An Improved Leveled Fully Homomorphic Encryption Scheme over the Integers

Xiaoqiang Sun(✉), Peng Zhang, Jianping Yu, and Weixin Xie

ATR Key Laboratory of National Defense Technology,
College of Information Engineering, Shenzhen University,
Shenzhen 518060, Guangdong, China
{xqsun,zhangp,yujp,wxxie}@szu.edu.cn

Abstract. A scale-invariant leveled fully homomorphic encryption (FHE) scheme over the integers is proposed by Coron *et al.* in PKC 2014, where the ciphertext noise increases linearly after each homomorphic multiplication. Then based on Coron's variant of the approximate greatest common divisor problem, we construct a more efficient leveled FHE scheme over the integers without the modulus switching technique, which could resist chosen plaintext attacks. The inner product operation in our homomorphic multiplication is eliminated by multiplying the multiplication key directly. The homomorphic multiplication in our scheme is realized by the more simplified multiplication key, in which the number of integers is decreased from $O(\Theta \cdot \eta)$ to $O(1)$ compared with Coron's scheme. Simulation results and analysis show that our scheme's performance of multiplication key and homomorphic multiplication is much more efficient than that of Coron's scheme.

Keywords: Leveled Fully Homomorphic Encryption
Approximate GCD · Homomorphic multiplication · Multiplication key

1 Introduction

Traditional public key encryption schemes are constructed based on several mathematical problems. For example, RSA [1] is based on the large integer factorization problem, and ElGamal [2] is constructed by the discrete logarithm problem. However, traditional public key encryption schemes don't support arbitrary operations on the ciphertext without the secret key. Homomorphic encryption (HE) allows computations on the ciphertext without decryption, which was originated in 1978 by Rivest, Adleman and Rertouzos [3]. Because of this special property, HE can be used in cloud computing, ciphertext search and etc. However, HE only supports finite homomorphic multiplications or homomorphic additions, for example, BGN [4] supports infinite homomorphic additions and once homomorphic multiplication. The first fully homomorphic encryption (FHE) scheme based on ideal lattices was proposed by Gentry [5] in 2009, which could support infinite homomorphic multiplications and homomorphic additions.

J. K. Liu and P. Samarati (Eds.): ISPEC 2017, LNCS 10701, pp. 835–846, 2017.
https://doi.org/10.1007/978-3-319-72359-4_52

However, the construction of the FHE scheme based on ideal lattices is complicated, which induces sizes of ciphertext, public key and secret key excessive long. In 2010, Dijk *et al.* [6] introduced a new FHE scheme over the integers (DGHV). DGHV only applies trivial operations on the integers and its security can be reduced to the approximate greatest common divisor (GCD) problem. The somewhat homomorphic encryption (SWHE) scheme in the ref. [7] converted DGHV scheme's public key into the form of quadratic, thus the number of the integers in the public key is decreased from τ to $2\sqrt{\tau}$. Coron *et al.* [8] decreased the public key size by using the public key compression technique. And the modulus switching technique is applied to replace Gentry's squashing decryption circuit technique. It can be noticed that FHE scheme is complicated when the modulus switching technique [9] is applied to DGHV scheme. Coron *et al.* [10] proposed a variant of DGHV scheme with the scale-invariant property, which security is also based on the approximate GCD problem. This scheme doesn't use the modulus switching technique, which requires no huge storage space for public keys. The ciphertext's noise increases linearly after each homomorphic multiplication. Above refs. [6–8,10] discuss single-bit FHE schemes over the integers.

The FHE scheme in the ref. [11] described a batch DGHV scheme based on Chinese Remainder Theorem (CRT). However, these FHE schemes' public key size is $O(\lambda^7)$ and secret key size is $O(\lambda^9)$, which are too far for practical application. Cheon *et al.* [12] also proposed a batch FHE scheme based on the CRT over rings, which could resist the approximate GCD problem and the sparse subset sum problem (SSSP) attack. The overhead of the SWHE scheme in the ref. [12] is small, whose public key size is similar to that of DGHV scheme, however it is still too large.

The security of former FHE schemes over the integers can be only reduced to the approximate GCD problem. Meanwhile, some more efficient FHE schemes have been constructed based on the learning with errors (LWE) assumption. Jacob [13] improved the LWE-based FHE scheme in the ref. [14] by using symmetric groups and permutation matrices with fast bootstrapping speed, whose ciphertext noise increases polynomially. Then, Ducas and Micciancio proposed a faster bootstrapping method [15] based on the ref. [13] over ring. In 2015, Cheon and Stehlé [16] reduced the LWE assumption to the approximate GCD problem innovatively. It means that the approximate GCD problem is no easier than the LWE assumption. And the LWE assumption has the advantage of resisting quantum attack. Then he constructed a new FHE scheme [16] based on the improved approximate GCD problem without Gentry's technique of squashing decryption circuit [5], which ciphertext size is only $O(\lambda \log \lambda)$.

