Future Generation Computer Systems 36 (2014) 180–186

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Efficient fully homomorphic encryption from RLWE with an extension to a threshold encryption scheme

Xiaojun Zhang [a,*], Chunxiang Xu [a,**], Chunhua Jin [a], Run Xie [a,b], Jining Zhao [a]

[a] School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

[b] School of Mathematical, Yibin University, Yibin 644000, China

HIGHLIGHTS

- We present an efficient fully homomorphic encryption (FHE) from RLWE.
- We get the FHE scheme with re-linearization and modulus reduction techniques.
- We extend the FHE scheme to the threshold fully homomorphic encryption scheme.

ARTICLE INFO

Article history:
Received 8 April 2013
Received in revised form
21 October 2013
Accepted 23 October 2013
Available online 11 November 2013

Keywords:
Fully homomorphic encryption
Threshold encryption
RLWE
Key-homomorphic

ABSTRACT

In this paper, we present an effective fully homomorphic encryption (FHE) from ring learning with errors (RLWE) assumption without using Gentry's standard squashing and bootstrapping techniques. Our FHE scheme is to modify the recent FHE scheme of Brakerski. We use the re-linearization technique to reduce the length of ciphertext considerably, and use the modulus reduction technique to manage the noise level and decrease the decryption complexity without introducing additional assumptions. Furthermore, with the key-homomorphic property, we extend our FHE scheme to a threshold fully homomorphic encryption (TFHE), which allows parties to cooperatively decrypt a ciphertext without learning anything but the plaintext. The TFHE scheme can be protected from related-key attacks, as long as we add extra smudging noise during sensitive operations.

1. Introduction

Fully homomorphic encryption (FHE) is one of the holy grails of modern cryptography. A FHE scheme allows a worker to perform arbitrary computations on encrypted data without decrypting it. The problem was first proposed by Rivest, Adleman and Dertouzos [1] back in 1978. However, until recently, a breakthrough work by Gentry [2,3] constructed the first FHE scheme based on the hardness of problem on ideal lattice, which is a sophisticated algebraic structure with useful properties. Naturally, subsequent FHE schemes [4–8] followed the same blueprint from Gentry's original construction.

Generally, the first step in Gentry's blueprint is to construct a somewhat homomorphic encryption scheme, which is capable of evaluating limited degree polynomials homomorphically. In the following step, Gentry transforms the somewhat homomorphic encryption scheme into the FHE scheme with bootstrapping and squashing techniques. The remarkable bootstrapping technique states that it can run the decryption circuit on a ciphertext homomorphically, using an encrypted secret key, resulting in reduced noise. However, the bootstrapping technique forces the public key of the scheme to grow linearly with the maximal depth of evaluation circuits. This is a major drawback regarding the usability and the efficiency of the scheme. The squashing technique can transform a somewhat homomorphic encryption scheme into one with the same homomorphic capacity even a decryption circuit that is simple enough to allow bootstrapping, yet the squashing step adds another assumption, namely the hardness of the sparse subset sum problem. Consequently, considering the performance and usability, we need to look for some appropriate techniques to resolve the problem.

Recently, Brakerski and Vaikuntanathan [9] found a very different way to construct FHE scheme based on LWE. In the scheme, they introduced a new dimension-modulus reduction technique, which shortens the ciphertext and reduces the decryption complexity, without using the squashing step. From then on, another FHE scheme [10] based on LWE assumption with the similar technique appeared. Furthermore, Brakerski and Vaikuntanathan presented another FHE scheme [11] that followed the standard

* Corresponding author. Tel.: +86 18011394462.
** Corresponding author.
E-mail addresses: zhangxjdzkd2012@163.com, xiaojunzhang_019@126.com (X. Zhang), chxxu@uestc.edu.cn (C. Xu).

0167-739X/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.future.2013.10.024

squashing and bootstrapping techniques. And the scheme was based on ring learning with errors (RLWE) assumption which was recently introduced by Lyubashevsky [12], whose security is reduced to the worst-case hardness of problems on ideal lattices, resulting in an extremely simple scheme. Subsequently, some similar schemes based on RLWE have been proposed, such as [13,14].

Meanwhile, we observe that recent FHE schemes based on LWE or RLWE own a special property, namely (additive) key homomorphism. A key-homomorphic encryption allows us to deterministically combine public keys into a combined public key, and simultaneously combine corresponding secret keys into a corresponding combined secret key. This property allows combining encryptions of messages under different keys to produce an encryption (of the sum of the messages) under the sum of the keys. Since lattice based schemes own key-homomorphic property, it plays a great role in constructing some useful cryptographic primitives, especially in the construction of threshold fully homomorphic encryption (TFHE), which was pointed out in Gentry [2]. The TFHE scheme allows parties to cooperatively generate a common public key whose secret key is shared among them. Moreover, the parties can cooperatively decrypt a ciphertext without learning anything but the plaintext.

1.1. Our results and techniques

In this paper, we present an efficient FHE scheme and extend it to a TFHE scheme. First of all, we modify the recent FHE scheme of Brakerski [11], which was based on RLWE, with its security reduced to worst-case problems on ideal lattices. The primitive scheme followed Gentry's standard bootstrapping and squashing steps, while in our modified scheme, the squashing step can be avoided. Moreover, the ciphertext produced by the primitive scheme contained two ring elements. However, the multiplications increased the number of ring elements in the ciphertext considerably. In general, given two ciphertexts $c = (c_0, c_1, \ldots, c_\delta)$ and $c' = (c'_0, c'_1, \ldots, c'_\gamma)$, the output of homomorphic multiplication contains $\delta+\gamma+1$ ring elements. While in our modified scheme, we employ the re-linearization technique from [9] to reduce the size of the resulting ciphertext after each multiplication, thus the ciphertext still contains two elements, and dramatically decreases the communication. The crucial question in the process of constructing FHE scheme is noise level, which grows exponentially with the number of multiplications, we have to manage the noise level so that it can be decrypted correctly. Confronted with the difficulty, Gentry leverages the bootstrapping procedure, however, which performs with great complexity to reduce the noise level. The key technique we use for noise level management is modulus reduction first introduced in the work of [9], developed in [10]. With the modulus reduction technique, our scheme enjoys the same amount of homomorphism but has a much smaller decrypt circuit. Thanks to the two techniques, we get our modified FHE scheme.

The basic idea of combining homomorphic encryption with threshold decryption was first noticed by Cramer [15]. Subsequently, some similar research work appeared, such as [16,17]. Our idea of constructing TFHE scheme benefits greatly from the [18]. In particular, we exploit the key-homomorphic property to construct the threshold scheme and we use extra smudging noise to keep security of joint keys, so that it can withstand against related-key attacks. The construction of our TFHE scheme based on RLWE instead of LWE is a new interesting attempt. We also observe that in the work of [18], the researcher made a great effort to generate the combined evaluation key. While in our scheme, for simplicity, we resort to a Functionality F_{KeyGen} to solve the thorny problem, and it executes computing honestly and prudently. Meanwhile, we find that our TFHE scheme is superior to the instantiation of [19], whose public key contains much more ring elements, yet our scheme only needs two. Furthermore, we employ extra smudging noise to keep security, while the scheme in [19] makes use of an algorithm ReRand to output a rerandomization ciphertext, which has greater complexity than ours. Moreover, we claim that our TFHE scheme can be applied to construct multiparty computation protocols, which maybe play an important role in cloud computing.

