SCIENCE CHINA
Information Sciences

• LETTER •

February 2018, Vol. 61 029101:1–029101:3
https://doi.org/10.1007/s11432-017-9206-y

Multi-key FHE for multi-bit messages

Zengpeng LI[1,2*], Chunguang MA[1*] & Hongsheng ZHOU[2*]

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

Received 2 March 2017/Revised 31 May 2017/Accepted 19 July 2017/Published online 3 January 2018

Citation Li Z P, Ma C G, Zhou H S. Multi-key FHE for multi-bit messages. Sci China Inf Sci, 2018, 61(2): 029101, https://doi.org/10.1007/s11432-017-9206-y

Dear editor,

A fully homomorphic encryption (FHE) allows us to perform any complex computation on encrypted data without decryption. Since the breakthrough by Gentry [1], researchers have made significant efforts to improve the construction of the FHE schemes. More specifically, because of well-understood assumptions such as learning with errors (LWE), the FHE constructions are now efficient and nearly practical. Concurrently, these constructions also support rich functionalities (e.g., multi-key FHE, hereafter referred as MFHE). Notably, in [2], the authors introduced new mathematical techniques such as key and modulus switching that resulted in highly efficient leveled FHE schemes. Furthermore, unexpectedly, these constructions can also be based on the well understood LWE. Subsequently, numerous attempts have been made to achieve performances that are orders of magnitude faster than before. Techniques following the above-mentioned approach have recently been further improved [3–6]. In [7], the authors demonstrated an MFHE scheme based on number theory research unit (or NTRU). Lately, this construction method has been enhanced, and in [8, 9], the authors could construct MFHE schemes based on the more standard LWE assumptions. Subsequently, Mukherjee and Wichs [9] formulated an important "linear combination procedure" (referred as LCP) that could offer auxiliary information to a player and assist him to decrypt the ciphertexts from other players [9]. Our current efforts are directed along this approach. In this study, we examine MFHE for multi-bit messages. As discussed above, we observe that in the Mukherjee-Wichs scheme (i.e., multi-key and single-bit FHE), the core of the LCP construction is a variant of the "single-key and multi-bit" FHE scheme that uses the Gentry-Sahai-Waters (GSW) [3] encryption algorithm to encrypt each entry of a random matrix (or bit-by-bit). This apparently introduced an undesirable overhead. This leads to the following natural question:

Is it possible to improve the LCP to develop a more efficient MFHE scheme for encrypting multi-bit messages?

In our study, to avoid encrypting each entry of random matrix $\boldsymbol{R}$, we improve the "LCP" construction in the Mukherjee-Wichs scheme by encrypting random diagonal matrix $\boldsymbol{R}$ directly; this can be viewed as a variant of multi-bit FHE (referred as mFHE).

Remark 1. The definition of FHE, related notations, and analysis of the correctness and security are provided in Appendices A and B. Here we focus on the construction of the scheme.

Improved LCP. Mukherjee and Wichs [9] constructed a two-round multi-party computation via the MFHE scheme. However, for encrypting multi-bits with their scheme, the encryption algorithm may be required to be repeated numerous times.

---

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

Most notably, we utilize the mFHE scheme1) to achieve the MFHE scheme for multi-bit messages (referred as mMFHE). Below we present our "improved LCP" (hereafter referred as iLCP). The key concept is to use a variant of the mFHE scheme to improve the original LCP.

iLCP *construction.* The encryption of random diagonal matrix $\widetilde{\boldsymbol{R}}$ is at the core of the iLCP2). Before describing our iLCP, we first denote $\boldsymbol{C}_{\#}$ and $\boldsymbol{C}_{\&}$ as the ciphertext of $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{R}}$, respectively.

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

Take parameters λ and L as input. We denote $\chi = \chi(\lambda)$, $n = n(\lambda)$, and $m = m(\lambda, L) = \mathcal{O}(n \log q)$, so that the (m, n, q, χ)-LWE assumption achieves at least 2^{λ} security against known attacks. Subsequently, choose parameter $t = \mathcal{O}(\log(n))$ (as the number of secret keys). Output params=(n, q, χ, m), where $\ell = \lfloor \log q \rfloor + 1$.