1.1 Contribution

In this paper, we still use the classical approximate GCD problem to improve the efficiency of FHE schemes over the integers. Our contributions consist of two parts, which are shown as follows.

On the one hand, based on the ref. [10]'s variant of the approximate GCD problem, we present a more efficient leveled FHE scheme over the integers without the modulus switching technique. To compress the size of the multiplication key, we decrease the number of integers in the multiplication key. Then, the homomorphic multiplication can be achieved by multiplying the multiplication key directly without the inner product in the ref. [10].

On the other hand, we implement the homomorphic multiplication on the personal computer, and demonstrate the efficiency of our scheme's homomorphic multiplication according to the detailed analysis.

1.2 Organization

The remainder of the paper is organized as follows. The preliminary is introduced in Sect. 2. In Sect. 3, an improved leveled FHE scheme over the integers is presented. The security analysis is given in Sect. 4. Section 5 shows simulation results and analysis. The whole paper is concluded in Sect. 6.

2 Preliminary

2.1 Basic Symbols

Given the security parameter λ, let lowercase English letters denote real number and integer, and uppercase English letters denote matrix.

For a real number z, let $\lceil z \rceil$, $\lfloor z \rfloor$, $\lfloor z \rceil$ denote the rounding of an up, down or the nearest integer, namely, they are the integers in the half open intervals $[z, z+1)$, $(z-1, z]$, $(z-1/2, z+1/2]$ respectively.

For a real number z and an integer p, let $q_p(z)$ and $r_p(z)$ denote the remainder of z with respect to p, namely $q_p(z) = [z/p]$ and $r_p(z) = z - q_p(z) \cdot p$. Note that $r_p(z) \in (-p/2, p/2]$. $[z]_p$ or $z \mod p$ also denotes the remainder.

Given an m-dimensional vector $\boldsymbol{a} = (a_0, a_1, \cdots, a_{m-1})$, let $BitDecomp(\boldsymbol{a}) = (a_{0,0}, \cdots, a_{0,l-1}, \cdots, a_{m-1,0}, \cdots, a_{m-1,l-1})$, where $a_{i,j}$ is a_i's j-th bit and ordered from the least significant bit to the most significant bit, $l = \lceil \log q \rceil$. Let $Powersof2(\boldsymbol{a}) = (a_0, 2a_0, \cdots, 2^{l-1}a_0, \cdots, a_{m-1}, 2a_{m-1}, \cdots, 2^{l-1}a_{m-1})$.

Lemma 1 (Simplified Leftover Hash Lemma [17])**.** Let H be a family of 2-universal hash functions from X to Y. Suppose that $h \xleftarrow{R} H$ and $x \xleftarrow{R} X$ are chosen uniformly and independently. Then, $(h, h(x))$ is $\sqrt{|Y|/|X|}/2$-uniform over $H \times Y$.

2.2 Parameters

We use following five parameters (all polynomials of the security parameter λ) will be used in this paper [10,16]:

- η is the bit-length of the secret key. Let $\eta \geq \rho + O(L \log \lambda)$, in order to make the depth of the squashed decryption circuit less than that of the permitted circuit, where L is the multiplicative depth of the circuit to be evaluated;
- ρ is the bit-length of the first noise parameter. Let $\rho = \eta - L \log \lambda$, for reduction to the LWE assumption, where L is the multiplicative depth of the circuit to be evaluated;
- γ is the bit-length of an approximate GCD sample. Let $\gamma \geq \frac{\lambda}{\log \lambda}(\eta - \rho)^2$, to thwart various lattice-based attacks on the approximate GCD problem;
- τ is the number of integers in the public key. Let $\tau \geq \gamma + 2\lambda$, in order to apply the simplified leftover hash lemma.

To satisfy the constraints of above parameters, for convenience, we let $\rho = O(\lambda)$, $\eta = O(\lambda + L)$, $\gamma = O(L^2\lambda + \lambda^2)$ and $\tau = \gamma + 2\lambda$.

2.3 Approximate GCD

Definition 1 (Approximate GCD [10])**.** For a random η-bit secret number p, an integer q uniformly distributed in $[0, 2^\gamma/p^2)$, and an error distribution χ. The distribution $A_{q,\chi}^{APGCD}(p)$ is defined as follows: select q from $\mathbb{Z} \bigcap [0, 2^\gamma/p^2)$ randomly and small error r from χ, then return $x = q \cdot p^2 + r$. Then, generate samples from $A_{q,\chi}^{APGCD}(p)$ polynomially, output p.

