# Cryptanalysis of a New Additive Homomorphic Encryption based on the co-ACD Problem

Moon Sung Lee

Seoul National University (SNU), Republic of Korea
moolee@snu.ac.kr

December 29, 2014

**Abstract.** In CCS'14, Cheon et al. proposed a new additive homomorphic encryption scheme which is claimed to be the most efficient among the additive homomorphic encryption schemes. The security is proved based on the hardness of a new problem, the (decisional) co-approximate common divisor problem. In this paper, we cryptanalyze the scheme and investigate the hardness of an aforementioned problem. Our first result shows that Cheon et al.'s scheme is insecure for the range of parameters considered in the original paper [2]. Experiments show that the message can be recovered in seconds for the proposed parameters. We also analyze the condition of the parameters to thwart the proposed attack. As a second result, we show that the co-approximate common divisor problem is easy for the similar range of parameters, in condition that the modulus is known and is a product of two primes. In our estimate, to thwart the proposed attack, the parameters should be enlarged many times. Apart from the scheme, the co-approximate common divisor problem itself is interestingly related to the well-known hard problem, an approximate common divisor problem. And further investigation on this relationship would be desirable.

## 1 Introduction

Quite recently, Cheon et al. proposed a new additive homomorphic encryption scheme in CCS'14 [2], which will be referred as a CLS scheme throughout this paper. This new scheme is very fast. Especially, the decryption takes only few micro-seconds. Compared to other additive homomorphic encryption schemes such as Paillier encryption [10] and Joye and Libert encryption [6], the CLS scheme is claimed to be the most efficient with similar security. Although there exist some drawbacks regarding the public key size and the limited homomorphic additions, the CLS scheme is quite interesting and have simple construction similar to the van Dijk et al.'s fully homomorphic encryption (FHE) scheme [5].

Let us briefly review the CLS scheme. Let $N = \prod_{i=1}^{k} p_i$ be a product of hidden primes. In the symmetric CLS scheme, $M \in \mathbb{Z}_Q$ is encrypted into a vector whose components are $(M+eQ \bmod p_i)$ for $i = 1, \ldots, k$, where $e$ is chosen in a sufficiently large interval $\mathbb{Z}\cap(-2^\rho, 2^\rho)$. Addition can be done component-wise, and the Chinese remainder theorem ensures the decryption. For the security, it is easy to see that $M + eQ \bmod p_i$ hides $M$ when $e \gg p_i$. However, it is not clear whether $(M + eQ \bmod p_1, M + eQ \bmod p_2)$ hides $M$ since there may exist information that can be extracted.

To overcome this difficulty, the authors of the CLS scheme introduced a new hard problem, co-approximate common divisor (co-ACD) problem. Then the security of the CLS scheme is proved based on the hardness of the decisional co-ACD problem. Roughly speaking, it is assumed that $(eQ \bmod p_1, \ldots, eQ \bmod p_k)$ is indistinguishable from the random vectors in $\mathbb{Z}_{p_1} \times \cdots \times \mathbb{Z}_{p_k}$ when $e$ is sufficiently large. This reminds the strategy taken on the ACD based FHE [1, 7] where the security is proved based on the new decisional ACD problem, which is later proved to be equivalent to the computational ACD problem [4]. As the name suggests, co-ACD problem has the similarity with the (extended) ACD problem. Based on this similarity, known attacks against the ACD problem is considered in the original paper [2].

Considering this similarity, the parameter choice in [2] is somewhat puzzling because it is several orders of magnitude smaller than the ACD based FHEs. For example, the ciphertext size of the CLS scheme providing 128-bit security starts from $3 \times 10^3$-bits while the recent (extended) ACD based FHE [4] suggest to use $15.8 \times 10^6$-bit ciphertexts for 72-bit security. Such a difference seems to be unnatural.

*Our Contributions.* In this paper, by cryptanalyzing the CLS scheme and investigating the co-ACD problem, we close this gap and show that the parameter of the co-ACD problem should be enlarged in the order of magnitude.