2. Preliminaries

In the remainder of this paper, we use the following notation. We use κ to denote the security parameter and negl(κ) to denote a negligible function. For a real number κ, we denote by $\lceil \kappa \rceil$, $\lfloor \kappa \rfloor$, $\lfloor \kappa \rceil$ the rounding of a up, down, or to the nearest integer respectively. For an integer n, we use the notational $[n]$ to denote the set $[n] = \{1, \ldots, n\}$. For some distribution χ, writing $x \leftarrow \chi$ means that x is distributed according to χ.

2.1. Fully homomorphic encryption

Now we give two definitions about fully homomorphic encryption.

Definition 1 (*$\mathcal{C}$-homomorphism*)**.** Let $\mathcal{C} = \{\mathcal{C}_\kappa\}_{\kappa \in N}$ be a class of function (together with their respective representations). A scheme HE is $\mathcal{C}$-homomorphic if for any sequence of function $f_\kappa \in \mathcal{C}_\kappa$ and respective inputs $m_1, \ldots, m_\ell \in \{0, 1\}$, it holds that $\Pr[\text{HE.Dec}_{sk}(\text{HE.Eval}_{evk}(f, c_1, \ldots, c_\ell)) \neq f(m_1, \ldots, m_\ell)] = \text{negl}(\kappa)$, where $(pk, evk, sk) \leftarrow \text{HE.Keygen}(1^\kappa)$ and $c_i \leftarrow \text{HE.Enc}_{pk}(m_i)$.

Definition 2 (*Leveled Fully Homomorphic Encryption*)**.** A leveled fully homomorphic encryption scheme is a homomorphic scheme where the HE.Keygen gets an additional input 1^L (now $(pk, evk, sk) \leftarrow \text{HE.Keygen}(1^\kappa, 1^L)$) and the resulting scheme is homomorphic for all depth-L binary arithmetic circuits. The bound $s(\kappa)$ on the ciphertext length must remain independent of L.

From then on, when we say fully homomorphic, we refer to leveled fully homomorphic encryption.

2.2. The ring LWE assumption

The ring of polynomials over the integers is denoted $Z[x]$, the ring of polynomials modulo the ideal $\langle f(x)\rangle$ is denoted $R = Z[X]/\langle f(x)\rangle$. The ring of polynomials with coefficients in Z_q is denoted $Z_q[X]$, quotient ring $R_q = Z_q[X]/\langle f(x)\rangle$ is defined similarly to R. We write elements of R in lowercase (e.g. x) and vectors in bold (e.g. $\mathbf{v}$), the notation $\mathbf{v}_i$ refers to the ith coefficient of $\mathbf{v}$. For $a \in R$, where R is a polynomial ring, $\|a\|$ refers to the Euclidean norm of $a's$ coefficient vector. For $a(x) = a_0 + a_1x + \cdots + a_{n-1}x^{n-1} \in R$, we let $\|a\|_\infty = \max |a|_i$ denote its l_∞ norm and $\|a\|_1 = \sum_{i=0}^{n-1} a_i$ denote its l_1 norm. For $a \in R$, we use the notation $[a]_q$ to refer to a mod q, with coefficients reduced into the range $(-q/2, q/2]$.

Definition 3 (*The RLWE Assumption-Hermite Normal From [11,12]*)**.** For all $\kappa \in N$, let $f(x) = f_\kappa(x) \in Z[x]$ be a polynomial of degree $n = n(\kappa)$, let $q = q(\kappa) \in Z$ be a prime integer, let the ring $R = Z[x]/\langle f(x)\rangle$ and R_q, and let χ denote a distribution over the ring R. The decisional ring LWE assumption states that for any polynomial samples of the form $(a_i, b_i = a_is + e_i)$, and $b_i{}'s$ are computationally indistinguishable from uniform in R_q, where s and e_i are sampled from the noise distribution χ, a_i is uniform in R_q.

We define a B-bounded distribution to be a distribution over R where the l_∞ norm of a sample is bounded.

Definition 4 (*B-Bounded Polynomial*)**.** A polynomial $e \in R$ is called B-bounded if $\|e\|_\infty \leq B$.

Definition 5 (*B-Bounded Distribution*). A distribution ensemble $\{\chi_n\}_{n\in N}$, supported over R, is called B-bounded if for all e in $\{\chi_n\}$, we have $\|e\|_\infty \le B$. In other words, a B-bounded distribution over R outputs a B-bounded polynomial.

Now we let $f(x) = x^n + 1$ be the nth cyclotomic polynomial, where n is a power of two.

Lemma 1 (*See [3]*). *We let $n \in N, f(x) = x^n + 1$ and let $R = Z[x]/\langle f(x)\rangle$. For any $s, t \in R$, $\|st(\bmod f(x))\| \le \sqrt{n}\cdot\|s\|\cdot\|t\|$; $\|st(\bmod f(x))\|_\infty \le n\cdot\|s\|_\infty\cdot\|t\|_\infty$.*

Lemma 2 (*See [18]*). *Let $B_1 = B_1(\kappa)$, and $B_2 = B_2(\kappa)$ be positive integers, set a fixed ring element e_1 to be bounded by B_1, and e_2 bounded by B_2 with its coefficients being chosen uniformly random from $[-B_2, B_2]$. Then the distribution of e_2 is statistically indistinguishable from that of $e_2 + e_1$ as long as $B_1/B_2 = \mathrm{negl}(\kappa)$.*

2.3. The worst-case to average-case connection

We state a worst-case to average-case reduction from the shortest vector problem on ideal lattices to the RLWE problem for our setting of parameters.

Theorem 1 (*See [12]*). *Let $\Phi_m(x) = x^n + 1$ be the mth cyclotomic polynomial of degree $n = \varphi(m) = m/2$, where $m = 2^{\lfloor\log\kappa\rceil}$. Let $r \ge \omega(\sqrt{\log n})$ be a real number, and let $q \equiv 1(\bmod m)$ be a prime integer. Let $R = Z[x]/\langle x^n+1\rangle$. Then there is a randomized reduction from $2^{\omega(\sqrt{\log n})}\cdot(q/r)$-approximate R-SVP to Ring LWE, where $\chi = D_{Z^n,r}$ is the discrete Gaussian distribution. The reduction runs in time* poly (n, q).

3. A somewhat homomorphic encryption scheme

In this section, we begin to describe a somewhat homomorphic public-key encryption scheme based on RLWE which is modified from the private-key encryption scheme in [11]. In order to guarantee correctness and security, we set parameters below which depend on the security parameter κ. Now we define the ring $R = Z[X]/\langle f(x)\rangle$ and $R_q = Z_q[x]/\langle f(x)\rangle$, with the cyclotomic polynomial $f(x) = x^n + 1$. We set the error distribution χ to be the truncated discrete Gaussian $D_{Z^n,r}$ for standard deviation r. A sample from this distribution is a B-bounded polynomial.

