Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/090329

A Depth Specific Description of Somewhat Homomorphic Encryption and Its Applications

Hyang-Sook Lee[1,*] *and Seongan Lim*[2,*]

[1] Department of Mathematics, Ewha Womans University, Seoul, Korea
[2] Institute of Mathematical Sciences, Ewha Womans University, Seoul, Korea

Received: 30 Jul. 2014, Revised: 31 Oct. 2014, Accepted: 1 Nov. 2014
Published online: 1 May 2015

Abstract: In this paper, we consider the depth-specific description of somewhat homomorphic encryption(SHE) schemes over integers. The ciphertexts of SHE scheme may have various forms depending on its encryption depth, and this makes the correctness check of the encryption scheme cumbersome. However, if one can present a SHE scheme depth-specifically, the correctness check is enough with depth-wise checks. We relate the homomorphic evaluation algorithms and binary operations on the set $\mathscr{C}$ of ciphertexts, and investigate what makes the depth-specific description is enough for a somewhat homomorphic encryption. We conclude that it is sufficient to have $\mathscr{C}$ with a ring-like structure with respect to the evaluation algorithms for a somewhat homomorphic encryption with relatively small depth. In fact, it is common to have the set of ciphertexts in a fully homomorphic encryption(FHE) scheme as a ring with respect to the evaluation algorithms. It is previously known that one can expand the message size of a SHE as t times larger with the ciphertexts t times larger using the Chinese Remainder Theorem(CRT). In this paper, we rewrite the message expansion method with CRT by using the depth specific description. Moreover, in the case of BGN cryptosystem, we show that one can expand the message size with smaller ciphertexts by using CRT twice. The rate of reduction of the ciphertext size depends on the security level. For example, for BGN cryptosystem using a bilinear group of 2048 bit, one can expand the size of plaintexts as t times larger with $t/3$ times larger ciphertexts. We see that the reducing rate becomes better if the security level increases.

Keywords: homomorphic encryption, somewhat homomorphic encryption scheme, binary operation, Chinese Remainder Theorem, BGN cryptosytem

1 Introduction

Recently many improvements on the construction and implementation of fully homomorphic encryption(FHE) schemes have been proposed since its first concrete introduction by Gentry [4]. An efficient and secure fully homomorphic encryption scheme allows to use untrusted computing resources without risk of disclosure of sensitive data. In particular, one can efficiently evaluate any multivariate polynomial over ciphertexts using FHE. A somewhat homomorphic encryption(SHE) scheme allows to evaluate multivariate polynomials up to a predetermined degree over ciphertexts. A fully homomorphic encryption scheme commonly starts from a somewhat homomorphic encryption scheme and sophisticated techniques such as modulus reductions and key switching are used to make it fully homomorphic. Therefore, the efficiency of the fully homomorphic encryption scheme is closely related to the efficiency of the underlying somewhat homomorphic encryption scheme. We also believe that the efficient somewhat homomorphic encryption scheme is important as itself.

The definition of $(+,\times)$-homomorphic encryption over integers can be illustrated by the following commutative diagrams.

$$\begin{array}{ccc} Z\times Z & \xrightarrow{+} & Z \\ {\scriptstyle \mathrm{Dec}\times\mathrm{Dec}}\uparrow & & \uparrow{\scriptstyle \mathrm{Dec}} \\ \mathscr{C}\times\mathscr{C} & \xrightarrow{\mathrm{Eval}_+} & \mathscr{C} \end{array} \qquad \begin{array}{ccc} Z\times Z & \xrightarrow{\times} & Z \\ {\scriptstyle \mathrm{Dec}\times\mathrm{Dec}}\uparrow & & \uparrow{\scriptstyle \mathrm{Dec}} \\ \mathscr{C}\times\mathscr{C} & \xrightarrow{\mathrm{Eval}_\times} & \mathscr{C} \end{array}$$

We consider the $(+,\times)$-homomorphic encryption over integers with depth k, which means a somewhat homomorphic encryption scheme to evaluate multivariate polynomials of degree up to k over ciphertexts. For example, BGN cryptosystem is a $(+,\times)$-homomorphic

* Corresponding author e-mail: hsl@ewha.ac.kr,seongannym@ewha.ac.kr

encryption with depth two [2]. We call an encryption of a plaintext as a fresh ciphertext. We denote $\mathscr{C}_i$ as the set of ciphertexts of the depth i, that is, ciphertexts evaluated multivariate polynomials of degree i over fresh ciphertexts. The set of fresh ciphertexts is a subset of $\mathscr{C}_1$. Checking the correctness of a $(+,\times)$-homomorphic encryption with depth k is not simple in general, because the type of ciphertexts varies as its encryption depth. A depth specific description of homomorphic encryption of depth k will simplify the correctness-check of the scheme.

Our main contributions are as follows. We suggest when a depth-specific description is enough for a $(+,\times)$-homomorphic encryption with depth k to be correct. We rewrite a previously known message expansion method by using the depth specific description. Moreover, in the case of BGN cryptosystem, we show that one can expand the message size with smaller ciphertexts by using CRT twice.

The homomorphic evaluations on ciphertexts essentially have some integer-arithmetics-like properties because they involve the additions and multiplications on integers. We relate the requirements of homomorphic evaluations on ciphertexts with properties of binary operations on the set of integers Z. Note that the set of integers Z forms a ring with respect to the addition and the multiplication. In particular, rearranging the parentheses or the order of operands in adding many integers(or multiplying many integers) will not change the value. Moreover, the multiplication is distributive over the addition in the integer arithmetics. We reflect these facts to the above commutative diagram of the $(+,\times)$-homomorphic encryption with depth k and present a sufficient condition on the homomorphic evaluation $\mathsf{Eval}_+, \mathsf{Eval}_\times$ for a depth specific description is enough.

Many homomorphic encryption requires plaintexts with small bit sizes and expanding the plaintext properly is another issue in constructing homomorphic encryption schemes. It is known that one can expand plaintexts using Chinese Remainder Theorem(CRT) by [6]. To expand the size of plaintext as t times larger, this method makes the ciphertexts t times larger, too. We rewrite the expanding method by using the depth-specific description which has a simplified correctness check. In the case of BGN cryptosystem, we expand the size as t times larger with smaller ciphertexts using CRT twice. For example, for BGN cryptosystem using a bilinear group of 2048 bit, one can expand the size of plaintext as t times larger with $t/3$ times larger ciphertexts, and this is comparable with the result using the direct usage as in [6]. The rate of reduction of the ciphertext size depends on the security level and we see that the reducing rate becomes better if the security level increases.

The rest of the paper is organized as follows. In Section 2, we review the definition of the $(+,\times)$-homomorphic encryption scheme and binary operations. In Section 3, we present a sufficient condition on the homomorphic evaluations $\mathsf{Eval}_+, \mathsf{Eval}_\times$ for a depth specific description is enough. In Section 4, we rewrite the expanding method using CRT depth-specifically. We show that one can expand the message size with smaller ciphertexts by using CRT twice in the case of BGN cryptosystem. In Section 5, we conclude our paper.