In refs. [6,10,16], it has been proved that there is no effective attack could solve the approximate GCD problem.

2.4 Coron's Scale-Invariant Fully Homomorphic Encryption Scheme over the Integers

In this section, we first recall Coron's scale-invariant fully homomorphic encryption scheme over the integers [10]. For a random $\eta-$bit odd integer p and an integer $q_0 \in [0, 2^\gamma/p^2)$. We use the following distribution:

$$\mathcal{D}_{p,q_0}^{\rho} = \{q \cdot p^2 + r, q \in \mathbb{Z} \bigcap [0, q_0), r \in \mathbb{Z} \bigcap (-2^\rho, 2^\rho)\}.$$

Coron's Leveled Fully Homomorphic Encryption scheme $CLFHE = (KeyGen, Encrypt, Add, Convert, Mult, Decrypt)$ is described as follows:

- $CLFHE.KeyGen(1^\lambda)$: Given the security parameter λ, generate random $\eta-$bit secret key p and a $\gamma-$bit integer $x_0 = q_0 \cdot p^2 + r_0$, where $r_0 \in (-2^\rho, 2^\rho) \bigcap \mathbb{Z}$ and $q_0 \in [0, 2^\gamma/p^2)$. Randomly select the public key $x_i \in \mathcal{D}_{p,q_0}^{\rho}$, where $i = 1, 2, \cdots, \tau$. Let $y' \in \mathcal{D}_{p,q_0}^{\rho}$ and $y = y' + (p-1)/2$. Let $\boldsymbol{z}$ denote a vector of Θ numbers, which keeps $\kappa = 2\gamma + 2$ bits of precision after the binary point. Let $\boldsymbol{s}$ denote a vector such that

$$\frac{2^\eta}{p^2} = \ < \boldsymbol{s}, \boldsymbol{z} > + \varepsilon \mod 2^\eta,$$

where $|\varepsilon| \leq 2^{-\kappa}$. Let

$$\boldsymbol{\sigma} = \boldsymbol{q} \cdot p^2 + \boldsymbol{r} + \lfloor Powersof2(\boldsymbol{s} \cdot \frac{p}{2^{\eta+1}}) \rceil,$$

where elements of $\boldsymbol{q}$ are randomly generated from $[0, q_0) \bigcap \mathbb{Z}$. Output the secret key $sk = \{p\}$ and the public key $pk = \{x_0, x_1, \cdots, x_\tau, y, \boldsymbol{\sigma}, \boldsymbol{z}\}$.

- $CLFHE.Encrypt(pk, m \in \{0,1\})$: Given the public key pk and a random subset $S \subset 1, 2, \cdots, \tau$, output the ciphertext as follows:

$$c = [m \cdot y + \sum\nolimits_{i \in S} x_i]_{x_0}.$$

- $CLFHE.Add(pk, c_1, c_2)$: Given the public key pk, ciphertexts c_1 and c_2, output the fresh ciphertext $c_{fresh} = (c_1 + c_2) \mod x_0$.
- $CLFHE.Convert(pk, c)$: Given the public key pk and the ciphertext c, output $c' = 2 \cdot < \boldsymbol{\sigma}, BitDecomp(\boldsymbol{c}) >$, where $\boldsymbol{c} = (\lfloor c \cdot z_i \rceil \mod 2^\eta)_{1 \le i \le \Theta}$.
- $CLFHE.Mul(pk, c_1, c_2)$: Given the public key pk, ciphertexts c_1 and c_2, output the fresh ciphertext $c_{fresh} = CLFHE.Convert(pk, 2 \cdot c_1 \cdot c_2) \mod x_0$.
- $CLFHE.Decrypt(sk, c)$: Given the ciphertext c and the secret key sk, output the decryption result $m = ((2c) mod\, p) \mod 2$.

3 Leveled Fully Homomorphic Encryption Scheme

3.1 The Construction

Let η' and $\theta = O(\lambda)$ be two more parameters. Our Leveled Fully Homomorphic Encryption scheme $LFHE = (KeyGen,\ Enc, Add, Mul, Dec)$ is defined as follows:

- $LFHE.KeyGen(1^\lambda)$: Given the security parameter λ, generate random η-bit secret key p. Randomly select the public key $x_i \in A_{q,\chi}^{APGCD}(p)$, for $i = 0, 1, \cdots, \tau - 1$, where x_0 is the largest one, $\lfloor \frac{x_1}{p^2} \rceil$ is an odd number. Restart the generation of x_0 and x_1 if they don't satisfy above conditions. The multiplication key is generated as follows:

$$mk = [\lfloor \frac{2^{\eta'}}{p^2} \rceil + 2^{\eta'-1} \cdot p^2 \cdot q]_{x_0},$$

  where $q \in \mathbb{Z} \bigcap [0, 2^\gamma / p^2)$, $\lfloor \frac{2^{\eta'}}{p^2} \rceil \in \chi$. Output the public key $pk = \{(x_i)_{i=0,1,\cdots,\tau-1},\ mk\}$ and the secret key $sk = p$.
- $LFHE.Enc(pk, S, m \in \{0,1\})$: Given the public key pk and the randomly generated subset $S \subset \{1, 2, \cdots, \tau - 1\}$ of size θ, output the ciphertext c as follows:

$$c = [\sum\nolimits_{i \in S} x_i + \lfloor \frac{x_1}{2} \rceil \cdot m]_{x_0}.$$

- $LFHE.Add(pk, c_1, c_2)$: Given the public key pk, ciphertexts c_1 and c_2, output the fresh ciphertext $c_{fresh} = (c_1 + c_2) \mod x_0$.
- $LFHE.Mul(pk, c_1, c_2)$: Given the public key pk, ciphertexts c_1 and c_2, output the fresh ciphertext c_{fresh} as follows:

$$c_{fresh} = (\frac{1}{2^{\eta'-1}} c_1 \cdot c_2 \cdot mk) \mod x_0.$$

– $LFHE.Dec(sk, c)$: Given the ciphertext c and the secret key sk, output the decryption result

$$m = [\lfloor \frac{2c}{p^2} \rceil]_2.$$

3.2 Correctness

Lemma 2 (Encryption noise). Let the key pair (sk, pk) generated by $LFHE.KeyGen(1^\lambda)$ and the ciphertext c generated by $LFHE.Enc(pk, S, m \in \{0,1\})$. Then

$$c' = c(mod\, p^2) = r + \lfloor \frac{p^2}{2} \rceil m (\mathrm{mod}\, p^2),$$

where $|r| \leq (2\theta + 1/2) \cdot 2^\rho + 1/2$.

Proof. We can represent the public key x_i as the form of $x_i = p^2 \cdot q_i + r_i$, where $q_i \in \mathbb{Z} \bigcap [0, 2^\gamma / p^2)$, $i = 1, 2, \cdots, \tau - 1$. Then we have $\lfloor \frac{x_1}{2} \rceil = \frac{p^2 \cdot q_1}{2} + \frac{r_1}{2} + \delta$, where $|\delta| \leq 1/2$. Hence,

$$\begin{aligned} c' &= c(\mathrm{mod}\, p^2) \\ &= (\sum_{i \in S} x_i + \lfloor \frac{x_1}{2} \rceil m - k x_0)(\mathrm{mod}\, p^2) \\ &= (\sum_{i \in S} r_i + \lfloor \frac{p^2}{2} \rceil m + (\frac{r_1}{2} + \delta) m - k r_0)(\mathrm{mod}\, p^2), \end{aligned}$$

where $k \in [0, \theta]$, $|\delta| \leq 1/2$. Consequently, the noise $|r| \leq (2\theta + 1/2) \cdot 2^\rho + 1/2$ for $c' = c(\mathrm{mod}\, p^2) = r + \lfloor \frac{p^2}{2} \rceil m (\mathrm{mod}\, p^2)$.

Lemma 3 (Addition noise). Let the key pair (sk, pk) generated by $LFHE.KeyGen(1^\lambda)$ and the ciphertext c_i generated by $LFHE.Enc(pk, S, m_i)$, where $i = 1, 2$. If $c_{add} = LFHE.Add(pk, c_1, c_2)$, then

$$c_{add} = r + \lfloor \frac{p^2}{2} \rceil (m_1 + m_2) (\mathrm{mod}\, p^2),$$

where $|r| \leq |r_1 + r_2| + 2^\rho + 1$.

Proof. We have $c_i' = c_i (\mathrm{mod}\, p^2) = r_i + \lfloor \frac{p^2}{2} \rceil m_i (\mathrm{mod}\, p^2)$, then

$$\begin{aligned} c_{add} &= c_1 + c_2 - \delta x_0 (\mathrm{mod}\, p^2) \\ &= r_1 + r_2 - \delta r_0 + \lfloor \frac{p^2}{2} \rceil (m_1 + m_2) + \delta' (\mathrm{mod}\, p^2), \end{aligned} \tag{1}$$

where $|\delta| \leq 1$, $|\delta'| \leq 1$, $i = 1, 2$. Hence, the noise $|r| \leq |r_1 + r_2| + 2^\rho + 1$ for $c_{add} = r + \lfloor \frac{p^2}{2} \rceil (m_1 + m_2) (\mathrm{mod}\, p^2)$.