SH.Keygen(1^κ): We sample a ring element $s \leftarrow \chi$, a uniformly random ring element $a_0 \leftarrow R_q$ and an error $e_0 \leftarrow \chi$. Set the secret key $sk = s$, the public key $pk = (a_0, b_0 = -(a_0 s + 2e_0))$.

SH.Enc(pk, m): To encrypt a message $m \in \{0, 1\}$, we sample $u \leftarrow \chi$ and $e_1, e_2 \leftarrow \chi$, compute $v = b_0 u + 2e_1 + m$ and $w = -(a_0 u + 2e_2)$, output the ciphertext $\mathbf{c} = (v, w)$.

SH.Dec$(sk, \mathbf{c})$: With the secret key $sk = s$, we output the plaintext $m = [v - ws]_q(\bmod 2)$.

Correctness and security.

Firstly, we compute $v - ws = b_0 u + 2e_1 + m + (a_0 u + 2e_2)s = -(a_0 s + 2e_0)u + 2e_1 + m + (a_0 u + 2e_2)s = 2(e_2 s + e_1 - e_0 u) + m$. By Lemma 1, the coefficients of $v - ws$ are bounded by $2(nB^2 + B + nB^2) + 1 < q/2$. In other words, as long as we set $q > 16nB^2$, a fresh ciphertext $\mathbf{c} = (v, w)$ is guaranteed to decrypt correctly, namely $[v - ws]_q(\bmod 2) = m$.

We show the security of the somewhat homomorphic encryption scheme below.

Theorem 2 (*[11]*). *let $n, q, f(x)$ be as in the scheme, let $q = 2^{n^\epsilon}$ for some $0 < \epsilon < 1$. Then the scheme allows evaluation of degree-$O(n^\epsilon/\log n)$ polynomials with at most $2^{O(n^\epsilon/\log n)}$ terms, and is secure under the worst-case hardness of approximating shortest vectors on ideal lattices to within a factor of $O(2^{n^\epsilon})$.*

Since $v = b_0 u + 2e_1 + m$ and $w = -(a_0 u + 2e_2)$ are both RLWE samples, according to the Theorem 2, we get the scheme's semantic security.

3.1. An optimization to the somewhat homomorphic encryption scheme

As described above, although a ciphertext produced by SH.Enc contains two ring elements, the homomorphic multiplication will increase the number of ring elements greatly in the ciphertext.

For convenience, we execute homomorphic multiplication on two ciphertexts. Define a symbolic linear function $\phi_{\mathbf{c}}(x) = v - wx(\bmod 2)$, which means that decrypting $\mathbf{c}$ corresponds to simply computing $\phi_{\mathbf{c}}(s)$. Suppose c and c' are ciphertexts of m and m' respectively, then $mm' = \phi_{\mathbf{c}}(s)\phi_{\mathbf{c}'}(s) = (v - ws)(v' - w's)(\bmod 2) = vv' + (-wv' - vw')s + ww's^2(\bmod 2) = \lambda_0 + \lambda_1 s + \lambda_2 s^2(\bmod 2)$. Let $\mathbf{c}_{\mathrm{mult}} = (\lambda_0, \lambda_1, \lambda_2)$, note that the size of the ciphertext grows linearly with the number of multiplications. This means if we set L as the maximal degree of evaluation we can compute, then we require the secret key $\mathbf{s} = (1, s, s^2, \ldots, s^L)$ for decryption. Now, we employ the re-linearization technique to realize our optimization below. In order to homomorphically evaluate a polynomial of degree L while keeping the ciphertext size constant, we sample $L + 1$ different polynomials $s_0, s_1, \ldots, s_L$ from the error distribution χ, one for each level, where s_0 is used to create the public key $pk = (a_0, b_0 = -(a_0 s_0 + 2e_0))$, s_L is used for decryption. Our main challenge is that the public key must also contain additional information in the form of an evaluation key, which has a more complex structure. The evaluation key is computed as follows. First of all, for all $\ell \in [L]$, $\tau \in \{0, \ldots, \lfloor\log q\rfloor\}$, we sample $(a_{\ell,\tau}, b_{\ell,\tau} = -(a_{\ell,\tau} s_\ell + 2e_{\ell,\tau})) \in R_q^2$, $(a'_{\ell,\tau}, b'_{\ell,\tau} = -(a'_{\ell,\tau} s_\ell + 2e'_{\ell,\tau})) \in R_q^2$, where $a_{\ell,\tau}, a'_{\ell,\tau} \leftarrow R_q$, and $e_{\ell,\tau}, e'_{\ell,\tau} \leftarrow \chi$. Compute:

$$\xi_{0,\ell,\tau} = a_{\ell,\tau}, \qquad \xi_{1,\ell,\tau} = b_{\ell,\tau} - 2^\tau s_{\ell-1} \in R_q;$$

$$\zeta_{0,\ell,\tau} = a'_{\ell,\tau}, \qquad \zeta_{1,\ell,\tau} = b'_{\ell,\tau} - 2^\tau s_{\ell-1}^2 \in R_q$$

we let $\xi_{0,\ell} = (\xi_{0,\ell,0}, \ldots, \xi_{0,\ell,\lfloor\log q\rfloor})$ and $\xi_{1,\ell} = (\xi_{1,\ell,0}, \ldots, \xi_{1,\ell,\lfloor\log q\rfloor})$.

And we let: $\zeta_{0,\ell} = (\zeta_{0,\ell,0}, \ldots, \zeta_{0,\ell,\lfloor\log q\rfloor})$ and $\zeta_{1,\ell} = (\zeta_{1,\ell,0}, \ldots, \zeta_{1,\ell,\lfloor\log q\rfloor})$.

Finally, we get $evk = \{\xi_{0,\ell}, \xi_{1,\ell}, \zeta_{0,\ell}, \zeta_{1,\ell}\}_{\ell\in[L]}$.

Given two ciphertexts $c = (v, w)$ and $c' = (v', w')$ under the same secret key $s_{\ell-1}$. We denote $\mathbf{c}_{\mathrm{mult}}$ as an encryption of the product of the underlying messages, that is $\mathbf{c}_{\mathrm{mult}} = (\lambda_0, \lambda_1, \lambda_2)$. In particular, $\lambda_0 + \lambda_1 s_{\ell-1} + \lambda_2 s_{\ell-1}^2(\bmod 2) = mm'$. We also denote the binary representation of λ_1 and λ_2 by $\mu = (\mu_0, \ldots, \mu_{\lfloor\log q\rfloor}) \in R_2^{\lceil\log q\rceil}$ and $\nu = (\nu_0, \ldots, \nu_{\lfloor\log q\rfloor}) \in R_2^{\lceil\log q\rceil}$, for $\mu_i, \nu_i \in R_2$ respectively. We get $\lambda_1 = \sum_{\tau=0}^{\lfloor\log q\rfloor} 2^\tau \mu_\tau$ and $\lambda_2 = \sum_{\tau=0}^{\lfloor\log q\rfloor} 2^\tau \nu_\tau$.