2 Preliminaries

2.1 Definition of $(+,\times)$-Homomorphic Encryption Scheme

The $(+,\times)$-homomorphic encryption scheme allows anyone to add and multiply the plaintext values implicitly while working on ciphertexts only. It consists of five polynomial time algorithms

$$(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Eval}_+, \mathsf{Eval}_\times, \mathsf{Dec}).$$

KeyGen: It outputs a pair (pk, sk) of public key pk and secret key sk on inputting security parameter λ.

Enc: It outputs a ciphertext on inputting the public key pk and message m.

Dec: It recovers the plaintext from a ciphertext on inputting the secret key sk and a ciphertext.

Eval_+: On inputting the public key and ciphertexts C_1 and C_2, it evaluates the addition homomorphically which we denote $\tilde{C} = \mathsf{Eval}_+(pk, (C_1, C_2))$.

$\mathsf{Eval}_\times$: On inputting the public key and ciphertexts C_1 and C_2, it evaluates the multiplication homomorphically which we denote $\tilde{C} = \mathsf{Eval}_\times(pk, (C_1, C_2))$.

We say that a $(+,\times)$-homomorphic encryption scheme is correct if it satisfies the followings, for any valid key pair (pk, sk).

1.For any message M, we have

$$\mathsf{Dec}(sk, \mathsf{Enc}(pk, M)) = M;$$

2.For any ciphertexts C, C', we have

$$\mathsf{Dec}(sk, \mathsf{Eval}_+(pk, (C, C'))) = M + M',$$

$$\mathsf{Dec}(sk, \mathsf{Eval}_\times(pk, (C, C'))) = M \times M',$$

where $\mathsf{Dec}(sk, C) = M$ and $\mathsf{Dec}(sk, C') = M'$

2.2 Binary operations

A binary operation $\oplus$ on a non-empty set A is a well-defined map $\oplus : A \times A \to A$ and we denote $x \oplus y = \oplus(x, y) \in A$. Examples of binary operation on the set of integers include the addition $(+)$, the subtraction $(-)$, the multiplication $(\times)$ and the division $(\div)$. Note that the number of ways of associating d applications of a binary operation for $(m_1, m_2, \ldots, m_{d+1})$ is known as the d-th Catalan Number and it is quantified as $\frac{1}{d+1}\binom{2d}{d}$

and it is asymptotically estimated as $\frac{4^d}{\sqrt{d^3\pi}}$. Therefore, expressing and computing many applications of a binary operation is very complicated in general. A nice property for efficient expression and computation several applications of a binary operation is associativity. We say that a binary operation $\oplus$ on A is associative if it holds $x \oplus (y \oplus z) = (x \oplus y) \oplus z$ for all $x,y,z \in A$. If a binary operation $\oplus$ on A is associative, then rearranging the parentheses in its computation will not change the value. Therefore, one can express $\oplus_{j=1}^{k} c_j$ without specifying the parenthesis in the expression. If a binary operation $\oplus$ is associative and commutative, that is, $x \oplus y = y \oplus x$, then the output of $\oplus_{j=1}^{k} c_j$ is independent to the choice of circuits in the computation. If there are two different binary operations $\oplus$ and $\odot$ on a non-empty set A, we say $\odot$ is distributive over $\oplus$ if it holds $x \odot (y \oplus z) = (x \odot y) \oplus (x \odot z)$ for all $x,y,z \in A$. The distributive law of two binary operations is very crucial in many fast algorithms associated with $\oplus$ and $\odot$.

3 A depth specific description of $(+,\times)$-homomorphic encryption

We consider a $(+,\times)$-homomorphic encryption scheme over integers with the depth k and denote $\mathscr{C}_i$ as the set of ciphertexts of depth i for $i = 1,\ldots,k$. The outputs of Enc belong to $\mathscr{C}_1$. We see that $\mathscr{C} = \cup_{1\le i\le k}\mathscr{C}_i$ is the set of ciphertexts. Then clearly, we have

$$\mathsf{Eval}_{+,i} := \mathsf{Eval}_{+}|_{\mathscr{C}_i\times\mathscr{C}_i} : \mathscr{C}_i \times \mathscr{C}_i \to \mathscr{C}_i$$
$$\mathsf{Eval}_{\times,(i,j)} := \mathsf{Eval}_{\times}|_{\mathscr{C}_i\times\mathscr{C}_j} : \mathscr{C}_i \times \mathscr{C}_j \to \mathscr{C}_{i+j}$$

The depth-specific description of a $(+,\times)$-homomorphic encryption scheme over integers with the depth k consists of

$$(\mathsf{KeyGen}, \mathsf{Enc}, (\mathsf{Eval}_{+,i})_{1\le k}, (\mathsf{Eval}_{\times,(i,j)})_{1\le i,j\le n, i+j\le k}, \mathsf{Dec}).$$

If the depth specific description is enough to define a $(+,\times)$-homomorphic encryption scheme, the correctness check can be done depthwise.

The definition of homomorphic encryption scheme over integers requires the following diagrams are commutative.

$$\begin{array}{ccc} Z\times Z & \xrightarrow{+} & Z \\ {\scriptstyle \mathsf{Dec}\times\mathsf{Dec}}\uparrow & & \uparrow{\scriptstyle \mathsf{Dec}} \\ \mathscr{C}_i\times\mathscr{C}_i & \xrightarrow[\mathsf{Eval}_{+,i}]{} & \mathscr{C}_i \end{array} \qquad \begin{array}{ccc} Z\times Z & \xrightarrow{\times} & Z \\ {\scriptstyle \mathsf{Dec}\times\mathsf{Dec}}\uparrow & & \uparrow{\scriptstyle \mathsf{Dec}} \\ \mathscr{C}_i\times\mathscr{C}_j & \xrightarrow[\mathsf{Eval}_{\times,(i,j)}]{} & \mathscr{C}_{i+j} \end{array}$$

In this section, we relate the evaluation algorithms $\mathsf{Eval}_{+,i}$ and $\mathsf{Eval}_{\times,(i,j)}$ with binary operations over ciphertexts and investigate what makes the depth specific description is enough for $(+,\times)$-homomorphic encryption over integers with the depth k.

3.1 Evaluation algorithms Eval_+ and $\mathsf{Eval}_\times$

We first consider the algorithm $\mathsf{Eval}_{+,i} : \mathscr{C}_i \times \mathscr{C}_i \to \mathscr{C}_i$. Because the encryption homomorphically evaluates polynomially many addition over ciphertexts, we have the following diagram commutes, for any $d' \le d = d(\lambda)$ where $d(\lambda)$ is a polynomial in the security parameter λ.

$$\begin{array}{ccc} Z^{d'} & \xrightarrow{\Sigma_{j=1}^{d'}} & Z \\ {\scriptstyle \mathsf{Dec}^{d'}}\uparrow & & \uparrow{\scriptstyle \mathsf{Dec}} \\ \mathscr{C}_i^{d'} & \xrightarrow[\mathsf{Eval}_{\Sigma_{j=1}^{d'}}]{} & \mathscr{C}_i \end{array}$$

The commutativity of the diagram assumes that two maps $\mathsf{Eval}_{\Sigma_{j=1}^{d'}}$ and Dec are well-defined.