As our first contribution, we propose a message recovery attack against the CLS scheme using an orthogonal lattice. Following the usual strategy, we show that a vector $\boldsymbol{x}$ in the orthogonal lattice constructed from the ciphertexts yields a linear equation modulo $N$, and the obtained equation holds *over the integers* if $\boldsymbol{x}$ is short. Viewing this integer linear equation modulo $Q$, unknown errors can be removed and the message can be recovered. Our attack succeeds for the range of parameters including suggested parameters. Experiments with Sage [11] confirms that the proposed parameters are insecure. To thwart our attack, the ciphertext size should be $\omega(\lambda^2)$.

As a second contribution, we investigate the co-ACD problem when $N(= p_1 p_2)$ is known. We remark that the threshold version of the CLS scheme publishes $N$ whereas other versions do not. Combining a known attack against the ACD problem considered in [5] with a Coppersmith algorithm finding small roots of a univariate modulo equation [3], we show that the co-ACD problem is easy for the proposed parameters. Our analysis shows that this attack would work for the similar range of parameters to the message recovery attack.

*Organization.* We first review the CLS scheme in Section 2. The CLS scheme is cryptanalyzed in Section 3 when $k = 2$. General case is treated in Section 4. In Section 5, we show that the co-ACD problem is easy for wide range of parameters when the modulus $N(= p_1 p_2)$ is known.

## 2 Preliminary

*Notation.* We use lower-case bold letters for vectors, and usual inner product is denoted by $\langle \cdot \rangle$. For a pairwise coprime integers $p_1, \ldots, p_k$ and $N = \prod_{i=1}^{k} p_i$, we will use the following notation [2]:

$$\Phi_{(p_1,\ldots,p_k)} : \mathbb{Z}_N \to \prod_{i=1}^{k} \mathbb{Z}_{p_i}, \quad x \mapsto (x \bmod p_1, \ldots, x \bmod p_k).$$

Note that $\Phi$ is an isomorphism and $\Phi^{-1}$ is well-defined.

**Definition 1 (Orthogonal lattice).** *For vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n \in \mathbb{Z}^m$, the orthogonal lattice $L^{\perp}_{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n}$ consists of the integer vectors orthogonal to all $\boldsymbol{v}_i$'s. Namely,*

$$L^{\perp}_{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n} = \{\boldsymbol{x} \in \mathbb{Z}^m \mid \langle \boldsymbol{x}, \boldsymbol{v}_i \rangle = 0 \text{ for } i = 1, ..., n\}.$$

*This lattice has a dimension $m - n$ if $\boldsymbol{v}_i$'s are linearly independent.*

### 2.1 Review of the CLS scheme

We first review the co-ACD problem.

**Definition 2 (co-ACD problem).** *The $(\rho, \eta, k; Q)$-co-Approximate Common Divisor (co-ACD) problem is defined as follows. Pick $\eta$-bit random hidden primes $p_i$ for $i = 1, \ldots, k$. Given polynomially many samples $\Phi_{(p_1,\ldots,p_k)}(eQ)$ where $e$ is randomly selected in $\mathbb{Z} \cap (-2^{\rho}, 2^{\rho})$, find a nontrivial factor of $\prod_{i=1}^{k} p_i$.*

Now we describe the symmetric version of the CLS scheme. Let $Q$ be a positive integer such that the message space is $\mathbb{Z}_Q$. The message $M$ is added with an error, and reduced modulo hidden moduli. The details follows.

- Setup($1^\lambda$): Generate two $\eta$-bit primes $p_1, p_2$ with the condition $\gcd(Q, p_i) = 1$. Set $N = p_1p_2$. For each $i = 1, 2$, compute $\bar{p_i} = p_j(p_j^{-1} \bmod p_i) \bmod N$ for $j \neq i$. Output the secret key $sk = \{N, p_1, p_2, \bar{p_1}, \bar{p_2}\}$.
- Enc($sk$,$M$): Choose $e \leftarrow \mathbb{Z} \cap (-2^\rho, 2^\rho)$. For a message $M \in \mathbb{Z}_Q$, the ciphertext is a vector $\boldsymbol{c} = \Phi_{(p_1,p_2)}(M + eQ) = (M + eQ \bmod p_1, M + eQ \bmod p_2)$.
- Dec($sk$, $\boldsymbol{c}$): For a ciphertext $\boldsymbol{c} = (c_1, c_2)$, output $M = \Phi^{-1}_{(p_1,p_2)}(c_1, c_2) = (c_1\bar{p_1} + c_2\bar{p_2} \bmod N) \bmod Q$.
- Add($\boldsymbol{c_1}, \ldots, \boldsymbol{c}_\ell$): Output $\boldsymbol{c} = \sum_{i=1}^{\ell} \boldsymbol{c}_i$ through component-wise integer additions.