With the evaluation key evk, we output $\mathbf{c}_{\mathrm{mult}} = (\lambda_0 - \langle\mu, \xi_{1,\ell}\rangle - \langle\nu, \zeta_{1,\ell}\rangle, \langle\mu, \xi_{0,\ell}\rangle + \langle\nu, \zeta_{0,\ell}\rangle)$.

Notice that $\langle\mu, \xi_{1,\ell}\rangle = \sum_{\tau=0}^{\lfloor\log q\rfloor} \mu_\tau \xi_{1,\ell,\tau} = -\langle\mu, \xi_{0,\ell}\rangle s_\ell - \lambda_1 s_{\ell-1} - 2e$, where $e = \sum_{\tau=0}^{\lfloor\log q\rfloor} \mu_\tau e_{\ell,\tau}$; and $\langle\nu, \zeta_{1,\ell}\rangle = \sum_{\tau=0}^{\lfloor\log q\rfloor} \nu_\tau \zeta_{1,\ell,\tau} = -\langle\nu, \zeta_{0,\ell}\rangle s_\ell - \lambda_2 s_{\ell-1}^2 - 2e'$, where $e' = \sum_{\tau=0}^{\lfloor\log q\rfloor} \nu_\tau e'_{\ell,\tau}$.

Thus we have $(\lambda_0 - \langle\mu, \xi_{1,\ell}\rangle - \langle\nu, \zeta_{1,\ell}\rangle) - (\langle\mu, \xi_{0,\ell}\rangle + \langle\nu, \zeta_{0,\ell}\rangle)s_\ell(\bmod 2) = \lambda_0 + \lambda_1 s_{\ell-1} + \lambda_2 s_{\ell-1}^2 + 2(e + e')(\bmod 2) = mm'$, as long as e and e' are small enough errors. Therefore, $\mathbf{c}_{\mathrm{mult}}$ is a valid encryption of mm' under secret key s_ℓ, and the resulting ciphertext still contains two ring elements.

4. Fully homomorphic encryption scheme

As described in Section 3.1, with the re-linearization technique, we keep the ciphertext size constant when performing evaluation. However, we left out the crucial question of noise level, whose magnitude grows exponentially with the number of multiplications. To tackle this, we employ a modulus reduction technique, which uses progressively smaller moduli q_ℓ for each level ℓ and simply rescales the ciphertext to the smaller modulus to reduce its noise level. In particular, for a secret key s, we let $\mathbf{s} = (1, -s)$ and rewrite the decryption function $m = [v - ws]_q(\bmod 2)$ as $m = [\langle\mathbf{c}, \mathbf{s}\rangle]_q(\bmod 2)$. Modulus reduction allows us to transform a ciphertext $\mathbf{c} \in R_q^2$ into a different ciphertext $\mathbf{c}' \in R_q^2$ with simply scaling by p/q and rounding appropriately while keeping the correctness: $[\langle \mathbf{c}', \mathbf{s} \rangle]_p \equiv [\langle \mathbf{c}, \mathbf{s} \rangle]_q (\text{mod } 2)$. We refer the detail to the following theorem.

Theorem 3 ([10]). *Let p and q be two odd moduli, and let $\mathbf{c} \in R_q^2$. Set $\mathbf{c}' \in R_q^2$ such that it is closest to $(p/q)\mathbf{c}$ and $\mathbf{c}' \equiv \mathbf{c} (\text{mod } 2)$. Then, for any $\mathbf{s}$ with $\|[\langle \mathbf{c}, s \rangle]_q\|_\infty < q/2 - (q/p)\|\mathbf{s}\|_1$, we have $[\langle \mathbf{c}', \mathbf{s} \rangle]_p \equiv [\langle \mathbf{c}, s \rangle]_q (\text{mod } 2)$, and $\|[\langle \mathbf{c}', \mathbf{s} \rangle]_p\|_\infty < (p/q)\|[\langle \mathbf{c}, s \rangle]_q\|_\infty + \|s\|_1$.*

Most attractively, if s is short and p is sufficiently smaller than q, the magnitude of noise in the ciphertext actually deceases.

According to Theorem 3, we assume L is the depth of the circuit to be evaluated, thus we can construct a ladder of decreasing moduli $q_0, \ldots, q_L$ and perform modulus reduction after each operation so that at level ℓ all ciphertexts reside in R_{q_L}. Now we present the modified FHE scheme as follows.

Parameters: We set params $= \{\text{params}_\ell = (1^\kappa, q_\ell, n, \chi)\}_{0 \le \ell \le L}$, where χ is a discrete B-bounded error distribution. Note that only the modulus q_ℓ differs for each level ℓ.

FHE.Keygen(params): The key generation algorithm creates (pk, sk, evk) as follows. For every $\ell \in \{0, 1, \ldots, L\}$, $\tau \in \{0, \ldots, \lfloor \log q_{\ell-1} \rfloor\}$. We sample $L+1$ ring elements $s_0, s_1, \ldots, s_L \leftarrow \chi$.

- *The public key*: $pk = (a_0, b_0 = -(a_0 s_0 + 2e_0))$, where a_0 uniformly generates from R_q, an error ring element $e_0 \leftarrow \chi$.
- *The secret key*: The secret key is simply $sk = s_L$.
- *The evaluation key*: The evaluation key is computed similarly in Section 3.1, we get the $evk = \{\xi_{0,\ell}, \xi_{1,\ell}, \zeta_{0,\ell}, \zeta_{1,\ell}\}_{\ell \in [L]}$. With the only difference being that from ladder level $\ell - 1$ to level ℓ, each coefficient of evk is reduced modulo $R_{q_{\ell-1}}$ rather than R_q.

FHE.Enc$_{pk}$(m): Recall that $pk = (a_0, b_0)$, to encrypt a message $m \in \{0, 1\}$, we sample $u \leftarrow \chi$ and $e_1, e_2 \leftarrow \chi$, set $v = b_0 u + 2e_1 + m$, $w = -(a_0 u + 2e_2)$, output ciphertext $\mathbf{c} = ((v, w), 0)$. In addition to a level tag which is used during homomorphic evaluation and indicates the multiplicative depth where the ciphertext has been generating, for freshly encrypted ciphertext, we set the level tag to be zero.

FHE.Dec$_{sk}$(c): On input $\mathbf{c} = ((v, w), L)$, the decryption algorithm outputs plaintext $m = [v - ws_L]_{q_L} (\text{mod } 2)$.

FHE.Eval$_{evk}$(f, c$_1$, c$_2$, ..., c$_t$). Now we assume the circuit contains addition gates and multiplication gates. Given two ciphertexts $\mathbf{c} = ((v, w), \ell - 1)$, $\mathbf{c}' = ((v', w'), \ell - 1)$ under the same $s_{\ell-1}$ and modulus $q_{\ell-1}$.

- *Addition gates*. $c_{\text{add}} = ((v, w), \ell - 1) + ((v', w'), \ell - 1) = ((v + v', w + w'), \ell - 1)$.
- *Multiplication gates*. As computed before, since $\mathbf{c}_{\text{mult}} = (\lambda_0, \lambda_1, \lambda_2)$ is the encryption of mm' under the secret key $s_{\ell-1}$, and with the evaluation key $evk = \{\xi_{0,\ell}, \xi_{1,\ell}, \zeta_{0,\ell}, \zeta_{1,\ell}\}_{\ell \in [L]}$, we get $\mathbf{c}_{\text{mult}} = (\lambda_0 - \langle \mu, \xi_{1,\ell} \rangle - \langle \nu, \zeta_{1,\ell} \rangle, \langle \mu, \xi_{0,\ell} \rangle + \langle \nu, \zeta_{0,\ell} \rangle)$ under the secret key s_ℓ, and the level tag increases to ℓ.