Because the addition in Z is commutative, $\Sigma_{j=1}^{d'}$ is well-defined and the output is independent to the choice of circuits to compute the summation. Therefore, we need the output of $\mathsf{Eval}_{\Sigma_{j=1}^{d'}}$ to be independent (up to decryption) to the choice of the circuit to compute $\Sigma_{j=1}^{d'}$. Here, we say that $\mathsf{Eval}_{\Sigma_{j=1}^{d'}}$ is independent up to decryption if $\mathsf{Dec}(\mathsf{Eval}_{\Sigma_{j=1}^{d'}})$ coincides for any circuit to compute $\mathsf{Eval}_{\Sigma_{j=1}^{d'}}$. A sufficient way to achieve this is that $\mathsf{Eval}_{+,i} : \mathscr{C}_i \times \mathscr{C}_i \to \mathscr{C}_i$ is a commutative and associative binary operation on the set $\mathscr{C}_i$.

Now we consider $\mathsf{Eval}_{\times,(i,j)}$. Similarly as in the case of $\mathsf{Eval}_{+,i}$, the definition of $(+,\times)$-homomorphic encryption scheme with the depth k requires that the following diagram commutes for any $k' \le k$.

$$\begin{array}{ccc} Z^{k'} & \xrightarrow{\Pi_{j=1}^{k'}} & Z \\ {\scriptstyle \mathsf{Dec}^{k'}}\uparrow & & \uparrow{\scriptstyle \mathsf{Dec}} \\ \mathscr{C}_1^{k'} & \xrightarrow[\mathsf{Eval}_{\Pi_{j=1}^{k'}}]{} & \mathscr{C}_{k'} \end{array}$$

Note that $\prod_{j=1}^{k'}$ in Z is well-defined and the output is independent to the choice of circuits to compute the multiplication. A sufficient way to give commutativity of the above diagram for general k is that $\mathscr{C} = \mathscr{C}_i$ for all i and $\mathsf{Eval}_\times : \mathscr{C} \times \mathscr{C} \to \mathscr{C}$ is a commutative and associative binary operation for $\mathscr{C} = \mathscr{C}_i$. Note that the operations of plaintexts are implicit in the homomorphic evaluation process, it is good to have the output of evaluation algorithm is decrypted independently to a specific circuit of underlying computations of plaintexts. For the case $k = 2$, it is enough that $\mathsf{Eval}_\times : \mathscr{C}_1 \times \mathscr{C}_1 \to \mathscr{C}_2$ is efficiently computable well-defined map and commutative up to decryption. A sufficient way to give commutativity of the above diagram for general k is that $\mathscr{C} = \mathscr{C}_i$ for all i and $\mathsf{Eval}_\times : \mathscr{C} \times \mathscr{C} \to \mathscr{C}$ is a commutative and associative binary operation for $\mathscr{C} = \mathscr{C}_i$.

Now we consider $\mathsf{Eval}_{+,i}$ and $\mathsf{Eval}_{\times,(i,j)}$ simultaneously. We assume that operations $\mathsf{Eval}_{\sum_{j=1}^{d'}}$ independent to the choice of the circuit to compute and $\mathsf{Eval}_{\prod_{j=1}^{k'}}$ are independent to parenthesizing the input. Now we consider $c \in \mathscr{C}_i$ and $c' \in \mathscr{C}_j$ with $i > j$. Then $\mathsf{Eval}_+(c,c')$ can be naturally defined as follows using homomorphic property, that is,

$$\mathsf{Eval}_+(c,c') = \mathsf{Eval}_{+,i}(c, \mathsf{Eval}_{\times,(i-j,j)}(\tilde{c},c')),$$

where $\tilde{c} = \mathsf{Eval}_{\prod_{1\le\ell\le i-j}}(c_1,\ldots,c_\ell,\ldots,c_{i-j})$ with $c_\ell = \mathsf{Enc}(1)$ for $1 \le \ell \le i-j$. Therefore, the depth-specific algorithm $\mathsf{Eval}_{+,i} : \mathscr{C}_i \times \mathscr{C}_i \to \mathscr{C}_i$ naturally extends to the full evaluation algorithm $\mathsf{Eval}_+ : \mathscr{C} \times \mathscr{C} \to \mathscr{C}$. When we consider both of the addition and multiplication together, we should have the following diagram commutes.

$$\begin{array}{ccc} Z \times Z \times Z & \xrightarrow{m_1\times(m_2+m_3)} & Z \\ {\scriptstyle \mathsf{Dec}^3}\uparrow & & \uparrow{\scriptstyle \mathsf{Dec}} \\ \mathscr{C}_i \times \mathscr{C}_j \times \mathscr{C}_\ell & \xrightarrow{\mathsf{Eval}_{\times,(i,\ell')}(c_1,\mathsf{Eval}_+(c_2,c_3))} & \mathscr{C}_{i+\max(j,\ell')} \end{array}$$

where $\ell' = \max(j,\ell)$. Again, the commutativity of the above diagram requires that $\mathsf{Eval}_{\times,(i,j)}$ is distributive over Eval_+ where it is defined.

For a large k, using a commutative ring $(\mathscr{C}, \mathsf{Eval}_+, \mathsf{Eval}_\times)$ could be a solution. In that case, making the decryption algorithm Dec as well-defined in the diagram is the main focus in the design of $(+,\times)$ homomorphic encryption scheme. In fact, all the known fully homomorphic encryption scheme are defined over a commutative ring and the core research issues in the construction of (fully or somewhat) homomorphic encryption schemes is how to make Dec as an efficient well-defined map in the commutative diagram above.

If we consider small k, there are solutions without any ring structure such as the BGN cryptosystem and the GHV cryptosystem. In the BGN cryptosystem, it is assumed that the size of the plaintext is small in order to make Dec as a well-defined and efficient map.

3.2 A Sufficient Condition for the Depth Specific Description is Enough

We start from a public key encryption scheme $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ with the depth specific homomorphic evaluations $\mathsf{Eval}_{+,i} : \mathscr{C}_i \times \mathscr{C}_i \to \mathscr{C}_i$ and $\mathsf{Eval}_{\times,(i,j)} : \mathscr{C}_i \times \mathscr{C}_j \to \mathscr{C}_{i+j}$. The following Theorem states a sufficient condition to extend $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ to an efficient $(+,\times)$ homomorphic encryption scheme of depth k with the set of ciphertexts $\mathscr{C} = \cup_{j=1}^k \mathscr{C}_j$.