$(\mathrm{pk}, \mathrm{sk}) \leftarrow$ mFHE.KeyGen(params):

(1) Output pk $:= \boldsymbol{A} = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_t, \boldsymbol{B}] \in \mathbb{Z}_q^{m \times n}$, where $\boldsymbol{b}_j = \boldsymbol{B}\boldsymbol{t}_j + \boldsymbol{e}_j \pmod q \in \mathbb{Z}_q^{m \times 1}$ for common public matrix $\boldsymbol{B} \in \mathbb{Z}_q^{m \times (n-t)}$ and secret vector $\boldsymbol{t}_j = (t_{j,1}, \ldots, t_{j,n}) \in \mathbb{Z}_q^{(n-t) \times 1}$.

(2) Output sk $:= \boldsymbol{S} = [\boldsymbol{s}_1, \ldots, \boldsymbol{s}_t] \in \mathbb{Z}_q^{n \times t}$, where $\boldsymbol{s}_j = [0, \ldots, 1, \ldots, 0 \mid -\boldsymbol{t}_j^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{Z}_q^{n \times 1}$ for $j \in [t]$. We observe that $\boldsymbol{A} \cdot \boldsymbol{s}_j = \boldsymbol{e}_j$ and $\boldsymbol{A} \cdot \boldsymbol{S} = [\boldsymbol{e}_1, \ldots, \boldsymbol{e}_t]$.

$\boldsymbol{C}_{\&} \leftarrow$ mFHE.Encode$(\boldsymbol{G}, \widetilde{\boldsymbol{R}})$:

We first encode random diagonal matrix $\widetilde{\boldsymbol{R}} \in \{0,1\}^{m \times m}$ by using gadget matrix $\boldsymbol{G} \in \mathbb{Z}_q^{m \times m\ell}$. This yields the following encoded message: $\boldsymbol{C}_{\&} := \widetilde{\boldsymbol{R}} \cdot \boldsymbol{G} \pmod q \in \mathbb{Z}_q^{m \times m\ell}$.

$\widetilde{\mathcal{V}} \leftarrow$ mFHE.Extend$(\mathrm{pk}^{(i)}, \mathrm{pk}^{(j)})$:

(1) We denote the public key and secret key of an i-th User as $\mathrm{pk}^{(i)}$ and $\mathrm{sk}^{(i)}$, respectively. We note that $\boldsymbol{S}^{(i)}$ is the secret key matrix of an i-th User and $\boldsymbol{s}_j^{(i)}$ is the j-th row vector of an i-th User, where $i \in [N]$ and $j \in [t]$.

(2) For any two players User1 and User2, we parse $\mathrm{pk}^{(1)} = [\boldsymbol{b}_1^{(1)}, \ldots, \boldsymbol{b}_t^{(1)} \mid \boldsymbol{B}]$, $\mathrm{pk}^{(2)} = [\boldsymbol{b}_1^{(2)}, \ldots, \boldsymbol{b}_t^{(2)} \mid \boldsymbol{B}]$ over $\mathbb{Z}_q^{m \times n}$.

(3) Output connection vector $\widetilde{\mathcal{V}} = [\mathcal{V} \mid \boldsymbol{0}] \in \mathbb{Z}_q^{m \times n}$ by combining $\mathcal{V} = [(\boldsymbol{b}_1^{(2)} - \boldsymbol{b}_1^{(1)}), (\boldsymbol{b}_2^{(2)} - \boldsymbol{b}_2^{(1)}), \ldots, (\boldsymbol{b}_t^{(2)} - \boldsymbol{b}_t^{(1)})] \in \mathbb{Z}_q^{m \times t}$ with $(m \times (n-t))$-dimension $\boldsymbol{0}$ together.