Finally, we exploit the modulus reduction technique to convert $\mathbf{c}_{\text{mult}}$ over the modulus $q_{\ell-1}$ to ones over the smaller modulus q_ℓ. By the Theorem 3, we let $\mathbf{c}'_{\text{mult}}$ be a polynomial ring vector closest to $(q_\ell / q_{\ell-1}) \cdot \mathbf{c}_{\text{mult}}$ such that $\mathbf{c}_{\text{mult}} = \mathbf{c}'_{\text{mult}} (\text{mod } 2)$. At last, we get $\mathbf{c}'_{\text{mult}}$, which is over the modulus q_ℓ, and the magnitude of the noise at each level is almost unchanged. Once the evaluation is completed, it is possible to decrypt the resulting ciphertext without decryption errors.

4.1. Correctness and managing the noise level

Firstly, we define the magnitude of noise in a ciphertext $\mathbf{c} = (v, w)$ (with respect to a key s and a modulus q) as $\text{noise}_q(\mathbf{c}, s) = [v - ws]_q$. In order to make sure the scheme can be decrypted correctly, we set $q_L \gg B$. Now we are setting an appropriate upper bound for the magnitude of noise below.

Theorem 4. *Let ρ be some value such that $q_{\ell-1}/q_\ell \ge \rho$ for all $\ell \in [L]$ and Let f be some Boolean function whose circuit has at most L multiplication levels, and let $c = \text{FHE.Eval}_{evk}(f, c_1, \ldots, c_t)$. Then $\text{FHE.Dec}_{sk}(c) = f(m_1, \ldots, m_2)$, with the condition that $\rho = 2^{\omega(\log(\kappa))} \cdot \max\{B^2, B_{\text{eval}}\}$, Furthermore, $\text{noise}_{q_L}(c, s_L) \le \rho$.*

Proof. We investigate the noise of the intermediate ciphertexts created during evaluation of the circuit and show its magnitude never exceeds ρ and its parity corresponds to the correct bit for the corresponding wire in the circuit.

Initial Noise. For the initial ciphertext $c_i = ((v_i, w_i), 0)$, and $v_i = w_i s + 2e_i + m_i$, where $\|e_i\|_\infty \le B$, thus we have that $\text{noise}_{q_0}(c_i, s_0) \le B_{\text{init}} = 2B + 1$ and the parity of the noise is m_i by assumption, we set $\rho \ge 2B + 1$.

Multiplicative Noise. Let $c_1 = ((v_1, w_1), \ell - 1)$, $c_2 = ((v_2, w_2), \ell - 1)$ be two ciphertexts such that $\text{noise}_{q_{\ell-1}}(c_j, s_{\ell-1}) \le \rho$, where $j = 1$ or 2, and the parity of the noise is m_1, m_2 respectively. Let $\mathbf{c}_{\text{mult}}$ be the resulting ciphertext after evaluating a multiplication gate on c_1, c_2.

Firstly, let us consider the ciphertexts c_1' and c_2' produced by performing modulus reduction. Namely, by applying Theorem 3, we see that $\text{noise}_{q_\ell}(c_1', s_{\ell-1}) < \text{noise}_{q_{\ell-1}}(c_1, s_{\ell-1})/\rho + l_1(s_{\ell-1}) \le nB + 1$.

Similarly, we can easily get the bound of $\text{noise}_{q_\ell}(c_2', s_{\ell-1})$. Furthermore, the parity of the noise remains the same.

Now we let $\mathbf{c}_{\text{mult}} = (\mathbf{v}_{\text{mult}}, \mathbf{w}_{\text{mult}}) = (\lambda_0 - \langle \mu, \xi_{1,\ell} \rangle - \langle \nu, \zeta_{1,\ell} \rangle, \langle \mu, \xi_{0,\ell} \rangle + \langle \nu, \zeta_{0,\ell} \rangle)$. Note that $\mathbf{v}_{\text{mult}} - \mathbf{w}_{\text{mult}} s_\ell = \lambda_0 + \lambda_1 s_{\ell-1} + \lambda_2 s_{\ell-1}^2 + 2(\sum_{\tau=0}^{\lfloor \log q_{\ell-1} \rfloor} \mu_\tau e_{\ell,\tau} + \sum_{\tau=0}^{\lfloor \log q_{\ell-1} \rfloor} \nu_\tau e'_{\ell,\tau})$. As before, $\lambda_0 + \lambda_1 s_{\ell-1} + \lambda_2 s_{\ell-1}^2 (\text{mod } 2) = \phi_{c_1}(s_{\ell-1}) \phi_{c_2}(s_{\ell-1})$, and the noise of $\lambda_0 + \lambda_1 s_{\ell-1} + \lambda_2 s_{\ell-1}^2$ is bounded by $(nB + 1)^2$. Therefore, $\text{noise}_{q_\ell}(\mathbf{c}_{\text{mult}}, s_\ell) \le B_{\text{mult}}$, where $B_{\text{mult}} = (nB+1)^2 + 4(\log q_{\ell-1} + 1) B_{\text{eval}} \le \rho$ and its parity again remains $m_1 m_2$.

Additive Noise. In each level, we assume there are at most $\eta = \text{poly}(\kappa)$ additions. Then the output of an addition gate is bounded by $B_{\text{add}} = \eta \cdot \max\{B_{\text{init}}, B_{\text{mult}}\}$. Since $B_{\text{init}}, B_{\text{mult}} = \rho / 2^{\omega(\log \kappa)}$, we get that $B_{\text{add}} \le \rho$.

Therefore we get the result that in order to realize our modified FHE scheme perfectly employing the technique of re-linearization and modulus reduction, we have to bound the magnitude of noise to $\rho = 2^{\omega(\log(\kappa))} \cdot \max\{B^2, B_{\text{eval}}\}$. □

4.2. Security analysis

From the above fully Homomorphic Encryption, we can see the evk is so complex that we may worry about the security. In fact, in the evk, since the element $\xi_{1,\ell,\tau} = b_{\ell,\tau} - 2^\tau s_{\ell-1} \in R_{q_{\ell-1}}$, which can be thought of as pseudo-encryption of multiples of the secret key $s_{\ell-1}$, we can see the scheme owns the property of circular security, namely the scheme can securely encrypt polynomial functions (over an appropriately defined ring) of its own secret key.

Since the view of the attacker consists of $pk = (a_0, b_0 = -(a_0 s_0 + 2e_0))$ and $c = ((v, w), \ell)$, where $v = b_0 u + 2e_1 + m$ and $w = -(a_0 u + 2e_2)$, as $a_0 \leftarrow R_q$ and the errors $e_1, e_2 \leftarrow \chi$. By RLWE assumption, we know v and w are both RLWE samples and pseudorandom. Consequently, the above scheme is semantically secure with pseudorandom ciphertexts, which means, given pk, a ciphertext of a chosen message is indistinguishable from a uniformly random ciphertext.