Theorem 1. *Suppose that we have an encryption scheme $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ where the following diagrams commute*

$$\begin{array}{ccc} Z \times Z & \xrightarrow{+} & Z \\ {\scriptstyle \mathsf{Dec}\times\mathsf{Dec}}\uparrow & & \uparrow{\scriptstyle \mathsf{Dec}} \\ \mathscr{C}_i \times \mathscr{C}_i & \xrightarrow{\mathsf{Eval}_{+,i}} & \mathscr{C}_i \end{array} \qquad \begin{array}{ccc} Z \times Z & \xrightarrow{\times} & Z \\ {\scriptstyle \mathsf{Dec}\times\mathsf{Dec}}\uparrow & & \uparrow{\scriptstyle \mathsf{Dec}} \\ \mathscr{C}_i \times \mathscr{C}_j & \xrightarrow{\mathsf{Eval}_{\times,(i,j)}} & \mathscr{C}_{i+j} \end{array}$$

and

- *$\mathsf{Eval}_{\times,(i,j)}$ are efficiently computable and the outputs of $\mathsf{Eval}_{\prod_{j=1}^{k'}}$ are independent to parenthesizing the input for any $k' \le k$;*
- *$\mathsf{Eval}_{+,i}$ are associative binary operations for all i;*
- *$\mathsf{Eval}_{\times,(i,j)}$ is distributive over $\mathsf{Eval}_{+,i+j}$ up to decryption, i.e., for $c_1 \in \mathscr{C}_i$ and $c_2, c_3 \in \mathscr{C}_j$ with $i+j \le k$, we have*

$$\begin{aligned} &\mathsf{Dec}(\mathsf{Eval}_{\times,(i,j)}(c_1, \mathsf{Eval}_{+,j}(c_2,c_3))) \\ &= \mathsf{Dec}(\mathsf{Eval}_{+,i+j}(\mathsf{Eval}_{\times,(i,j)}(c_1,c_2), \mathsf{Eval}_{\times,(i,j)}(c_1,c_3))). \end{aligned}$$

Then $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Eval}_+, \mathsf{Eval}_\times, \mathsf{Dec})$ defines an efficient $(+,\times)$ homomorphic encryption of depth k with the set of ciphertexts $\mathscr{C} = \cup_{j=0}^k \mathscr{C}_j$, where $\mathsf{Eval}_+ : \mathscr{C} \times \mathscr{C} \to \mathscr{C}$ is defined as follows: for $c \in \mathscr{C}_i$ and $c' \in \mathscr{C}_j$ assuming $j \le i \le k$, define

$$\mathsf{Eval}_+(c,c') = \mathsf{Eval}_{+,i}(\tilde{c},c')),$$

where $\tilde{c} = \mathsf{Eval}_{\times,(i-j,j)}(\mathsf{Eval}_{\prod_{1\le\ell\le i-j}}(c_1,\ldots,c_{i-j})$ with $c_\ell = \mathsf{Enc}(1)$ for $1 \le \ell \le i-j$.

Proof. To show that the encryption scheme $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Eval}_+, \mathsf{Eval}_\times, \mathsf{Dec})$ defines an efficient $(+,\times)$ homomorphic encryption scheme of depth k, it is enough to show that, for all $c, c', c'' \in \mathscr{C}$ and $c_1 \in \mathscr{C}_i$ and $c_2 \in \mathscr{C}_j, c_3 \in \mathscr{C}_\ell$ with $(i + \max(j,\ell)) \le k$,

$$\begin{aligned} &\mathsf{Dec}(\mathsf{Eval}_+(c,c')) = \mathsf{Dec}(\mathsf{Eval}_+(c',c)), \\ &\mathsf{Dec}(\mathsf{Eval}_+(c, \mathsf{Eval}_+(c',c''))) \\ &= \mathsf{Dec}(\mathsf{Eval}_+(\mathsf{Eval}_+(c,c'),c'')), \\ &\mathsf{Dec}(\mathsf{Eval}_\times(c_1, \mathsf{Eval}_+(c_2,c_3))) \\ &= \mathsf{Dec}(\mathsf{Eval}_+(\mathsf{Eval}_\times(c_1,c_2), \mathsf{Eval}_\times(c_1,c_3))). \end{aligned}$$

All these follows from the fact that for any $c \in \mathscr{C}_j$ and $c' \in \mathscr{C}_\ell$, we can assume that $c, c' \in \mathscr{C}_{\max(j,\ell)}$ by multiplying 1 homomorphically. Note that the associativity of $\mathsf{Eval}_+ : \mathscr{C}_i \times \mathscr{C}_i \to \mathscr{C}_i$ for ever i assures that the associativity of Eval_+ and $\mathsf{Eval}_{\times,(i,j)}$ is distributive over $\mathsf{Eval}_{+,i+j}$ for every i, j with $i + j \le k$ assures that $\mathsf{Eval}_\times$ is distributive over Eval_+ where it is defined.

3.3 Examples

Now we review some examples of the $(+,\times)$-homomorphic encryption schemes of depth two in

the framework of Theorem 1. We show that the BGN cryptosystem satisfies all the conditions for depth specific encryption is enough [2]. On the otherhand, we show that the GHV cryptosystem has a subtle difference from the BGN cryptosystem because $\mathsf{Eval}_\times : \mathscr{C}_1 \times \mathscr{C}_1 \to \mathscr{C}_2$ is not commutative up to decryption.

3.3.1 BGN cryptosystem

The BGN cryptosystem is the first $(+,\times)$ homomorphic encryption scheme with depth two [2]. It is defined over a bilinear map $e : G \times G \to G_T$ with cyclic groups G, G_T of order $N = pq$, and the depth specific description of evaluation algorithms of the BGN cryptosystem is given as follows.

1. $\mathscr{C}_1$: the group G of generator g;
2. $\mathscr{C}_2$: the group G_T;
3. $\mathsf{Eval}_{+,1} : \mathscr{C}_1 \times \mathscr{C}_1 \to \mathscr{C}_1$ is the multiplication in G, which is associative and commutative;
4. $\mathsf{Eval}_{+,2} : \mathscr{C}_2 \times \mathscr{C}_2 \to \mathscr{C}_2$ is the multiplication in G_T, which is associative and commutative;
5. $\mathsf{Eval}_\times$ is the pairing $e : G \times G \to G_T$, which is associative and commutative.

It is clear that the pairing $\mathsf{Eval}_\times = e$ is efficiently computable. Moreover, we have $e(c,c') = e(c',c)$ and this implies that one can evaluate independently to the order in the multiplication of integers(plaintexts). It is also distributive over Eval_+, because

$$\begin{aligned}&\mathsf{Eval}_\times(c_1, \mathsf{Eval}_{+,1}(c_2,c_3))\\ &= e(c_1, c_2 \cdot c_3)\\ &= e(c_1,c_2)\cdot e(c_1,c_3)\\ &= \mathsf{Eval}_{+,2}(\mathsf{Eval}_\times(c_1,c_2), \mathsf{Eval}_\times(c_1,c_3))\end{aligned}$$

For security reason, one can add a randomness in the evaluation algorithms. However, one can easily see that the evaluations $\mathsf{Eval}_{+,1}, \mathsf{Eval}_{+,2}$ and $\mathsf{Eval}_\times$ are associative and commutative up to decryption. In that case, $\mathsf{Eval}_\times$ is distributive over Eval_+ up to decryption.

In order to make the algorithm Dec correctly decrypts, the message size is restricted so small as the discrete logarithm can be efficiently solved in the setting the message as the exponent.