$\boldsymbol{X} \leftarrow$ mFHE.iLCP$(\boldsymbol{C}_{\&}, \widetilde{\mathcal{V}})$:

To obtain auxiliary information $\boldsymbol{X} \in \mathbb{Z}_q^{m \times n}$, the mFHE.iLCP$(\cdot)$ algorithm takes $\widetilde{\mathcal{V}}$ and $\boldsymbol{C}_{\&}$ as input. Then compute and output the lower dimensional ciphertext3) by using a symmetrical encryption.

$$\begin{aligned}\boldsymbol{X} &= \text{mFHE.iLCP}(\boldsymbol{C}_{\&}, \widetilde{\mathcal{V}}) = \boldsymbol{A} + \boldsymbol{C}_{\&} \cdot \widetilde{\boldsymbol{G}}^{-1}(\widetilde{\mathcal{V}}) \\ &= \boldsymbol{A} + \widetilde{\boldsymbol{R}} \cdot \widetilde{\mathcal{V}} \pmod q \in \mathbb{Z}_q^{m \times n}.\end{aligned}$$

mFHE.Decode(sk,$\boldsymbol{X}$):
Compute and output

$$\begin{aligned}\boldsymbol{S}^{\mathrm{T}} \cdot (\boldsymbol{A} + \widetilde{\boldsymbol{R}} \cdot \widetilde{\mathcal{V}})^{\mathrm{T}} &= \boldsymbol{S}^{\mathrm{T}} \widetilde{\mathcal{V}}^{\mathrm{T}} \cdot \widetilde{\boldsymbol{R}} + \boldsymbol{A}\boldsymbol{S} \\ &= \mathcal{V}^{\mathrm{T}} \cdot \widetilde{\boldsymbol{R}} + [\boldsymbol{e}_1, \ldots, \boldsymbol{e}_t] \pmod q \in \mathbb{Z}_q^{t \times m}.\end{aligned}$$

The security and correctness of the iLCP are proved by the following lemmas.

Lemma 1 (Correctness)**.** If we set the noise of $\boldsymbol{X}$ under secret key $\boldsymbol{S}$ by $\mathcal{T} := [\boldsymbol{e}_1, \ldots, \boldsymbol{e}_t]$, then the bound of $\mathcal{T}$ is $\|[\boldsymbol{e}_1, \ldots, \boldsymbol{e}_t]\| \leqslant t \cdot \|\boldsymbol{e}\| < t \cdot E$.

Lemma 2 (Security)**.** If the LWE assumption is hard, then the iLCP is **IND-CPA**-secure.

Take note that the detailed analysis of the correctness and security is provided in Appendix D.

Construction of mMFHE *via* iLCP. Below we describe our mMFHE construction by the iLCP4).

params $\leftarrow$ mMFHE.Setup(1^{λ}, 1^{L}): Identical to the algorithm of mFHE.Setup(1^{λ}, 1^{L}).

$(\mathrm{pk}, \mathrm{sk}) \leftarrow$ mMFHE.KeyGen(params): Identical to the algorithm of mFHE.KeyGen(params).

$(\boldsymbol{C}_{\#}) \leftarrow$ mMFHE.Enc(pk,$\widetilde{\boldsymbol{U}}$):

(1) To encrypt t bits $u_i \in \{0,1\}$ for $i \in [t]$, we first embed these t bits into message matrix $\boldsymbol{U} = \mathrm{diag}(\mu_1, \ldots, \mu_t) \in \mathbb{Z}_q^{t \times t}$, and then create the plaintext matrix as follows:

$$\boldsymbol{M} = \left(\begin{array}{c:c} \boldsymbol{U}_{t \times t} & \boldsymbol{0}_{t \times (n-t)} \\ \hdashline \boldsymbol{0}_{(n-t) \times t} & \boldsymbol{E}_{(n-t) \times (n-t)} \end{array}\right) \in \{0,1\}^{n \times n},$$

where matrix $\boldsymbol{E} = \mathrm{diag}(1, \ldots, 1) \in \{0,1\}^{(n-t) \times (n-t)}$ is the identity matrix. Concurrently, $\widetilde{\boldsymbol{U}}$ and $\boldsymbol{E}$ are also the two partitioned matrices of plaintext matrix $\boldsymbol{M}$.