5. Threshold fully homomorphic encryption scheme

5.1. Key homomorphic properties

In this part, we describe the useful key-homomorphic properties of the FHE scheme, which play an important role in constructing a threshold scheme.

Let s, s' be two secret keys, and e_0, e_0' be two error ring elements from χ. First of all, we keep a_0 fixed. Note that: $pk = (a_0, b_0 = -(a_0 s + 2e_0)) = \text{FHE.PubKeygen}(s; a_0; e_0)$; $pk' = (a_0, b_0' = -(a_0' s' + 2e_0')) = \text{FHE.PubKeygen}(s'; a_0; e_0')$.

We get $(a_0, b_0 + b'_0) = (a_0, -a_0(s + s') - 2(e_0 + e'_0)) =$ FHE.PubKeygen$(s + s'; a_0; e_0 + e'_0)$, thus we get our combined $pk^* = (a_0, b_0 + b'_0)$, and its corresponding combined $sk^* = s + s'$. With the same approach, we get combined evaluation key evk^*. Therefore the sum of two related keys gives a new valid key pair.

5.2. Construction of threshold fully homomorphic encryption scheme

With the useful key-homomorphic properties, we can easily make our fully homomorphic scheme convert into a threshold scheme. In a TFHE scheme, since the construction of the combined evaluation key is complex, and it needs each party to carefully release some extra information about its key-share. In our TFHE.Keygen stage, we assume that there exists a trusted third party Functionality F_{KeyGen}, which computes the combined public key, secret key, and evaluation key honestly, then it publishes the combined public key and evaluation key to each party, and keeps the combined secret key secret. Moreover, in order to guarantee semantic security, we add some additional smudging noise during sensitive operations.

Common setup. Let $a_0 \leftarrow R_q$, for $i \in [N]$, $\ell \in \{0, \ldots, L\}$, $\tau \in \{0, \ldots, \lfloor \log q_{\ell-1} \rfloor\}$, we sample $a^{(i)}_{\ell,\tau}, a'^{(i)}_{\ell,\tau} \leftarrow R_{q_\ell}$. All parties share a common setup consisting of:

• *Params* $= (a_0, \{a^{(i)}_{\ell,\tau}, a'^{(i)}_{\ell,\tau}\}_{\ell,\tau,i}, \{\text{params}_\ell = (1^\kappa, q_\ell, n, \chi)\}_{0\le\ell\le L}, B, B^{eval}_{smdg}, B^{\text{enc}}_{smdg}, B^{\text{dec}}_{smdg})$.

Where params_ℓ are parameters for the FHE scheme with different moduli q_ℓ. χ is B-bounded, and $B^{eval}_{smdg}, B^{\text{enc}}_{smdg}, B^{\text{dec}}_{smdg}$ are bounded for extra smudging noise.

TFHE.Keygen. We assume for now that generation and distribution of keys and key shares to parties are computed by the Functionality F_{KeyGen}.

Sample different ring elements $s^{(i)}_0, \ldots, s^{(i)}_L, e^{(i)}_0 \leftarrow \chi$. Set $sk_i = (s^{(i)}_0, \ldots, s^{(i)}_L)$ to each party and compute $pk_i = (a_0, b^{(i)}_0 = -(a_0 s^{(i)}_0 + 2e^{(i)}_0))$, while the evaluation key is computed as before, using the $a^{(i)}_{\ell,\tau}, a'^{(i)}_{\ell,\tau}$ in params instead of sampling them randomly from $R_{q_{\ell-1}}$, we get $evk_i = \{\xi^{(i)}_{0,\ell}, \xi^{(i)}_{1,\ell}, \zeta^{(i)}_{0,\ell}, \zeta^{(i)}_{1,\ell}\}_{\ell\in[L]}$.

• When receiving pk_i, sk_i, evk_i from all parties, the Functionality F_{KeyGen} honestly computes the combined key: $pk^* = (a_0, b^*_0 = \sum_{i=1}^N b^{(i)}_0)$, $sk^* = (s_0, \ldots, s_L)$, where $s_j = \sum_{i=1}^N s^{(i)}_j$. With the sk^*, Functionality F_{KeyGen} can compute the combined evaluation key. In more detail, for all $\ell \in [L]$, $\tau \in \{0, \ldots, \lfloor \log q_{\ell-1} \rfloor\}$, sample $a_{\ell,\tau}, a'_{\ell,\tau} \leftarrow R_{q_{\ell-1}}$, $e_{\ell,\tau}, e'_{\ell,\tau} \leftarrow \chi$, and it adds smudging noise e and e' which are bounded by B^{eval}_{smdg}. Compute as follows.

$\xi_{0,\ell,\tau} = a_{\ell,\tau}$, $\xi_{1,\ell,\tau} = -(a_{\ell,\tau} s_\ell + 2e_{\ell,\tau} + 2e) - 2^\tau s_{\ell-1} \in R_{q_{\ell-1}}$; $\zeta_{0,\ell,\tau} = a'_{\ell,\tau}$, $\zeta_{1,\ell,\tau} = -(a'_{\ell,\tau} s_\ell + 2e'_{\ell,\tau} + 2e') - 2^\tau s^2_{\ell-1} \in R_q$.

let $\xi_{0,\ell} = (\xi_{0,\ell,0}, \ldots, \xi_{0,\ell,\lfloor \log q_{\ell-1}\rfloor})$ and $\xi_{1,\ell} = (\xi_{1,\ell,0}, \ldots, \xi_{1,\ell,\lfloor \log q_{\ell-1}\rfloor})$.

we also let $\zeta_{0,\ell} = (\zeta_{0,\ell,0}, \ldots, \zeta_{0,\ell,\lfloor \log q_{\ell-1}\rfloor})$ and $\zeta_{1,\ell} = (\zeta_{1,\ell,0}, \ldots, \zeta_{1,\ell,\lfloor \log q_{\ell-1}\rfloor})$.

At last, the Functionality F_{KeyGen} gets $evk^* = \{\xi_{0,\ell}, \xi_{1,\ell}, \zeta_{0,\ell}, \zeta_{1,\ell}\}_{\ell\in[L]}$.

• Finally, the Functionality F_{KeyGen} sends the combined pk^*, evk^* to all parties.

TFHE.Enc$_{pk^*}(m)$: Once the Functionality F_{KeyGen} generates the $pk^* = (a_0, b^*_0)$, anyone can encrypt as follows: choose $(v, w) \leftarrow$ *FHE.Enc$_{pk^*}(m)$*, with additional smudging errors e^*_1, e^*_2 bounded by B^{enc}_{smdg}, output $\mathbf{c} = ((v + 2e^*_1, w + 2e^*_2), 0)$.

TFHE.Eval$_{evk^*}(f, c_1, \ldots, c_t)$: The evaluation algorithm is the same as before.

TFHE.Dec(c). Initially all parties hold a common ciphertext $\mathbf{c} = ((v, w), L)$. Moreover, each party P_i holds its share $s^{(i)}_L$ for the joint secret key $s^*_L = s^{(1)}_L + \cdots + s^{(N)}_L$.