The asymmetric version of the CLS scheme can be constructed by publishing encryptions of zeros. The security of the resulting scheme can be proved using the leftover hash lemma over the lattices. For details, we refer to the original paper [2].

This scheme is simple and efficient additive homomorphic encryption. The authors provided concrete parameters for 128-bit security with implementation results which shows that the decryption and the addition is faster than the previous additive homomorphic schemes.

In the next section, we cryptanalyze the CLS scheme and it turns out that their parameter choice is too optimistic.

## 3 Cryptanalysis of the CLS scheme

In this section, we describe the known plaintext attack against the CLS scheme with implementation results. Proposed parameters are weak and it needs only seconds to recover the message.

In the following, we assume that all ciphertexts are fresh.

### 3.1 Message Recovery Attack against the CLS scheme

Our attack uses an orthogonal lattice which has various applications in cryptanalysis including the hidden subset sum problem analyzed by Nguyen and Stern in [9]. The orthogonal lattice attack is used to estimate the security of the ACD-based FHE scheme [5]. Similarly, an orthogonal lattice attack is also considered to analyze the parameter of the CLS scheme in the original paper [2], where an orthogonal lattice constructed from $(k-1)$ components of ciphertext vectors is used to find a hidden prime $p_k$ when $N = \prod_{j=1}^{k} p_j$. And it is concluded that $\rho = (k-1)\eta + 2\lambda$ is a safe choice when the security parameter is $\lambda$.

In the following, we describe a different orthogonal lattice attack using all $k$ components of ciphertext vectors. To thwart our attack, $\rho$ should be enlarged greatly, which reduces the homomorphic capacity. Our attack does not disclose the hidden prime, but reveals hidden linear equations of the messages, which enables to recover the message on the target ciphertext.

In this section, we cryptanalyze the CLS scheme when $N = p_1p_2$. When $\rho$ is set to be $(k-1)\eta + 2\lambda$, choosing $k = 2$ is the most efficient on the same ciphertext size. The general case will be treated in the next section.

We follow the same strategy as in the previous orthogonal lattice attacks [9, 5]. The analysis is slightly different, and simplified because of the conditions on the parameters, in our case.

To describe the proposed attack, let $N = p_1p_2$ and $\boldsymbol{c}_i = \Phi_{(p_1,p_2)}(M_i + e_iQ) = (M_i + e_iQ \bmod p_1, M_i + e_iQ \bmod p_2) = (c_{i,1}, c_{i,2})$ with known messages $M_i \in \mathbb{Z}_Q$ for $i = 1, \ldots, t$. Let the target ciphertext $\boldsymbol{c} = \Phi_{(p_1,p_2)}(M + eQ) = (c_1, c_2)$ with an unknown message $M$.

Let the hidden vector $\boldsymbol{e} = (M+eQ, M_1+e_1Q, \ldots, M_t+e_tQ)$. From $\boldsymbol{c}_i$ and $\boldsymbol{c}$, we construct two vectors $\boldsymbol{v}_j = (c_j, c_{1,j}, \ldots, c_{t,j})$ for $j = 1, 2$. By the construction, it is easy to see that $\boldsymbol{v}_j = \boldsymbol{e} \bmod p_j$. To see the relation to the hidden subset sum problem, let $\boldsymbol{e}'_j \in \mathbb{Z}^{t+1}$ such that

$$\boldsymbol{e} - \boldsymbol{v}_j = p_j \boldsymbol{e}'_j \text{ for } j = 1, 2. \tag{1}$$