3.3.2 GHV cryptosystem

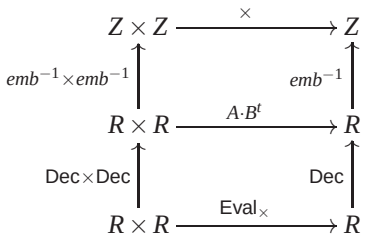
In the GHV cryptosystem [5], the authors consider the set R of $k \times k$ matrices over a finite field and construct the cryptosystem in the setting

1. $\mathscr{C}_0 = \mathscr{C}_1 = R$: the ring of $k \times k$ matrices;
2. $\mathsf{Eval}_+ = \mathsf{Eval}_{+,1} = \mathsf{Eval}_{+,2} : R \times R \to R$ as the sum of matrices in R;
3. $\mathsf{Eval}_\times : R \times R \to R$ defined as $\mathsf{Eval}_\times(c_1,c_2) = c_1 \cdot c_2^T \in R$, where c_2^T is the transpose of the matrix c_2.

It is clear that Eval_+ is an associative binary operation and $\mathsf{Eval}_\times$ is distributive over Eval_+, because

$$\begin{aligned}&\mathsf{Eval}_\times(c_1, \mathsf{Eval}_+(c_2,c_3))\\ &= c_1 \cdot (c_2 + c_3)^T\\ &= c_1 \times c_2^T + c_1 \times c_3^T\\ &= \mathsf{Eval}_+(\mathsf{Eval}_\times(c_1,c_2), \mathsf{Eval}_\times(c_1,c_3))\end{aligned}$$

The security of GHV cryptosystem is based on the LWE problem, which requires the size of underlying matrices to be large. Moreover, it is a challenging problem to find an efficient embedding *emb* of integers to matrices so that the related diagram commutes. We also note that $\mathsf{Eval}_\times$ is not commutative even up to decryption, therefore the order of the operation $\mathsf{Eval}_\times$ should be consistent to the order of the multiplication in Z. Recall that it is good to have the outputs of the evaluation algorithms are decrypted independently to a specific circuit of underlying computations of plaintexts. Therefore, this feature of GHV cryptosystem is undesirable in homomorphic encryption scheme.



4 Application to Message Expansion of SHE

Suppose that we have a $(+,\times)$ homomorphic encryption scheme HE_ω of depth k whose message space is integers less than ω-bits:

$$(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Eval}_+, \mathsf{Eval}_\times, \mathsf{Dec}).$$

We denote algorithms of HE_ω using the notations in Theorem 1 as follows;

$$\begin{aligned}&\mathsf{Enc} : Z \to \mathscr{C}_1\\ &\mathsf{Eval}_\times : \mathscr{C}_i \times \mathscr{C}_j \to \mathscr{C}_{i+j}\\ &\mathsf{Dec} : \mathscr{C} \to Z, \text{ where } \mathscr{C} = \cup_{i=1}^k \mathscr{C}_i\\ &\mathsf{Eval}_+ : \mathscr{C} \times \mathscr{C} \to \mathscr{C}, \text{where } \mathsf{Eval}_+ : \mathscr{C}_i \times \mathscr{C}_i \to \mathscr{C}_i\end{aligned}$$

4.1 Expanding the plaintexts by using CRT(Chinese Remainder Theorem)

It is known that one can expand plaintexts of somewhat homomorphic encryptions using Chinese Remainder Theorem(CRT) by [6]. To expand the size of plaintext as t times larger, this method makes the ciphertexts t times

larger, too. We rewrite the expanding method by using the depth-specific description which has a simplified correctness check.

Note that $(Z^t,+,\times)$ is a commutative ring with unity, where $+$ and $\times$ are addition and multiplication componentwise, that is,

$$(a_1,\dots,a_t)+(a'_1,\dots,a'_t)=(a_1+a'_1,\dots,a_t+a'_t)\in Z^t$$
$$(a_1,\dots,a_t)\times(a'_1,\dots,a'_t)=(a_1\times a'_1,\dots,a_t\times a'_t)\in Z^t.$$

Now we construct a $(+,\times)$ homomorphic encryption scheme

$$HE^t=(\mathsf{KeyGen}^t,\mathsf{Enc}^t,\mathsf{Eval}^t_+,\mathsf{Eval}^t_\times,\mathsf{Dec}^t)$$

using the product spaces $Z^t,\mathscr{C}^t_i,\mathscr{C}^t$ and applying each algorithm componentwise. We omit the index i of the evaluation algorithms because it is clear from their domains.

$\mathsf{KeyGen}^t:(pk,sk)\leftarrow\mathsf{KeyGen}(1^\lambda)$
$\mathsf{Enc}^t_{pk}:Z^t\to\mathscr{C}^t_0$ defined by

$$\mathsf{Enc}^t_{pk}(m_1,\dots,m_t)=(\mathsf{Enc}_{pk}(m_1),\dots,\mathsf{Enc}_{pk}(m_t))$$

$\mathsf{Eval}^t_{pk,\times}:\mathscr{C}^t_i\times\mathscr{C}^t_j\to\mathscr{C}^t_{i+j}$ defined by

$$\mathsf{Eval}^t_{pk,\times}((c_1,\dots,c_t),(c'_1,\dots,c'_t))$$
$$=(\mathsf{Eval}_{pk,\times}(c_1,c'_1),\dots,\mathsf{Eval}_{pk,\times}(c_t,c'_t))$$

$\mathsf{Dec}^t_{sk}:\mathscr{C}^t\to Z^t$ defined by

$$\mathsf{Dec}^t_{sk}(c_1,\dots,c_t)=(\mathsf{Dec}_{sk}(c_1),\dots,\mathsf{Dec}_{sk}(c_t))$$

$\mathsf{Eval}^t_{pk,+}:\mathscr{C}^t_i\times\mathscr{C}^t_i\to\mathscr{C}^t_i$ defined by

$$\mathsf{Eval}^t_{pk,+}((c_1,\dots,c_t),(c'_1,\dots,c'_t))$$
$$=(\mathsf{Eval}_{pk,+}(c_1,c'_1),\dots,\mathsf{Eval}_{pk,+}(c_t,c'_t))$$

This introduces the following commutative diagrams.

$$\begin{array}{ccc} Z^t\times Z^t & \xrightarrow{+} & Z^t \\ {\scriptstyle \mathsf{Dec}^t\times\mathsf{Dec}^t}\uparrow & & \uparrow{\scriptstyle \mathsf{Dec}^t} \\ \mathscr{C}^t_i\times\mathscr{C}^t_i & \xrightarrow{\mathsf{Eval}^t_+} & \mathscr{C}^t_i \end{array} \qquad \begin{array}{ccc} Z^t\times Z^t & \xrightarrow{\times} & Z \\ {\scriptstyle \mathsf{Dec}^t\times\mathsf{Dec}^t}\uparrow & & \uparrow{\scriptstyle \mathsf{Dec}^t} \\ \mathscr{C}^t_i\times\mathscr{C}^t_j & \xrightarrow{\mathsf{Eval}_\times} & \mathscr{C}^t_{i+j} \end{array}$$

It is clear to see that $(\mathsf{KeyGen}^t,\mathsf{Enc}^t,\mathsf{Eval}^t_+,\mathsf{Eval}^t_\times,\mathsf{Dec}^t)$ satisfies the depth-specific hypothesis of Theorem 1. Therefore $(\mathsf{KeyGen}^t,\mathsf{Enc}^t,\mathsf{Eval}^t_+,\mathsf{Eval}^t_\times,\mathsf{Dec}^t)$ defines a $(+,\times)$ homomorphic encryption scheme with depth k.