Lemma 4 (Multiplication noise). Given the key pair (sk, pk) and the multiplication key mk generated by $LFHE.KeyGen(1^\lambda)$. The ciphertext c_i satisfying the condition that $c_i' = c_i (\mathrm{mod}\, p^2) = r_i + \lfloor \frac{p^2}{2} \rceil m_i (\mathrm{mod}\, p^2)$, where $i = 1, 2$. Then

$$c_{mul} = [\frac{1}{2^{\eta'-1}} c_1 c_2 (\lfloor \frac{2^{\eta'}}{p^2} \rceil + 2^{\eta'-1} \cdot p^2 q)]_{x_0} = r + \lfloor \frac{p^2}{2} \rceil (m_1 m_2) (\mathrm{mod}\, p^2),$$

where $|r| \leq |r_1| + |r_2| + 2^\rho \cdot (\theta^2 + 2)$.

Proof.

$$\begin{aligned}
c_{mul} &= [\frac{1}{2^{\eta'-1}} c_1 c_2 (\frac{2^{\eta'}}{p^2} + 2^{\eta'-1} \cdot p^2 q)]_{x_0} (\mathrm{mod}\, p^2) \\
&= \frac{1}{2^{\eta'-1}} (p^2 q_1 + r_1 + \lfloor \frac{p^2}{2} \rceil m_1)(p^2 q_2 + r_2 + \lfloor \frac{p^2}{2} \rceil m_2)(\frac{2^{\eta'}}{p^2} + 2^{\eta'-1} p^2 q) - k x_0 (\mathrm{mod}\, p^2) \\
&= \frac{1}{2^{\eta'-1}} (p^4 q_1 q_2 + r_1 p^2 q_2 + \lfloor \frac{p^2}{2} \rceil m_1 p^2 q_2 + p^2 q_1 r_2 + r_1 r_2 + \lfloor \frac{p^2}{2} \rceil m_1 r_2 \\
&+ p^2 q_1 \lfloor \frac{p^2}{2} \rceil m_2 + r_1 \lfloor \frac{p^2}{2} \rceil m_2 + \lfloor \frac{p^2}{2} \rceil^2 m_1 m_2)(\frac{2^{\eta'}}{p^2} + 2^{\eta'-1} \cdot p^2 q) - k x_0 (\mathrm{mod}\, p^2) \\
&= \frac{1}{2^{\eta'-1}} (2 r_1 q_2 \cdot 2^{\eta'-1} + 2 r_2 q_1 \cdot 2^{\eta'-1} + \frac{r_1 r_2}{p^2} \cdot 2^{\eta'} + m_1 r_2 \cdot 2^{\eta'-1} \\
&+ r_1 m_2 \cdot 2^{\eta'-1} + \lfloor \frac{p^2}{2} \rceil m_1 m_2 \cdot 2^{\eta'-1}) - k x_0 (\mathrm{mod}\, p^2) \\
&= 2 r_1 q_2 + 2 q_1 r_2 + 2 \frac{r_1 r_2}{p^2} + m_1 r_2 + r_1 m_2 + \lfloor \frac{p^2}{2} \rceil m_1 m_2 - k r_0 (\mathrm{mod}\, p^2) \\
&= r + \lfloor \frac{p^2}{2} \rceil (m_1 m_2) (\mathrm{mod}\, p^2),
\end{aligned}$$

where $r = 2 r_1 q_2 + 2 q_1 r_2 + 2 \frac{r_1 r_2}{p} + m_1 r_2 + r_1 m_2 - k r_0$, $k \in [0, \theta^2]$. Therefore, the multiplication noise

$$\begin{aligned}
|r| &\leq |r_1| \cdot 2^{\theta+\gamma-\eta+1} + |r_2| \cdot 2^{\theta+\gamma-\eta+1} + 2^\rho \cdot 2 + |r_1| + |r_2| + \theta^2 \cdot \rho \\
&= (|r_1| + |r_2| + 2) \cdot 2^{\theta+\gamma-\eta+1} + 2^\rho \cdot (\theta^2 + 2).
\end{aligned}$$

Because ciphertexts c_1, c_2 and the multiplication key mk are all integers, the noise r is an integer. From above Lemmas 2 and 4, it can be known that our scheme's noise of addition or multiplication increases linearly.