Using (1), we obtain the following equation:

$$\boldsymbol{v}_1 - \boldsymbol{v}_2 = p_2 \boldsymbol{e}'_2 - p_1 \boldsymbol{e}'_1. \tag{2}$$

Since only $\boldsymbol{v}_1 - \boldsymbol{v}_2$ is known, the equation (2) can be seen as a variant of the hidden subset sum problem. The difference lies in that $\boldsymbol{e}'_1$ and $\boldsymbol{e}'_2$ are nearly parallel and have entries roughly $(2\lambda + \log Q)$-bit whereas hidden vectors are random independent binary vectors in the hidden subset sum problem [9]. This leads us to the consequence that, $\boldsymbol{e}'_1$ and $\boldsymbol{e}'_2$ can not be obtained directly from the (much shorter) reduced basis of the lattice generated by $\boldsymbol{e}'_1$ and $\boldsymbol{e}'_2$. And this is the reason why our attack does not disclose the hidden primes.

In the following, we show that the orthogonal lattice still contains enough information to recover the message of the target ciphertext.

The attack goes in two steps.

- Step 1: Find a short vector $\boldsymbol{x}$ in $L^{\perp}_{\boldsymbol{v}_1 - \boldsymbol{v}_2}$ such that $\langle \boldsymbol{x}, \boldsymbol{e} - \boldsymbol{v}_1 \rangle = 0$ which yields a linear equation over the integers.
- Step 2: Viewing the above linear equation $\langle \boldsymbol{x}, \boldsymbol{e} - \boldsymbol{v}_1 \rangle = 0$ modulo $Q$, we eliminates $e$ and $e_i$'s, and recover the message.

For the first step, we construct an orthogonal lattice $L^{\perp}_{\boldsymbol{v}_1 - \boldsymbol{v}_2}$ of dimension $t$. Let $\boldsymbol{x}$ be a vector in the lattice $L^{\perp}_{\boldsymbol{v}_1 - \boldsymbol{v}_2}$. Then, the following holds:

$$\langle \boldsymbol{x}, \boldsymbol{e} - \boldsymbol{v}_1 \rangle \equiv \langle \boldsymbol{x}, \boldsymbol{v}_1 - \boldsymbol{v}_1 \rangle = 0 \pmod{p_1},$$
$$\langle \boldsymbol{x}, \boldsymbol{e} - \boldsymbol{v}_1 \rangle \equiv \langle \boldsymbol{x}, \boldsymbol{v}_2 - \boldsymbol{v}_1 \rangle = 0 \pmod{p_2}.$$

Thus, we get the following equation:

$$\langle \boldsymbol{x}, \boldsymbol{e} - \boldsymbol{v}_1 \rangle \equiv 0 \pmod{N}.$$

Now if $\|\boldsymbol{x}\|$ is less than $N/\|\boldsymbol{e} - \boldsymbol{v}_1\| \approx 2^{2\eta - \rho - \log Q} = 2^{\eta - 2\lambda - \log Q}$, then

$$\|\langle \boldsymbol{x}, \boldsymbol{e} - \boldsymbol{v}_1 \rangle\| \le \|\boldsymbol{x}\| \|\boldsymbol{e} - \boldsymbol{v}_1\| < N,$$

which implies that the inner product $\langle \boldsymbol{x}, \boldsymbol{e} - \boldsymbol{v}_1 \rangle$ is actually zero over the integers. This yields an integer linear equation and finding a such short vector is enough to proceed to the next step.