Now we show that HE^t can be used to expand the plaintexts of HE_ω as t times larger. For $n=q_1q_2\cdots q_t$ with distinct prime numbers q_i, CRT introduces an ring isomorphism $CRT:Z_{q_1}\times\cdots\times Z_{q_t}\to Z_n$ as follows.

$$CRT(m_1,\dots,m_t)$$
$$=\left(\sum_{i=1}^{t}m_i\cdot\frac{n}{q_i}\cdot\left(\left(\frac{n}{q_i}\right)^{-1}\bmod q_i\right)\right)\bmod n$$
$$CRT^{-1}(m)=(m\bmod q_1,\dots,m\bmod q_t)$$

The primes q_i are chosen small enough for $\mathsf{Enc}:Z_{q_i}\to\mathscr{C}_0$ to be correctly decrypted in the original homomorphic encryption scheme HE_ω. We set $X=Z_{q_1}\times\cdots\times Z_{q_t}$ which can be considered as a subset of Z^t.

$$\begin{array}{ccc} Z_n\times Z_n & \xrightarrow{+\bmod n} & Z_n \\ {\scriptstyle CRT\times CRT}\uparrow & & \uparrow{\scriptstyle CRT} \\ X\times X & \xrightarrow{+} & X \\ {\scriptstyle \mathsf{Dec}^t\times\mathsf{Dec}^t}\uparrow & & \uparrow{\scriptstyle \mathsf{Dec}^t} \\ \mathscr{C}^t_i\times\mathscr{C}^t_i & \xrightarrow{\mathsf{Eval}^t_{+,i}} & \mathscr{C}^t_i \end{array} \qquad \begin{array}{ccc} Z_n\times Z_n & \xrightarrow{\cdot\bmod n} & Z_n \\ {\scriptstyle CRT\times CRT}\uparrow & & \uparrow{\scriptstyle CRT} \\ X\times X & \xrightarrow{\times} & X \\ {\scriptstyle \mathsf{Dec}^t\times\mathsf{Dec}^t}\uparrow & & \uparrow{\scriptstyle \mathsf{Dec}^t} \\ \mathscr{C}^t_i\times\mathscr{C}^t_j & \xrightarrow{\mathsf{Eval}_{\times,(i,j)}} & \mathscr{C}^t_{i+j} \end{array}$$

We consider $(\mathsf{keygen},\mathsf{enc},\mathsf{eval}_+,\mathsf{eval}_\times,\mathsf{dec})$ defined by

$\mathsf{keygen}=\mathsf{KeyGen}^t$
$\mathsf{enc}(m)=\mathsf{Enc}^t(CRT^{-1}(m))$
$\mathsf{dec}(c)=CRT(\mathsf{Dec}^t(c))$
$\mathsf{eval}_+(c,c')=\mathsf{Eval}^t_+(c,c')$
$\mathsf{eval}_\times(c,c')=\mathsf{Eval}^t_\times(c,c')$.

Therefore, $(\mathsf{keygen},\mathsf{enc},\mathsf{eval}_+,\mathsf{eval}_\times,\mathsf{dec})$ defines a homomorphic encryption over integers of the size t times larger than the original homomorphic encryption scheme HE_ω.

4.2 Application to BGN cryptosystem

Now we show that we can expand the size as t times larger with smaller ciphertexts using CRT twice, in the case of the BGN cryptosytem. The rate of reduction of the ciphertext size depends on the security level. As an example, for BGN cryptosystem using a bilinear group of 2048 bit, we show that one can expand the size of plaintext as t times larger with $t/3$ times larger ciphertexts, and this is comparable with the result using the direct usage as in [6].

4.2.1 A Modified BGN Cryptosystem using Multiprimes

The idea of expanding the plaintext size is to modify the type of the composite order of the bilinear group. The original BGN cryptosystem uses the bilinear group of composite order P_1P_2, where its security relies on the hardness of factorization of P_1P_2. In this section, we modify the BGN cryptosystem using bilinear groups of order N where N is a product of $\tau+1$ distinct prime numbers $P_1,P_2,\dots,P_\tau,P_{\tau+1}$ and it is computationally hard to factor N. Our modification expands the bit size of plaintexts of BGN cryptosystem as τ times larger while sustaining the size of order of bilinear group but in a different form.

Suppose that we have a bilinear map $e:G\times G\to G_T$ with $|G|=|G_T|=N=P_1\cdots P_{\tau+1}$ and assume that it is

hard to factor N. We also suppose that g_0 is a generator of G and $g_i = g_0^{\frac{N}{P_i P_{\tau+1}}} \in G$ for $i = 1,\ldots,\tau$ and $h = g_0^{\frac{N}{P_{\tau+1}}} \in G$. Note that we have $g_i^{\frac{N}{p_j}} = g_0^{\frac{N^2}{P_i P_j P_{\tau+1}}} = 1$ and $e(g_i,g_j)$ is of order $P_{\tau+1}$ if $i \neq j$. We also note that $e(g_i,g_i)$ is of order $P_i P_{\tau+1}$. Therefore, we see that

$$e(g_1^{\alpha_1}\cdots g_\tau^{\alpha_\tau} h^r, g_1^{\alpha'_1}\cdots g_\tau^{\alpha'_\tau} h^{r'}) = \left(\prod_{i=1}^{\tau} e(g_i,g_i)^{\alpha_i\cdot\alpha'_i}\right)\cdot B^{\tilde{r}},$$

where the order of $B \in G_T$ is $P_{\tau+1}$. Now we describe our modified BGN cryptosystem. From our Theorem 1, we only need a depth specific description by setting $\mathscr{C}_1 = G$ and $\mathscr{C}_2 = G_T$. The public key of our modification is $pk = (e : G\times G \to G_T, N, g_0, g_1, \ldots, g_\tau, h)$ and the private key is $sk = (P_1, P_2, \ldots, P_\tau)$. We define $\mathsf{Enc} : Z_n \to G$ where $n = q_1 q_2 \cdots q_\tau$ in the following manner:

$$\mathsf{Enc}(m) = g_1^{m_1}\cdots g_\tau^{m_\tau} h^r \in G, \text{ where } m_i = m \bmod q_i.$$

For the notational convenience, we omit the re-randomization in the homomorphic evaluation.

$\mathsf{Eval}_\times(C,C') = e(C,C')$ for $C,C' \in G$,
$\mathsf{Eval}_{+,1}(C,C') = C\cdot C' \in G$,
$\mathsf{Eval}_{+,2}(C,C') = C\cdot C' \in G_T$.