• Each party P_i broadcasts the decryption share $z_i = ws^{(i)}_L + 2e_k$ for some noise e_i bounded by B^{dec}_{smdg}.

• Given $z_1, \ldots, z_N$, each P_i computes $z = \sum_{i=1}^N z_i$, and outputs $m = [v - z]_{q_L} (\text{mod } 2)$.

Correctness.

Without loss of generality, we assume $\mathbf{c} = ((v, w), L)$ is an encryption of m under secret key s^*_L, then $v = ws^*_L + 2e^* + m$. Let $\text{noise}_{q_L}(\mathbf{c}, s^*_L) \le \rho^*$, then $\text{TFHE.Dec}_{s^*_L}(\mathbf{c}) = m$, as long as ρ^* is far less than $q_L/2$. Let $z_i = ws^{(i)}_L + 2e_k$, we get that:

ShareCombine$(c, z_1, \ldots, z_N) = v - \sum_{i=1}^N z_i = (ws^*_L + 2e^* + m) - \sum_{i=1}^N (ws^{(i)}_L + 2e_k) = m + 2e^* - 2\sum_{i=1}^N e_k$.

Thus we can ensure the correctness of decryption, as long as the magnitude of noise is less than $q_L/2$, namely $\rho^* + 2NB^{\text{dec}}_{smdg} < q_L/2$.

5.3. Security of joint keys

With the RLWE assumption, we can prove the ciphertexts are pseudorandom. Now we show a useful secure property of combining public keys. We assume there exists a malicious adversary among the N parties, denoted by $\mathcal{A}$. For simplicity, we suppose $pk_i = (a_0, b^{(i)}_0 = -(a_0 s^{(i)}_0 + 2e^{(i)}_0))$ $(i \in [N-1])$ are chosen honestly and $\mathcal{A}$ can adaptively choose some value $b'_0 = -(a_0 s'_0 + 2e'_0)$ for which it must know the corresponding s' and an error e'_0. Then the combined public key $pk^* = (a_0, b^*_0 = \sum_{i=1}^{N-1} b^{(i)}_0 + b'_0)$ may not be at all distributions like a correct public key. We define an experiment $JoinKeyS_{\mathcal{A}}(params, B, B^{\text{enc}}_{smdg})$ between $\mathcal{A}$ and a challenger as follows:

1. The challenger can get the $N-1$ honest public keys and gives $(a_0, \sum_{i=1}^{N-1} b^{(i)}_0)$ to $\mathcal{A}$.
2. $\mathcal{A}$ adaptively chooses b'_0, s', e'_0 so that $b'_0 = -(a_0 s' + 2e'_0)$. It also chooses $m \in \{0, 1\}$ and gives (b'_0, s', e'_0, m) to the challenger.
3. The challenger gets $pk^* = (a_0, b^*_0 = \sum_{i=1}^{N-1} b^{(i)}_0 + b'_0)$. It chooses a random bit $\alpha \leftarrow \{0, 1\}$. If $\alpha = 0$ it chooses v^*, w^* uniformly random from R_q. Else it chooses $(v, w) \leftarrow \text{FHE.Enc}_{pk^*}(m)$, with additional smudging e^*_1, e^*_2 bounded by B^{enc}_{smdg}, set $v^* = v + 2e^*_1$, $w^* = w + 2e^*_2$.
4. $\mathcal{A}$ gets (v^*, w^*) and outputs a bit α'.

The output of the experiment is 1 if $\alpha' = \alpha$, and 0 otherwise. We define $\mathcal{A}$ win the experiment as advantage $|\Pr[JoinKeyS_{\mathcal{A}}(params, B, B^{\text{enc}}_{smdg}) = 1] - 1/2|$.

Now we prove that the TFHE scheme can be protected from key-related attacks, namely we can ensure security of joint keys. This means $\mathcal{A}$ cannot distinguish public-key encryptions under the dishonest combined key $pk^* = (a_0, b^*_0 = \sum_{i=1}^{N-1} b^{(i)}_0 + b'_0)$ from uniformly random ones. Indeed, we can only show that the above holds if the ciphertext under the combined key is smudged with additional large noise. We detail the above security property formally via the following theorem.

Theorem 5. *Suppose the above threshold fully homomorphic encryption scheme can be decrypted correctly. Let B, B^{enc}_{smdg} be integers and $B/B^{\text{enc}}_{smdg} = \text{negl}(\kappa)$. Then, for any probabilistic polynomial time adversary $\mathcal{A}$, $\mathcal{A}$'s the advantage of winning the key-related attacks is $|\Pr[\text{JoinKeyS}_{\mathcal{A}}(params, B, B^{\text{enc}}_{smdg}) = 1] - 1/2| = \text{negl}(\kappa)$.*

Proof. We assume $\mathcal{A}$ has probability ϵ of distinguishing public-key encryptions under the dishonest combined public key from uniformly random ones. When the challenger gives $(a_0, \sum_{i=1}^{N-1} b^{(i)}_0)$ to $\mathcal{A}$. It adaptively chooses b'_0, s', e'_0 satisfying $b'_0 = -(a_0 s' + 2e'_0)$ and it also chooses $m = 0$, then gives (b'_0, s', e'_0, m) to the challenger. The challenger then computes the combined public key $pk^* = (a_0, b^*_0 = \sum_{i=1}^{N-1} + b'_0)$ and executes operations as follows. In case of $\alpha = 0$, the challenger gets v^*, w^* uniformly random from R_q. In case of $\alpha = 1$, the challenger computes $(v, w) = (b^*_0 u + 2e_1, -(a_0 u + 2e_2))$, with e_1, e_2 bounded by B, then it selects e^*_1, e^*_2 which

Table 1
Performance comparison.

Schemes	Magnitude of noise	Size of ciphertext
FHE in [11]	$AB_{\text{init}}^{2^L}$	$L+2$
SHE in [13]	$2\sqrt{A}B_{\text{init}}^{L+1}(\sqrt{2n})^L$	$L+2$
Our FHE	AB_{init}	2

are bounded by B_{smdg}^{enc}, let $c^* = (v^*, w^*)$, where $v^* = b_0^* u + 2e_1 + 2e_1^*$, $w^* = -(a_0 u + 2e_2 + 2e_2^*)$. Lastly, it sends (v^*, w^*) to $\mathcal{A}$ and outputs the bit α' produced by $\mathcal{A}$.

