O., Garg, S., O'Neill, A., and Thaler, J. (2020). Ad hoc multi-input functional encryption. In *11th Innovations in Theoretical Computer Science Conference (ITCS 2020)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

Ananth, P., Jain, A., Jin, Z., and Malavolta, G. (2020). Multikey FHE in the Plain Model. *IACR Cryptol. ePrint Arch.*, 2020:180.

Ananth, P., Jain, A., Naor, M., Sahai, A., and Yogev, E. (2016). Universal constructions and robust combiners for indistinguishability obfuscation and witness encryption. In *Annual International Cryptology Conference*, pages 491–520. Springer.

Ananth, P., Jain, A., and Sahai, A. (2017). Robust transforming combiners from indistinguishability obfuscation to functional encryption. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 91–121. Springer.

Asharov, G., Jain, A., López-Alt, A., Tromer, E., Vaikuntanathan, V., and Wichs, D. (2012). Multiparty computation with low communication, computation and interaction via threshold FHE. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 483–501. Springer.

Biswas, C. and Dutta, R. (2021). Dynamic multi-key fhe in symmetric key setting from lwe without using common reference matrix. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–14.

Boyle, E., Gilboa, N., and Ishai, Y. (2016). Breaking the circuit size barrier for secure computation under DDH. In *Annual International Cryptology Conference*, pages 509–539. Springer.

Boyle, E., Gilboa, N., and Ishai, Y. (2017). Group-based secure computation: optimizing rounds, communication, and computation. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 163–193. Springer.

Brakerski, Z. (2012). Fully Homomorphic Encryption without Modulus Switching from Classical GapSVP. In *CRYPTO*, volume 7417, pages 868–886. Springer.

Brakerski, Z., Döttling, N., Garg, S., and Malavolta, G. (2019). Leveraging linear decryption: Rate-1 fully-homomorphic encryption and time-lock puzzles. In *Theory of Cryptography Conference*, pages 407–437. Springer.

Brakerski, Z., Gentry, C., and Halevi, S. (2013). Packed ciphertexts in LWE-based homomorphic encryption. In *International Workshop on Public Key Cryptography*, pages 1–13. Springer.

Brakerski, Z., Gentry, C., and Vaikuntanathan, V. (2014). (leveled) fully homomorphic encryption without bootstrapping. *ACM Transactions on Computation Theory (TOCT)*, 6(3):13.

Brakerski, Z. and Perlman, R. (2016). Lattice-based fully dynamic multi-key FHE with short ciphertexts. In *Annual Cryptology Conference*, pages 190–213. Springer.

Brakerski, Z. and Vaikuntanathan, V. (2011). Fully homomorphic encryption from ring-LWE and security for key dependent messages. In *Annual cryptology conference*, pages 505–524. Springer.

Brakerski, Z. and Vaikuntanathan, V. (2014). Efficient fully homomorphic encryption from (standard) lwe. *SIAM Journal on Computing*, 43(2):831–871.

Cash, D., Hofheinz, D., Kiltz, E., and Peikert, C. (2012). Bonsai trees, or how to delegate a lattice basis. *Journal of cryptology*, 25(4):601–639.

Chen, L., Zhang, Z., and Wang, X. (2017). Batched Multi-hop Multi-key FHE from Ring-LWE with Compact Ciphertext Extension. In *Theory of Cryptography Conference*, pages 597–627. Springer.

Clear, M. and McGoldrick, C. (2014). Bootstrappable identity-based fully homomorphic encryption. In *International Conference on Cryptology and Network Security*, pages 1–19. Springer.

Clear, M. and McGoldrick, C. (2015). Multi-identity and multi-key leveled FHE from learning with errors. In *Annual Cryptology Conference*, pages 630–656. Springer.

Clear, M. and McGoldrick, C. (2016). Attribute-based fully homomorphic encryption with a bounded number of inputs. In *International Conference on Cryptology in Africa*, pages 307–324. Springer.

Coron, J.-S., Mandal, A., Naccache, D., and Tibouchi, M. (2011). Fully homomorphic encryption over the integers with shorter public keys. In *Annual Cryptology Conference*, pages 487–504. Springer.

Dodis, Y., Halevi, S., Rothblum, R. D., and Wichs, D. (2016). Spooky encryption and its applications. In *Annual International Cryptology Conference*, pages 93–122. Springer.

Gentry, C. (2009). Fully homomorphic encryption using ideal lattices. In *STOC*, volume 9, pages 169–178.

Gentry, C., Peikert, C., and Vaikuntanathan, V. (2008). Trapdoors for hard lattices and new cryptographic constructions. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 197–206. ACM.

Gentry, C., Sahai, A., and Waters, B. (2013). Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In *Advances in Cryptology–CRYPTO 2013*, pages 75–92. Springer.

Halevi, S., Ishai, Y., Jain, A., Komargodski, I., Sahai, A., and Yogev, E. (2017). Non-interactive multiparty computation without correlated randomness. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 181–211. Springer.

Hiromasa, R., Abe, M., and Okamoto, T. (2016). Packing messages and optimizing bootstrapping in GSW-FHE. *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, 99(1):73–82.

Hoffstein, J., Pipher, J., and Silverman, J. H. (1998). NTRU: A ring-based public key cryptosystem. In *International Algorithmic Number Theory Symposium*, pages 267–288. Springer.

Kim, E., Lee, H.-S., and Park, J. (2018). Towards round-optimal secure multiparty computations: Multikey FHE without a CRS. In *Australasian Conference on Information Security and Privacy*, pages 101–113. Springer.

Li, Z., Ma, C., Morais, E., and Du, G. (2016). Multi-bit Leveled Homomorphic Encryption via Dual. LWE-Based. In *International Conference on Information Security and Cryptology*, pages 221–242. Springer.

Li, Z., Ma, C., and Zhou, H. (2018). Multi-key FHE for multi-bit messages. *Science China Information Sciences*, 61(2):029101.

López-Alt, A., Tromer, E., and Vaikuntanathan, V. (2012). 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*, pages 1219–1234. ACM.

Lyubashevsky, V., Peikert, C., and Regev, O. (2013). On ideal lattices and learning with errors over rings. *Journal of the ACM (JACM)*, 60(6):43.

Malavolta, G. and Thyagarajan, S. A. K. (2019). Homomorphic time-lock puzzles and applications. In *Annual International Cryptology Conference*, pages 620–649. Springer.

Micciancio, D. (2018). On the hardness of learning with errors with binary secrets. *Theory of Computing*, 14(1):1–17.

Mukherjee, P. and Wichs, D. (2016). Two round multiparty computation via multikey FHE. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 735–763. Springer.

Pal, T. and Dutta, R. (2020). Chosen-ciphertext secure multi-identity and multi-attribute pure fhe. In *Cryptology and Network Security*. Springer International Publishing.

Peikert, C. and Shiehian, S. (2016). Multi-key FHE from LWE, Revisited. In *Theory of Cryptography Conference*, pages 217–238. Springer.

Regev, O. (2009). On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM (JACM)*, 56(6):34.

Smart, N. P. and Vercauteren, F. (2010). Fully Homomorphic Encryption with Relatively Small Key and Ciphertext Sizes. In *Public Key Cryptography*, volume 6056, pages 420–443. Springer.

Wang, B., Wang, X., and Xue, R. (2017). Leveled FHE with Matrix Message Space. In *International Conference on Information Security and Cryptology*, pages 260–277. Springer.