In the second step, we actually recover the message $M$. Using the vector $\boldsymbol{x} = (x_0, x_1, \ldots, x_t) \in L^{\perp}_{\boldsymbol{v}_1 - \boldsymbol{v}_2}$ obtained in the first step, we can obtain the following integer linear equation:

$$\langle \boldsymbol{x}, \boldsymbol{e} - \boldsymbol{v}_1 \rangle = x_0 \cdot (M + eQ - c_1) + \sum_{i=1}^{t} x_i \cdot (M_i + e_iQ - c_{i,1}) = 0.$$

Viewing this equation modulo $Q$, we get

$$x_0 \cdot (M - c_1) + \sum_{i=1}^{t} x_i \cdot (M_i - c_{i,1}) = 0 \pmod{Q}. \tag{3}$$

And solving the equation (3) modulo $Q$ yields the message $M$ if $\gcd(x_0, Q) = 1$. As long as $x_0 \not\equiv 0 \pmod{Q}$, we can still recover the partial message. Repeating these two steps with different known ciphertexts would eventually yield the message.

We now briefly analyze the above algorithm using the proposed parameters. As in the Table 1, $\rho$ is set to be $\eta + 2\lambda$ for the security parameter $\lambda = 128$ with $\log Q = 256$ [2]. Since $\eta$ is ranged from 1536 to 2706, $\eta - 2\lambda - \log Q > 1000$. Thus, it is enough to find a vector in the lattice $L^{\perp}_{\boldsymbol{v}_1 - \boldsymbol{v}_2}$ of length less than $2^{1000}$. Since this lattice has a determinant $\|\boldsymbol{v}_1 - \boldsymbol{v}_2\| \approx 2^{\eta}$ and the dimension is $t$, we need to choose $t$ such that $\eta/t < \eta - 2\lambda - \log Q$. Setting $t = 3$ satisfies the condition and lattice basis reduction of dimension three lattice is quite easy. As is described in the next subsection, our attack is efficient.

*Remark 1.* Any vector $\boldsymbol{x} \in L^{\perp}_{\boldsymbol{v}_1 - \boldsymbol{v}_2}$ of length less than $N/\|\boldsymbol{e} - \boldsymbol{v}_1\|$ can be used to generate integer linear equation. Thus, one can use several linearly independent vectors to reduce the required number of known plaintexts.

### 3.2 Experimental results

We implemented our attack on the proposed parameters [2] using Sage [11] with a desktop computer running on 2.8GHz with 12GB RAM. With three known plaintexts and the target ciphertext, we first constructed a lattice basis $L^{\perp}_{\boldsymbol{v}_1 - \boldsymbol{v}_2}$ using `kernel` command. Then LLL algorithm [8] is applied to find a reduced basis. In our experiment, at least one of the two short vectors in a reduced basis has a first component coprime to $Q$, and the message is successfully recovered.

We note that the experiments are performed 100 times on the three parameters. As in the Table 1, the message can be obtained less than 2 seconds.

**Table 1.** Message Recovery Attack, Timing results

| $\lambda$ | $\eta$ | $\rho$ | $\log Q$ | $\log A$ | $t$ | Time (seconds) |
|---|---|---|---|---|---|---|
| 128 | 1536 | 1792 | 256 | 1134 | 3 | $< 1$ |
| 128 | 2194 | 2450 | 256 | 1536 | 3 | 1 |
| 128 | 2706 | 2962 | 256 | 2048 | 3 | 1.8 |

## 4 Orthogonal Lattice Attack for $k \geq 2$

The CLS scheme in [2] is instantiated with a product of two prime modulus $N = p_1 p_2$ due to the efficiency reason. The extension to the three or more primes is very natural. In this section, we show that our attack can be extended to this case. The analysis shows that the CLS scheme would lose efficiency to thwart the proposed attack.

Throughout this section, we will use the following notations. Let $N = \prod_{j=1}^{k} p_j$ be (possibly) secret modulus. Let us assume that we know $t$ ciphertexts $\boldsymbol{c_i} = (c_{i,1}, c_{i,2}, \ldots, c_{i,k}) = (M_i + e_i Q \bmod p_1, \ldots, M_i + e_i Q \bmod p_k)$ for $i = 1, ..., t$ with corresponding plaintexts $M_i$. Let the target ciphertext $\boldsymbol{c} = (c_1, \ldots, c_k) = (M + eQ \bmod p_1, \ldots, M + eQ \bmod p_k)$.