For given a ciphertext $C \in \mathscr{C}_1 \cup \mathscr{C}_2 = G \cup G_T$, we decrypt the ciphertext C as follows:

(Case 1) : $C \in G$

1.Compute $C_i = C^{\frac{N}{P_i}} = \left(g_i^{m_i}\right)^{\frac{N}{P_i}}$ for all $i = 1,\ldots,\tau$.
2.Compute $m_i = \log_{\left(g_i^{\frac{N}{P_i}}\right)} C_i$ for $i = 1,\ldots,\tau$.
3.Compute $m = CRT_{q_1,\ldots,q_\tau}(m_1,\ldots m_\tau)$.

(Case 2) : $C \in G_T$

1.Compute $C_i = C^{\frac{N}{P_i}} = \left(e(g_i,g_i)^{m_i}\right)^{\frac{N}{P_i}}$ for all $i = 1,\ldots,\tau$.
2.Compute $m_i = \log_{\left(e(g_i,g_i)^{\frac{N}{P_i}}\right)} C_i$ for $i = 1,\ldots,\tau$.
3.Compute $m = CRT_{q_1,\ldots,q_\tau}(m_1,\ldots m_\tau)$.

Now we show that our modified BGN cryptosystem is a correct $(+,\times)$ homomorphic encryption scheme of depth two. For any ciphertext $C \in \mathscr{C}_1 \cup \mathscr{C}_2 = G \cup G_T$, we see the following holds.

(Case 1) : $C \in G$

$$C = g_1^{m_1}\cdots g_\tau^{m_\tau} h^r.$$

(Case 2) : $C \in G_T$

$$C = e(g_1,g_1)^{m_1}\cdots e(g_\tau,g_\tau)^{m_\tau} B^r,$$

where $B \in G_T$ is of order $P_{\tau+1}$.

Therefore, it correctly decrypts depth wise. As in the original BGN cryptosystem, we see that Eval_+ is associative binary operation and $\mathsf{Eval}_\times$ is distributive over Eval_+. We also see that the following diagrams commute.

$$\begin{array}{ccc} Z\times Z & \xrightarrow{+} & Z \\ {\scriptstyle \mathsf{Dec}\times\mathsf{Dec}}\uparrow & & \uparrow{\scriptstyle \mathsf{Dec}} \\ G\times G & \xrightarrow{\mathsf{Eval}_+} & G \end{array} \qquad \begin{array}{ccc} Z\times Z & \xrightarrow{\times} & Z \\ {\scriptstyle \mathsf{Dec}\times\mathsf{Dec}}\uparrow & & \uparrow{\scriptstyle \mathsf{Dec}} \\ G_T\times G_T & \xrightarrow{\mathsf{Eval}_+} & G_T \end{array}$$

$$\begin{array}{ccc} Z\times Z & \xrightarrow{\times} & Z \\ {\scriptstyle \mathsf{Dec}\times\mathsf{Dec}}\uparrow & & \uparrow{\scriptstyle \mathsf{Dec}} \\ G\times G & \xrightarrow{\mathsf{Eval}_\times} & G_T \end{array}$$

Therefore, our construction Enc of homomorphic encryption scheme satisfies all the depth-specific condition in Theorem 1 and we conclude that it is a $(+,\times)$ homomorphic encryption scheme of depth two which encrypts integers $\leq q_1 q_2 \cdots q_\tau$ under the assumption that the original BGN cryptosystem encrypts integers smaller than q_i for all $i = 1,\ldots,\tau$.

Definition 1.*(Subset Membership Problem) Consider Z_n for $n = \prod_{i=1}^k p_i^{e_i}$. Let $\mathscr{C}, \mathscr{V}$ be subsets of Z_n such that $\mathscr{V} \subset \mathscr{C}$. The subset membership problem defined by $(\mathscr{C},\mathscr{V})$ is the problem of deciding whether $x \in \mathscr{V}$ for a given $x \leftarrow \mathscr{C}$.*

Definition 2.*(Subgroup Decision Problem) Consider Z_n for $n = \prod_{i=1}^k p_i^{e_i}$ and let G be a cyclic group of the order n and G_i be a subgroup of G of order $p_i^{e_i}$. The subgroup decision problem is the subset membership problem $(\mathscr{C},\mathscr{V})$ with $\mathscr{C} = G, \mathscr{V} = G_i$.*

It was proven that the subgroup decision problem on G is computationally equivalent to the factorization problem of $n = |G|$ in the generic model [9,10]. We call the subgroup decision assumption as the assumption that the subgroup decision problem is hard.

As in the BGN cryptosystem, we see our modified BGN cryptosystem is IND-CPA secure is based on the subgroup decision assumption with auxiliary inputs. More precisely, we consider the subgroup decision problem in the cyclic groups G (and G_T) with $|G| = |G_T| = N = P_1 P_2 \cdots P_{\tau+1}$ with the subgroups of G (and G_T) of order $P_{\tau+1}$ with an auxiliary input $(g_1, g_2, \ldots, g_\tau) \in G^\tau$ (and $(e(g_i,g_j)) \in G_T^{\tau^2}$). Because we consider small τ's, the auxiliary input doesn't make the subgroup decision easier [3]. Therefore, we select the largest τ where integer factorization of $N = P_1 P_2 \cdots P_{\tau+1}$ is infeasible for a fixed bit size of N. The state of the art for the integer factorization is given in many literatures such as [1,7,11,12]. The current factorization technique suggests that one can use N as a product of four prime numbers of the same sizes for $\log_2 N = 2048$, i.e, $N = P_1 P_2 P_3 P_4$ and this implies that one can use $\tau = 3$ for N of 2048 bits. For N with 4096 bits, one can use N of

five prime numbers of the same size and this implies that $\tau = 4$ for $\log_2 N = 4096$. We denote the BGN cryptosystem over bilinear group of order α-bits as BGN_α. By using multiprimes in the original BGN cryptosystem, we can encrypt message of three times larger (four times) larger than the original BGN_{2048} (BGN_{4096})cryptosystem without increasing the size of ciphertexts.

4.2.2 A Method of using CRT twice with an Example for 2048 bit Security Level

For further expansion of the message space, one can apply CRT method to the modified BGN cryptosystem. We use the modified BGN cryptosystem in multiprime with $\tau+1$ distinct primes and apply CRT expansion on it with t/τ dimensional product space. In this way, one can expand the message size t times larger with t/τ times larger ciphertexts. Note that the selection of τ depends on the hardness of factoring of N. The value τ increases as the security level of the system increases, which means that the reduction rate of ciphertext size is better as the security level increases.

For example, if one wants to enlarge 12 times larger than the original BGN cryptosystem of RSA-2048 security level, then one can do as follows which is given in depth specific description. Suppose that the original BGN cryptosystem encrypts integers of ω bits.

KeyGen:

1.Generate distinct prime number P_i of 512 bits and set $N = P_1P_2P_3P_4$.

2.Generate bilinear group G of order N with a bilinear map $e : G \times G \to G_T$ and set $G =< g >, h = g^{\frac{N}{P_4}}$.

3.Generate distinct prime numbers q_{ij} of ω bits and set $n = n_1n_2n_3n_4$ where $n_i = q_{i1}q_{i2}q_{i3}$ for $i = 1,2,3,4$ and $j = 1,2,3$.