Lemma 5 (Decryption Correctness). Let the secret key p generated by $LFHE.KeyGen(1^\lambda)$ and the ciphertext c generated by $LFHE.Enc(pk, S, m \in \{0, 1\})$. Then

$$LFHE.Dec(sk, c) = m, \text{if } c' = c (\mathrm{mod}\, p^2) = r + \lfloor \frac{p^2}{2} \rceil m (\mathrm{mod}\, p^2),$$

where $|r| < p^2/4$.

Proof. Because $c' = c(\mathrm{mod}\, p^2) = r + \lfloor \frac{p^2}{2} \rceil m(\mathrm{mod}\, p^2)$, we can write as $c = r + \lfloor \frac{p^2}{2} \rceil m + p^2 q$. Then

$$\begin{aligned}[\lfloor \frac{2c}{p^2} \rceil]_2 &= [\lfloor 2q + m + \frac{2r}{p^2} \rceil]_2 \\ &= [\lfloor m + 2(\frac{r}{p^2} + q) \rceil]_2 \\ &= m,\end{aligned}$$

if $|r| < p^2/4$.

4 Security Analysis

Claim 1 (Security). For any parameters ρ, γ, η and τ, which are polynomials of the security parameter λ, the proposed leveled FHE scheme over the integers could resist chosen plaintext attacks (CPA), assumed that $A_{q,\chi}^{APGCD}(p)$ is difficult.

Proof. Let $\mathcal{A}$ be a probabilistic polynomial time (PPT) adversary which could distinguish the challenge ciphertext with the advantage ε. Detailed operations are as follows.

Setup: Take as input the security parameter λ, the challenger runs $LFHE.KeyGen(1^\lambda)$ to get the public key pk_i and the secret key sk_i polynomially, where $i = 1, 2, \cdots, t$, t is the maximum number of query. Then send pk_i and sk_i to $\mathcal{A}$, where $i = 1, 2, \cdots, t$.

Queries 1: The challenger chooses $m_i \in \{0, 1\}$ and the subset $S \subset \{1, 2, \cdots, \tau - 1\}$ randomly, then executes $LFHE.Enc(pk, S, m_i \in \{0, 1\})$ and sends the ciphertext c_i to $\mathcal{A}$, where $i = 1, 2, \cdots, t$.

Challenge: After queries, $\mathcal{A}$ outputs two different plaintexts m_0', $m_1' \in \{0, 1\}$, which have not been queried. Then the challenger chooses a random bit $k \in \{0, 1\}$, and generates the challenge public key pk^* which has not been queried. The challenge ciphertext c^* is obtained by running $LFHE.Enc(pk^*, S, m_k')$, which is generated as follows,

$$c^* = [\sum\nolimits_{i \in S} x_i + \lfloor \frac{x_1}{2} \rceil m_k']_{x_0}.$$

Queries 2: The same as ***Queries 1***, and the challenge public key pk^* can't be queried again.

Output: $\mathcal{A}$ outputs a guess $k' \in \{0, 1\}$. Output 1 if $\mathcal{A}$ guesses right, else 0.

In order to prove the security of the proposed leveled FHE scheme, we need to construct a distinguisher $\mathcal{D}$, namely

$$pk^* \text{ and } Unif(pk).$$

$\mathcal{D}$ takes pk^* and $pk' \in Unif(pk)$ as inputs, where $Unif(pk)$ represents the genuine public key distribution. $\mathcal{D}$ chooses $k \in \{0,1\}$ randomly, then returns the challenge ciphertext $c_k^* = [\sum_{i\in S} x_i + \lfloor \frac{x_1}{2} \rceil m'_k]_{x_0}$.

Assuming $\mathcal{D}$ has the advantage ε to distinguish ciphertexts $c_0^* = [\sum_{i\in S} x_i + \lfloor \frac{x_1}{2} \rceil m'_0]_{x_0}$ and $c_1^* = [\sum_{i\in S} x_i + \lfloor \frac{x_1}{2} \rceil m'_1]_{x_0}$. Because the challenge ciphertext $c_k^* (\mathrm{mod}\, p^2) = r_k + \lfloor \frac{p^2}{2} \rceil m'_k (\mathrm{mod}\, p^2)$, it can be regarded as the form of $c_k^* = x_k^* + r_k$, where $x_k^* = p^2 \cdot q_k + \lfloor \frac{p^2}{2} \rceil m'_k$, $\mathcal{D}$ has the same advantage to distinguish x_0^* and x_1^*. Hence, $\mathcal{D}$ has the same advantage ε to distinguish pk^* and pk', namely $\mathcal{D}$ could solve the approximate GCD problem successfully. In a word, the probability of distinguishing the challenge ciphertext is negligible, the proposed scheme could resist chosen plaintext attacks.