Let $\boldsymbol{e} = (M + eQ, M_1 + e_1 Q, \ldots, M_t + e_t Q)$ be a hidden vector. For $j = 1, \ldots, k$, we construct vectors $\boldsymbol{v}_j = (c_j, c_{1,j}, c_{2,j}, ..., c_{t,j}) \in \mathbb{Z}^{t+1}$ from known ciphertexts and target ciphertext. Note that $\boldsymbol{v}_j = \boldsymbol{e} \bmod p_j$. Let $L = L^{\perp}_{\boldsymbol{v}_2 - \boldsymbol{v}_1, ..., \boldsymbol{v}_k - \boldsymbol{v}_1}$ be an orthogonal lattice obtained from $\boldsymbol{v}_j$'s. Throughout this section, we will use $L$ for this lattice. The following lemma says that any vector in this lattice yields a linear equation modulo $N$.

**Lemma 1.** *For any vector $\boldsymbol{x} \in L^{\perp}_{\boldsymbol{v}_2 - \boldsymbol{v}_1, ..., \boldsymbol{v}_k - \boldsymbol{v}_1}$, the following holds for $j = 1, \ldots, k$:*

$$\langle \boldsymbol{x}, \boldsymbol{e} - \boldsymbol{v}_j \rangle \equiv 0 \pmod{N}$$

*Proof.* It is easy to see that $\langle \boldsymbol{x}, \boldsymbol{e} - \boldsymbol{v}_1 \rangle \equiv 0 \pmod{N}$ since $\boldsymbol{e} - \boldsymbol{v}_1 \equiv 0 \pmod{p_1}$ by the construction and $\langle \boldsymbol{x}, \boldsymbol{e} - \boldsymbol{v}_1 \rangle \equiv \langle \boldsymbol{x}, \boldsymbol{v}_j - \boldsymbol{v}_1 \rangle = 0 \pmod{p_j}$ for $j = 2, \ldots, k$ since $\boldsymbol{x}$ is contained in $L^{\perp}_{\boldsymbol{v}_2 - \boldsymbol{v}_1, \ldots, \boldsymbol{v}_k - \boldsymbol{v}_1}$. Now we have $\langle \boldsymbol{x}, \boldsymbol{e} - \boldsymbol{v}_j \rangle \equiv 0 \pmod{N}$ since $\langle \boldsymbol{x}, \boldsymbol{e} - \boldsymbol{v}_j \rangle = \langle \boldsymbol{x}, \boldsymbol{e} - \boldsymbol{v}_1 \rangle - \langle \boldsymbol{x}, \boldsymbol{v}_j - \boldsymbol{v}_1 \rangle \equiv 0 \pmod{N}$ for $j = 2, \ldots, k$. □

Similar to the previous section, we need to find short vectors from the lattice $L$ of dimension $(t - k + 2)$ using lattice basis reduction algorithms. We note that the determinant of $L$ is at most $\prod_{i=1}^{k-1} \|\boldsymbol{v}_i\| \approx 2^{(k-1)\eta}$.

By Lemma 1, we know that $\boldsymbol{x} \in L$ implies that $\langle \boldsymbol{x}, \boldsymbol{e} - \boldsymbol{v}_1 \rangle \equiv 0$ modulo $N$. Thus, $\|\boldsymbol{x}\| < N / \|\boldsymbol{e} - \boldsymbol{v}_1\|$ imples that $\langle \boldsymbol{x}, \boldsymbol{e} - \boldsymbol{v}_1 \rangle = 0$. Viewing this integer linear equation modulo $Q$, we can recover the message $M$ similar to the previous section, once we obtained a short vector $\boldsymbol{x} \in L$ such that the first component of $\boldsymbol{x}$ is coprime to $Q$. This coprime condition can be easily satisfied by trying several short vectors or using different known ciphertexts. And the satisfying probability is rather high assuming that the first component of $\boldsymbol{x}$ is random modulo $Q$. For example, when $Q = 2^{256}$ [2] to provide 256-bit message space, the probability is $\frac{1}{2}$. For a prime $Q$, the probability is $\frac{Q-1}{Q}$. Thus, the only obstacle might be the possibility of getting short vectors of length less than $N / \|\boldsymbol{e} - \boldsymbol{v}_1\| \approx 2^{k\eta - \rho - \log Q}$.