4.Output

$sk = (P_1, P_2, P_3)$,

$pk = (N, e, g, g_1, g_2, g_3, h, n_1, n_2, n_3, n_4)$,

where $g_1 = g^{\frac{N}{P_1P_4}}, g_2 = g^{\frac{N}{P_2P_4}}, g_3 = g^{\frac{N}{P_3P_4}}$.

Enc: For a given message m of 12ω bits, the encryption of m is computed as follows.

1.For $i = 1,2,3,4$, compute

$$m_i = m \pmod{n_i} \text{ with } m_{ij} = m_i \pmod{q_{ij}},$$

where q_{i1}, q_{i2} and q_{i3} are prime factors of n_i.

2.For $i = 1,2,3,4$, compute

$$c_i = \mathsf{Enc}(m_i) = g_1^{m_{i1}} g_2^{m_{i2}} g_3^{m_{i3}} h^{r_i} \in G.$$

3.Output

$$c = (\mathsf{Enc}(m_1), \mathsf{Enc}(m_2), \mathsf{Enc}(m_3), \mathsf{Enc}(m_4)).$$

Dec: For given ciphertext $c = (c_1, c_2, c_3, c_4)$, the decryption of c is proceeded as follows. Assume that $c \in G^4$.

1.For $j = 1,2,3$ and $i = 1,2,3,4$, compute

$$m_{ij} = \log_{\left(g_j^{\frac{N}{P_j}}\right)} (c_i)^{\frac{N}{P_j}}.$$

2.For $i = 1,2,3,4$, compute

$$m_i = CRT_{q_{i1},q_{i2},q_{i3}}(m_{i1}, m_{i2}, m_{i3}).$$

3.Output

$$m = CRT_{n_1,n_2,n_3,n_4}(m_1, m_2, m_3, m_4).$$

If $c \in G_T^4$, then we proceed with

$$m_{ij} = \log_{\left(e(g_j,g_j)^{\frac{N}{P_j}}\right)} (c_i)^{\frac{N}{P_j}}.$$

For given ciphertexts $c = (c_1, \ldots, c_4), c' = (c'_1, \ldots, c'_4)$, the homomorphic evaluation algorithms are proceeded as follows.

For $c, c' \in G^4$,

$\mathsf{Eval}_\times(c, c') = (e(c_1, c'_1), \ldots, e(c_4, c'_4)) \in G_T^4$,

$\mathsf{Eval}_{+,1}(c, c') = (c_1 \cdot c'_1, \ldots, c_4 \cdot c'_4) \in G^4$.

For $c, c' \in G_T^4$,

$\mathsf{Eval}_{+,2}(c, c') = (c_1 \cdot c'_1, \ldots, c_4 \cdot c'_4) \in G_T^4$.

We see that the direct application of CRT as in [6] has ciphertexts in $(G')^{12} \cup (G'_T)^{12}$, where $|G'| = |G'_T| = N'$ with $\log_2 N = \log_2 N'$ while we have ciphertexts in $(G)^4 \cup (G_T)^4$ using CRT twice.

5 Conclusion

In this paper, we investigate when a depth-specific description is enough for SHE schemes over integers. If a SHE is well defined using depth specific description only, then the correctness check is simply depth-wise check. We relate the homomorphic evaluation algorithms and binary operations on the set of ciphertexts and we show that the evaluation algorithms should be associative and binary (up to decryption) binary operations (or maps) and $\mathsf{Eval}_\times$ should be distributive (up to decryption) over Eval_+ in order to a depth-specific description is enough for a SHE. It is known that one can expand plaintexts using Chinese Remainder Theorem(CRT) by [6]. To expand the size of plaintext as t times larger, this method makes the ciphertexts t times larger, too. We rewrite the expanding method by using the depth-specific description which has a simplified correctness check. In the case of BGN cryptosystem, we modify the BGN cryptosystem in

multiprime setting and we show that one can expand the message size t times larger with t/τ times larger ciphertexts by using CRT twice, where $\tau+1$ is the number of prime factors of the order N of the underlying bilinear group. Note that the selection of τ depends on the hardness of factoring of N. The value τ increases as the security level of the system increases, which means that the reduction rate of ciphertext size is better as the security level increases.

Acknowledgement

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT and Future Planning(Grant Number: 2012R1A2A1A03006706).

References

[1] K. Aoki, J. Franke, A. Lenstra, E. Thome, J. Bos, P. Gaudry, A. Kruppa, P. Montgomery, D. Osvik, H. Riele, A. Timofeev and P. Zimmermann. "Factorization of a 768-bit RSA modulus", http://eprint.iacr.org/2010/006.pdf, (2010).
[2] D. Boneh, E.-J.Goh, and K. Nissim, "Evaluating 2-DNF formulas on ciphertexts." In Theory of Cryptography-TCC 2005, Springer Verlag, LNCS, **3378**, 325-341 (2005).
[3] J. Cheon, "Discrete logarithm problems with auxiliary inputs", Journal of Cryptology, **23**, 457-476 (2010).
[4] C. Gentry, "Fully homomorphic encryption using ideal lattices", Proceedings of the 41st annual ACM symposium on Theory of computing (STOC), 169-178 (2009).
[5] C. Gentry, S. Halevi, and V. Vaikuntanathan, "A Simple BGN-Type Cryptosystem from LWE", Advances in Cryptology - EUROCRYPT 2010, Springer Verlag, LNCS, **6110**, 506-522 (2010).
[6] Y. Hu, W. J. Martin, B. Sunar. "Enhanced Flexibility for Homomorphic Encryption Schemes via CRT", Proc. ACNS, Springer Verlag, LNCS, **2012**, 93-110 (2012).
[7] A. Lenstra and E. Verheul, "Selecting Cryptographic Key Sizes." PKC 2000, Springer Verlag, LNCS, **1751** , 446-465 (2000).
[8] J. Stern, "Evaluation Report on the Factoring Problem. Evaluation of Cryptographic Techniques in FY 2003 (CRYPTREC 2002-2003)" http://www.ipa.go.jp/security/enc/CRYPTREC/fy15/doc/1023-R4-FACT.pdf, (2003).
[9] J. Tibor and S. Jorg, "The Generic Hardness of Subset Membership Problems under the factoring Assumption", http://eprint.iacr.org/2008/482.pdf, (2008).
[10] J. Tibor and S. Jorg, "On the Analysis of Cryptographic Assumptions in the Generic Ring Model", Jornal of Cryptology, (2012).
[11] P. Zimmermann, "The Elliptic Curve Method for Factoring.", http://www.loria.fr/~zimmerma/papers/ecm-entry2.pdf.
[12] P. Zimmermann, "50 largest factors found by ECM" , http://www.loria.fr/~zimmerma/records/top50.html

Hyang-Sook Lee is a Professor at the Department of Mathematics, Ewha Womans University, Seoul in Korea. She received the Ph.D from Northwestern University in 1993. Her research interests are pairing based cryptography, especially pairing computations, constructing pairing friendly curves, digital signatures, PKC etc. She has served on the Steering Committee of National Science and Technology Council in Korea.

Seongan Lim received her Ph.D. in Mathematics from Purdue University. She is a research professor at Ewha Womans University. Her research interests include public key cryptography, privacy preserving mechanisms in IT services.