5 Simulation and Analysis

The key indicator of measuring the efficiency of a leveled FHE scheme is homomorphic multiplication. Because our scale-invariant leveled FHE scheme doesn't use the modulus switching technique, we only compare it with Coron's scheme. To compare the efficiency between the proposed scheme and Coron's scheme [10], two schemes are carried out on the same personal computer, and the experimental environment is as follows: the operating system is microsoft windows 7, featuring two Intel (R) Core (TM) i5-3470 CPU processors, running at 3.20 GHz, with 8.00 GB RAM, and the virtual machines operation system is Ubuntu 12.04, featuring single Intel (R) Core (TM) i5-3470 CPU processor, with 4.00 GB RAM. Our implementation uses the GMP large number library for high level numeric algorithms and the code is compiled on the GCC platform by the C++ language.

The implementation time of multiplication key and homomorphic multiplication between our scheme and Coron's scheme is shown in Tables 1 and 2, respectively. Each test has five iterations and datum shown in the tables are averages of them. As seen from Tables 1 and 2, the runtime of our scheme's multiplication key and homomorphic multiplication is reduced several magnitudes compared with Coron's scheme with the increasing of λ. Particularly, the number of integers in the multiplication key is reduced from $O(\Theta \cdot \eta)$ to $O(1)$. The detailed analysis is described as follows.

Table 1. Implementation time of multiplication key between our scheme and Coron's scheme [10] (unit: microsecond).

Security parameter λ	50	70	90	110	130
Coron's scheme	3970	29435	77359	134460	241478
Our scheme	4	13	17	24	30

Table 2. Implementation time of homomorphic multiplication between our scheme and Coron's scheme [10] (unit: microsecond).

Security parameter λ	50	70	90	110	130
Coron's scheme	302	1067	2292	4530	8308
Our scheme	9	26	51	74	107

Figures 1 and 2 show the efficiency of the proposed scheme and Coron's scheme. Simultaneously, two figures also indicate two schemes' changing trends of implementation time with the increasing of λ. As shown in Figs. 1 and 2, it can be easily known that our scheme's efficiency of multiplication key and homomorphic multiplication is much better than Coron's scheme with the increasing of λ. And the increasing tendency of our scheme's time of multiplication key and homomorphic multiplication is slower than that of Coron's scheme with the increasing of λ. In a word, our scheme is more efficient than Coron's scheme.

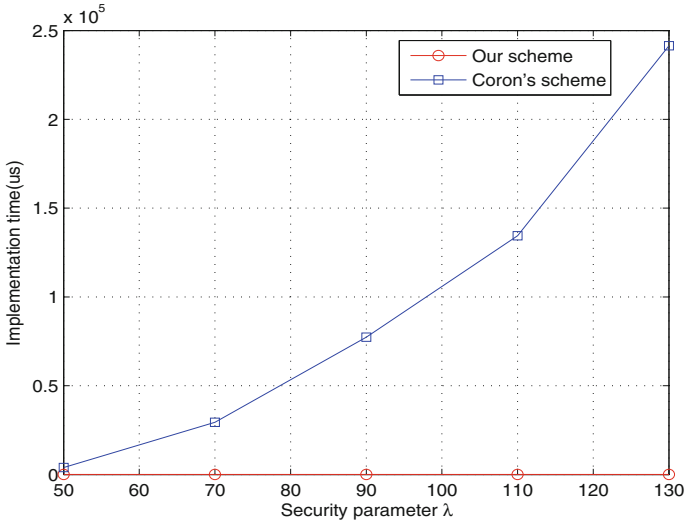

Fig. 1. Efficiency comparison of multiplication key in our scheme and Coron's scheme [10].

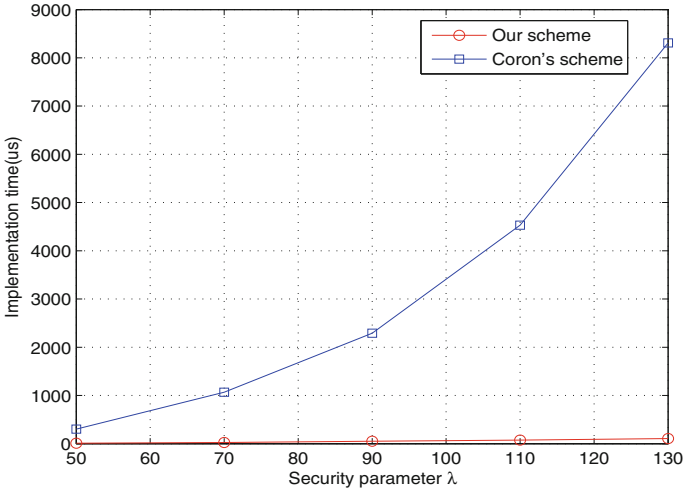

Fig. 2. Efficiency comparison of homomorphic multiplication in our scheme and Coron's scheme [10].