(2) Sample uniform matrix $\widetilde{\boldsymbol{R}} \leftarrow \{0,1\}^{m \times m}$, then compute and write $\boldsymbol{C} = \boldsymbol{M} \cdot \widetilde{\boldsymbol{G}} + \boldsymbol{A}^{\mathrm{T}} \cdot \widetilde{\boldsymbol{R}} \pmod q \in \mathbb{Z}_q^{n \times m}$, where $\widetilde{\boldsymbol{G}} \in \mathbb{Z}_q^{n \times m}$.

$\widehat{\boldsymbol{C}} \leftarrow$ mMFHE.Expand$((\mathrm{pk}^{(1)}, \ldots, \mathrm{pk}^{(N)}), i, \boldsymbol{C}_{\#})$:

(1) For $j \in \{\mathrm{pk}^{(1)}, \ldots, \mathrm{pk}^{(N)}\} \setminus \{i\}$, compute $\widetilde{\mathcal{V}} \leftarrow$ mFHE.Extend$(\mathrm{pk}^{(i)}, \mathrm{pk}^{(j)})$ and obtain $\boldsymbol{X}^{(j)} \leftarrow$ mFHE.iLCP$(\boldsymbol{C}_{\&}, \widetilde{\mathcal{V}})$.

(2) Then define matrix $\widehat{\boldsymbol{C}} \in \mathbb{Z}_q^{nN \times mN}$ as a concatenation of N^2 sub-matrix $\boldsymbol{C}_{a,b} \in \mathbb{Z}_q^{n \times m}$ for any $a, b \in [N]$ that is defined as

---

1) The construction of the mFHE scheme can be found in Appendix C.

2) We note that the concept of the "Improved GSW Masking Scheme" in [9] is not required in our scheme.

3) The iLCP algorithm can reduce the dimension of a ciphertext via the function $\widetilde{\boldsymbol{G}}^{-1}$, (i.e., from $\mathbb{Z}^{n \times m}$ to $\mathbb{Z}^{m \times n}$).

4) Note that the Extend$(\cdot)$ and UniEnc$(\cdot)$ algorithms in the MFHE scheme of Mukherjee and Wichs [9] are not used in our mMFHE.

$$\boldsymbol{C}_{a,b} := \begin{cases} \boldsymbol{C}_{\#}, & \text{when } a = b; \\ \boldsymbol{X}^{(j)}, & \text{when } a = i \neq j \text{ and } b = j; \\ \boldsymbol{0}^{n\times m}, & \text{otherwise.} \end{cases}$$

(3) Output $\widehat{c} := \widehat{\boldsymbol{C}}$ as the expanded ciphertext.

mMFHE.Eval(params, $\mathcal{C}$, pk, $(\widehat{\boldsymbol{c}}_1, \ldots, \widehat{\boldsymbol{c}}_N)$):

Take N expanded ciphertexts as input. The algorithm achieves the homomorphic evaluation by the two algorithms: mFHE.Add($\cdot$) and mFHE.Mult($\cdot$).

mMFHE.Dec(params, $(\mathrm{sk}^{(1)}, \ldots, \mathrm{sk}^{(N)})$, c):

Take a ciphertext $c = \widehat{\boldsymbol{C}}$ and the sequence of secret keys $(\mathrm{sk}^{(1)}, \ldots, \mathrm{sk}^{(N)})$ as input. Then parse each secret key $\boldsymbol{S}^{(i)} := \mathrm{sk}^{(i)}$ and construct a joint secret key by $\widehat{\boldsymbol{S}} = [(\boldsymbol{S}^{(1)})^{\mathrm{T}}, \ldots, (\boldsymbol{S}^{(N)})^{\mathrm{T}}] \in \mathbb{Z}_q^{t\times nN}$.