Obviously, if $\alpha = 0$, then (v^*, w^*) is just uniformly random. If $\alpha = 1$, by Lemma 2, we get $v^* = b_0^* u + 2e_1 + 2e_1^*$ is statistically close to $b_0^* u + 2e_1^*$, and $w^* = -(a_0 u + 2e_2 + 2e_2^*)$ is statistically close to $w^* = -(a_0 u + 2e_2^*)$. Therefore, the reduction acts indistinguishably from the real challenger with challenge bit α. Hence, if $\mathcal{A}$ wants to win the experiment with probability ϵ, it has to break pseudorandomness of ciphertexts with the least probability ϵ. Therefore, we build a reduction to the pseudorandom ciphertexts property of the TFHE scheme, we get the result $|\Pr[\text{JoinKeyS}_{\mathcal{A}}(params, B, B_{smdg}^{\text{enc}}) = 1] - 1/2| = \text{negl}(\kappa)$. □

6. Performance comparison

In this section, we give the detail of performance comparison among our modified FHE scheme, the FHE scheme of [11] and the SHE scheme of [13] in Table 1. Here, we assume that L is the depth of the circuit to be evaluated, B_{init} is the initial magnitude of the noise, clearly, $\text{noise}_q(c, s) = [v - ws]_q$ is bounded by B_{init}, where $c = ((v, w), 0)$. Given two initial ciphertexts, after L levels of multiplication followed by A additions, in the FHE scheme of [11], the noise grows from an initial magnitude of B_{init} to $AB_{\text{init}}^{2^L}$, the final ciphertext contains $L+2$ ring elements. In the SHE scheme of [13], the noise grows from an initial magnitude of B_{init} to $2\sqrt{A}B_{\text{init}}^{L+1}(\sqrt{2n})^L$, the final ciphertext also contains $L+2$ ring elements. While in our modified FHE scheme, we use a modulus reduction technique to keep the noise level constant. After L levels of multiplication and scaling, the noise magnitude is still B_{init}, but the modulus is down to q/B_{init}^L, followed by A additions, the final magnitude of the noise is AB_{init}. With the relinearization technique, our final ciphertext still contains two ring elements. Therefore, from the communication overhead and the usability perspective, our FHE scheme is superior to previous schemes.

7. Conclusion

In this paper, we exploit re-linearization and modulus reduction techniques to modify the FHE from Brakerski's scheme, and extend our modified FHE to a TFHE scheme. With the re-linearization technique, we keep the size of ciphertexts created during evaluation of the circuit constant. With the modulus reduction technique, we manage the magnitude of the noise to ensure its decryption successfully. We also prove that our TFHE scheme is achieved security against key-related attacks. We will be devoted to improving the computation efficiency in our future work, so as to make our FHE and TFHE schemes more practical.

Acknowledgments

The authors would like to thank the reviewers for their detailed reviews and constructive comments, which have helped improve the quality of this paper. This work is supported by the Science and Technology on Communication Security Laboratory Foundation (Grant No. 9140C110301110C1103) and the National Natural Science Foundation of China (No. 61370203).

References

[1] R. Rivest, L. Adleman, M. Dertouzos, On date banks and privacy homomorphisms, in: FOCS, Academic Press, 1978, pp. 169–177.
[2] C. Gentry, A fully homomorphic encryption scheme, Ph.D. Thesis, Stanford University, 2009.
[3] C. Gentry, Fully homomorphic encryption using ideal lattices, in: M. Mitzenmacher (Ed.), STOC, ACM, 2009, pp. 169–178.
[4] M. Dijk, C. Gentry, S. Halevi, V. Vaikuntanathan, Fully homomorphic encryption over the integers, in: EUROCRYPT 2010, pp. 24–43.
[5] N.P. Smart, F. Vercauteren, Fully homomorphic encryption with relatively small key and ciphertext sizes, in: PKC 2010, Vol. 6056, pp. 420–443.
[6] D. Stehle, R. Steinfeld, Faster fully homomorphic encryption, in: ASIACRYPT 2010, Vol. 6477, pp. 377–394.
[7] C. Gentry, S. Halevi, Implementing Gentry's fully homomorphic encryption scheme, in: EUROCRYPT 2011, Vol. 6632, pp. 129–148.
[8] J.S. Coron, A. Mandal, D. Naccache, M. Tibouchi, Fully homomorphic encryption over the integers with shorter public keys, in: CRYPTO 2011, Vol. 6841, pp. 487–504.
[9] Z. Brakeraki, V. Vaikuntanathan, Efficient fully homomorphic encryption from (standard) LWE, in: FOCS, 2011.
[10] Z. Brakerski, C. Gentry, V. Vaikuntanathan, Fully homomorphic encryption without bootstrapping, in: ITCS, 2012.
[11] Z. Brakerski, V. Vaikuntanathan, Fully homomorphic encryption from ring-LWE and security for key dependent messages, in: CRYPTO 2011, pp. 505–524.
[12] V. Lyubashevsky, C. Peikert, O. Regev, On ideal lattices and learning with errors over rings, in: Gilbert 2010, pp. 1–23.
[13] M. Naehrig, K. Lauter, V. Vaikuntanathan, Can homomorphic encryption be practical? in: CCSW 2011, ACM, 2011, pp. 113–124.
[14] Z. Brakerski, Fully homomorphic encryption without modulus switching from classical GapSVP, in: CRYPTO 2012, in: LNCS, vol. 7417, pp. 868–886.
[15] R. Cramer, I. Damgård, J.B. Nielsen, Multiparty computation from threshold homomorphic encryption, in: EUROCRYPT 2001, pp. 280–299.
[16] I. Damgård, J.B. Nielsen, Universally composable efficient multiparty computation from threshold homomorphic encryption, in: CRYPTO 2003, pp. 247–264.
[17] S. Myers, M. Sergi, A. Shelat, Threshold fully homomorphic encryption and secure computation, 2011, eprint arXiv:2011/454.
[18] G. Asharov, A. Jain, D. Wichs, Multiparty computation with low communication,computation and interaction via threshold FHE, in: EUROCRYPT 2012, in: LNCS, vol. 7237, pp. 483–501.
[19] A. López-Alt, E. Tromer, V. Vaikuntanathan, Cloud-assisted multiparty Computation from fully homomorphic encryption. Cryptology ePrint Archive, Report 2011/663, 2011, http://eprint,iacr,org/.

Xiaojun Zhang received his B.Sc. degree in mathematics and applied mathematics at Hebei Normal University in 2009, PR China and received M.Sc. degree in pure mathematics at Guangxi University in 2012. He is a Ph.D. degree candidate in information security at University of Electronic Science Technology of China (UESTC). He is presently engaged in cryptography, network security and cloud computing security.

Chunxiang Xu received her B.Sc., M.Sc. and Ph.D. degrees at Xidian University, in 1985, 1988 and 2004 respectively, PR China. She is presently engaged in information security, cloud computing security and cryptography as a professor at University of Electronic Science Technology of China (UESTC).

Chunhua Jin received her B.Sc. degree in telecommunication at Northwestern Polytechnical University in 2007, PR China and received M.Sc. degree in Xidian University, in 2011. She is a Ph.D. degree candidate in information security at University of Electronic Science Technology of China (UESTC). She is presently engaged in cryptography, network security and cloud computing security.

Run Xie received his M.Sc. degree in mathematics and applied mathematics at Southwest Jiaotong University in 2006, PR China. He is a Ph.D. degree candidate in information security at University of Electronic Science Technology of China (UESTC). He is presently engaged in cryptography, network security and cloud computing security.

Jining Zhao received his B.Sc. degree in information and computing science at Henan Normal University in 2009, PR China. He is a M.Sc. degree candidate in information security at University of Electronic Science Technology of China (UESTC). He is presently engaged in cloud computing security, network security and cryptography.