6 Conclusion

In this paper, we propose an efficient leveled FHE scheme over the integers based on Coron's variant of the approximate GCD problem. We prove that the proposed scheme also remains CPA secure under the approximate GCD problem. Compared with Coron's scheme, our scheme decreases the number of integers in the multiplication key from $O(\Theta \cdot \eta)$ to $O(1)$. Then, based on the more simplified multiplication key, the homomorphic multiplication can be efficiently achieved without the inner product. Simulation results and analysis show that our scheme's multiplication key and homomorphic multiplication is superior to Coron's scheme.

Acknowledgements. This work was supported by the National Natural Science Foundation of China (61702342), the Science and Technology Innovation Projects of Shenzhen (JCYJ20160307150216309, JCYJ20170302151321095, GJHZ20160226202520268) and Tencent Rhinoceros Birds-Scientific Research Foundation for Young Teachers of Shenzhen University. We would like to thank Jung Hee Cheon for his valuable comments.

References

1. Rivest, R.L., Shamir, A., Adleman, L.: A method for obtaining digital signatures and public-key cryptosystems. Commun. ACM **21**(2), 120–126 (1978)
2. ElGamal, T.: A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans. Inf. Theory **31**(4), 469–472 (1985)
3. Rivest, R.L., Adleman, L., Dertouzos, M.L.: On data banks and privacy homomorphisms. Found. Secur. Comput. **4**(11), 169–180 (1978)

4. Boneh, D., Goh, E.-J., Nissim, K.: Evaluating 2-DNF formulas on ciphertexts. In: Kilian, J. (ed.) TCC 2005. LNCS, vol. 3378, pp. 325–341. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-540-30576-7_18
5. Gentry, C.: Fully homomorphic encryption using ideal lattices. In: Proceedings of the 41st Annual ACM Symposium on Theory of Computing, pp. 169–178 (2009)
6. van Dijk, M., Gentry, C., Halevi, S., Vaikuntanathan, V.: Fully homomorphic encryption over the integers. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 24–43. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13190-5_2
7. Coron, J.-S., Mandal, A., Naccache, D., Tibouchi, M.: Fully homomorphic encryption over the integers with shorter public keys. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 487–504. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22792-9_28
8. Coron, J.-S., Naccache, D., Tibouchi, M.: Public key compression and modulus switching for fully homomorphic encryption over the integers. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 446–464. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-29011-4_27
9. Brakerski, Z., Gentry, C., Vaikuntanathan, V.: (Leveled) fully homomorphic encryption without bootstrapping. In: Proceedings of the 3rd Innovations in Theoretical Computer Science Conference, vol. 6(3), pp. 309–325 (2012)
10. Coron, J.-S., Lepoint, T., Tibouchi, M.: Scale-invariant fully homomorphic encryption over the integers. In: Krawczyk, H. (ed.) PKC 2014. LNCS, vol. 8383, pp. 311–328. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-54631-0_18
11. Cheon, J.H., Coron, J.-S., Kim, J., Lee, M.S., Lepoint, T., Tibouchi, M., Yun, A.: Batch fully homomorphic encryption over the integers. In: Johansson, T., Nguyen, P.Q. (eds.) EUROCRYPT 2013. LNCS, vol. 7881, pp. 315–335. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-38348-9_20
12. Cheon, J.H., Kim, J., Lee, M.S., Yun, A.: CRT-based fully homomorphic encryption over the integers. Inf. Sci. **310**, 149–162 (2015)
13. Alperin-Sheriff, J., Peikert, C.: Faster bootstrapping with polynomial error. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014. LNCS, vol. 8616, pp. 297–314. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-44371-2_17
14. Gentry, C., Sahai, A., Waters, B.: Homomorphic encryption from learning with errors: conceptually-simpler, asymptotically-faster, attribute-based. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013. LNCS, vol. 8042, pp. 75–92. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40041-4_5
15. Ducas, L., Micciancio, D.: FHEW: bootstrapping homomorphic encryption in less than a second. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015. LNCS, vol. 9056, pp. 617–640. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-46800-5_24
16. Cheon, J.H., Stehlé, D.: Fully homomophic encryption over the integers revisited. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015. LNCS, vol. 9056, pp. 513–536. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-46800-5_20
17. Håstad, J., Impagliazzo, R., Levin, L.A., Luby, M.: A pseudorandom generator from any one-way function. SIAM J. Comput. **28**(4), 1364–1396 (1999)