Below we will illustrate precisely how to achieve threshold decryption for the mMFHE.

mMFHE.PartDec($\widehat{c}$, $(\mathrm{pk}^{(1)}, \ldots, \mathrm{pk}^{(N)})$, i, $\mathrm{sk}^{(i)}$): On input of an expanded ciphertext $\widehat{\boldsymbol{c}} = \widehat{\boldsymbol{C}}$ under a sequence of keys $(\mathrm{pk}^{(1)}, \ldots, \mathrm{pk}^{(N)})$ and i-th secret key $\mathrm{sk}^{(i)} = (\boldsymbol{S}^{(i)})^{\mathrm{T}} \in {\mathbb{Z}_q}^{t\times n}$.

Then we perform the following steps:

(1) Parse $\widehat{\boldsymbol{C}}$ into $\widehat{\boldsymbol{C}} = (\widehat{\boldsymbol{C}}^{(1)}; \ldots; \widehat{\boldsymbol{C}}^{(N)})_{nN\times mN}$, where each sub-matrix $\widehat{\boldsymbol{C}}^{(i)}$ is over $\mathbb{Z}_q^{n\times mN}$.

(2) First denote the following matrix:

$$\boldsymbol{W} = \left(\begin{array}{ccc|c} \lceil q/2 \rceil & \ldots & 0 & \boldsymbol{0}^{1\times(n-t)} \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & \lceil q/2 \rceil & \boldsymbol{0}^{1\times(n-t)} \end{array} \right) \in \mathbb{Z}_q^{t\times n}.$$

Then obtain the following matrix: $\widehat{\boldsymbol{W}}^{\mathrm{T}} = [\underbrace{\boldsymbol{W}^{\mathrm{T}}, \ldots, \boldsymbol{W}^{\mathrm{T}}}_{N}] \in \mathbb{Z}_q^{nN\times t}$.

(3) For simplicity, we re-denote $\widehat{\boldsymbol{G}}_N \in \mathbb{Z}_q^{nN\times mN}$ and $\widehat{\boldsymbol{G}}_N^{-1} \in \mathbb{Z}_q^{mN\times nN}$, and we write each element of $\boldsymbol{r}^{(i)}$ as one column of $\mathcal{R}^{(i)} = (\boldsymbol{S}^{(i)})^{\mathrm{T}} \cdot \widehat{\boldsymbol{C}}^{(i)} \cdot \widehat{\boldsymbol{G}}^{-1}(\widehat{\mathbf{W}}^{\mathrm{T}}) \in \mathbb{Z}_q^{t\times t}$. Most notably, for convenience, here $\mathcal{R}^{(i)}$ can be regarded as the following matrix:

$$\mathcal{R}^{(i)} := [\boldsymbol{r}_1^{(i)}, \ldots, \boldsymbol{r}_t^{(i)}] = \begin{pmatrix} \gamma_{1,1}^{(i)} & \cdots & \gamma_{1,t}^{(i)} \\ \vdots & \ddots & \vdots \\ \gamma_{t,1}^{(i)} & \cdots & \gamma_{t,t}^{(i)} \end{pmatrix},$$

where $\gamma^{(i)} = \langle \boldsymbol{s}_j^{(i)}, \widehat{\boldsymbol{C}}^{(i)} \rangle \cdot \widetilde{\boldsymbol{G}}^{-1}(\widehat{\boldsymbol{W}}_j^{\mathrm{T}})$ for $i \in [N]$ and $j \in [t]$. We must stress that $\boldsymbol{s}_j^{(i)}$ is the j-th column of $\boldsymbol{S}^{(i)}$ and $\widehat{\boldsymbol{W}}_j$ is the j-th column of $\widehat{\boldsymbol{W}}$. Hence, there exists $p^{(i)} = \gamma^{(i)} + {e^{(i)}}^{\mathrm{sm}} \in \mathbb{Z}_q$, where ${e^{(i)}}^{\mathrm{sm}} \leftarrow [-B_{\mathrm{smdg}}^{\mathrm{dec}}, B_{\mathrm{smdg}}^{\mathrm{dec}}]$ is some random “smudging noise”.

(4) Output $\boldsymbol{P}^{(i)} = [\boldsymbol{p}_1^{(i)}, \ldots, \boldsymbol{p}_t^{(i)}] \in \mathbb{Z}_q^{t\times t}$, where $\boldsymbol{p}^{(i)} = \sum_j^t p_j^{(i)} \in \mathbb{Z}_q^{t\times 1}$.

mMFHE.FinDec($\boldsymbol{P}^{(1)}, \ldots, \boldsymbol{P}^{(N)}$):

The algorithm takes $\boldsymbol{P}^{(1)}, \ldots, \boldsymbol{P}^{(N)}$ as input, and computes

$$\boldsymbol{P} := \sum_{i=1}^{N} \boldsymbol{P}^{(i)} = \left[\sum_{i=1}^{N} \boldsymbol{p}_1^{(i)}, \ldots, \sum_{i=1}^{N} \boldsymbol{p}_t^{(i)} \right].$$

Output decryption message $\boldsymbol{U} := \frac{1}{N} \cdot \| \lceil \frac{\boldsymbol{P}}{q/2} \rfloor \|$.

Remark 2. $B_{\mathrm{smdg}}^{\mathrm{dec}}$ is bounded for extra “smudging” noise and $B_{\mathrm{smdg}}^{\mathrm{dec}} = 2^{d\lambda \log \lambda} B_{\chi}$.

Theorem 1. Assume that the MFHE scheme of Mukherjee and Wichs and our constructed scheme iLCP are indistinguishable chosen-plaintext attacks (or IND-CPA) secure, then the mMFHE scheme is IND-CPA-secure.

Remark 3. Assume that there exist two players (e.g., User1 and User2) in the system. If User2 uses his own secret key $\boldsymbol{S}^{(2)}$ to decrypt the ciphertext $\boldsymbol{C}_{\#}^{(1)}$ received from User1, then there exists

$$\langle \boldsymbol{S}^{(2)}, \boldsymbol{C}_{\#}^{(1)} \rangle + \langle \boldsymbol{S}^{(1)}, \boldsymbol{X}^{\mathrm{T}} \rangle = (\boldsymbol{S}^{(2)})^{\mathrm{T}} \cdot \boldsymbol{M}\widetilde{\boldsymbol{G}} + \text{noise}.$$

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant No. 61472097).

Supporting information Appendices A–E. The supporting information is available online at info.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

1 Gentry C. Fully homomorphic encryption using ideal lattices. Acm Symp Theory Comput, 2009, 9: 169–178

2 Brakerski Z, Vaikuntanathan V. Efficient fully homomorphic encryption from (standard) LWE. In: Proceedings of 2011 IEEE 52nd Annual Symposium on Foundations of Computer Science (FOCS), Palm Springs, 2011. 97–106

3 Gentry C, Sahai A, Waters B. Homomorphic encryption from learning with errors: conceptually-simpler, asymptotically-faster, attribute-based. In: Advances in Cryptology — CRYPTO 2013. Berlin: Springer, 2013. 8042: 75–92

4 Hiromasa R, Abe M, Okamoto T. Packing messages and optimizing bootstrapping in GSW-FHE. In: Public-Key Cryptography — PKC 2015. Berlin: Springer, 2015. 9020: 699–715

5 Li Z, Galbraith S, Ma C. Preventing adaptive key recovery attacks on the GSW levelled homomorphic encryption scheme. In: Proceedings of International Conference on Provable Security. Cham: Springer, 2016. 10005: 373–383

6 Li Z, Ma C, Morais E, et al. Multi-bit leveled homomorphic encryption via dual.lwe-based. In: Proceedings of International Conference on Information Security and Cryptology. Cham: Springer, 2016. 10143: 221–242

7 López-Alt A, Tromer E, Vaikuntanathan V. On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption. In: Proceedings of the 44th ACM Symposium on Theory of Computing, New York, 2012. 1219–1234

8 Clear M, McGoldrick C. Multi-identity and multi-key leveled FHE from learning with errors. In: Advances in Cryptology — CRYPTO 2015. Berlin: Springer, 2015. 9216: 630–656

9 Mukherjee P, Wichs D. Two round multiparty computation via multi-key FHE. In: Proceedings of Annual International Conference on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 2016. 9666: 735–763