By the Minkowski's theorem, we know that $L$ contains a vector of length less than $\sqrt{n} \det L^{1/n} \approx \sqrt{n}\, 2^{(k-1)\eta/n}$ where $n = t - k + 2$ is a dimension of $L$. Ignoring $\sqrt{n}$, we get the following condition:

$$\frac{(k-1)\eta}{n} < k\eta - \rho - \log Q \Longleftrightarrow t > \frac{(k-1)\eta}{k\eta - \rho - \log Q} + k - 2. \tag{4}$$

Let $t_0 = \lceil (k-1)\eta / (k\eta - \rho - \log Q) + k - 2 \rceil$. Then finding the shortest vector in a $t_0$-dimensional lattice suffices to break the CLS scheme with high probability.

On the other hand, LLL [8] algorithm is guaranteed to find a vector of length less than $2^{(n-1)/4} \det L^{1/n} \approx 2^{(n-1)/4 + (k-1)\eta/n}$. Thus when the following condition is satisfied, we can expect to recover the message in polynomial time:

$$\frac{n-1}{4} + \frac{(k-1)\eta}{n} < k\eta - \rho - \log Q \text{ where } n = t - k + 2. \tag{5}$$

Since $a + b \geq 2\sqrt{ab}$ for positive $a$,$b$, the left hand side of (5) is minimized when $(n-1)/4 = (k-1)\eta/n$. Then ignoring $\sqrt{(n-1)/n}$ term, we get the following condition:

$$\sqrt{(k-1)\eta} < k\eta - \rho - \log Q. \tag{6}$$

This proves the following theorem.

**Theorem 1** *The CLS scheme with parameter $(\rho, \eta, k; Q)$ is insecure if one of the following conditions are satisfied:*

1. $\lceil (k-1)\eta / (k\eta - \rho - \log Q) + k - 2 \rceil$ *is small,*
2. *(6) is satisfied.*

Since $k\eta - \rho - \log Q$ is $\log A$ for the symmetric CLS scheme where $A$ is the maximum number of allowed additions among fresh ciphertexts, this theorem means that $\sqrt{(k-1)\eta}$ should be larger than $\log A$. To provide $2^{\lambda}$ additions, parameters should be set to be $(k - 1)\eta = \omega(\lambda^2)$. Considering only this attack, larger $k$ reduces ciphertexts slightly since $\gamma = \frac{k}{k-1}(k-1)\eta$.

In the next section, we will analyze the computational co-ACD problem.

## 5 Analysis of the computational co-ACD problem

In this section, we examine the computational co-ACD problem assuming that the modulus $N$ is known. We note that $N$ is public in the threshold version of the CLS scheme.

Let $N = p_1 p_2$ with co-ACD samples $\boldsymbol{c}_i = (c_{i,1}, c_{i,2}) = (e_i Q \bmod p_1, e_i Q \bmod p_2)$ for $i = 1, \ldots, t$. Let $\boldsymbol{e} = (e_1 Q, \ldots, e_t Q)$ be a secret vector. Now, as we have done in previous sections, construct two vectors $\boldsymbol{v}_j = (c_{1,j}, \ldots, c_{t,j})$ for $j = 1, 2$ from $\boldsymbol{c}_i$'s. Note that $\boldsymbol{v}_j = \boldsymbol{e} \bmod p_j$ for $j = 1, 2$, once again. By the Chinese remainder theorem, we have the following equation:

$$\boldsymbol{e} - \boldsymbol{v}_1 \equiv (\boldsymbol{v}_2 - \boldsymbol{v}_1)\bar{p_2} \pmod{N},$$

where $\bar{p_2} = p_1(p_1^{-1} \bmod p_2) \bmod N$. Similar to the lattice used against the ACD problem in [5], considering the above equation, we construct a lattice generated by the following $(t+1) \times t$ matrix:

$$L = \begin{pmatrix} (c_{1,2} - c_{1,1}) & (c_{2,2} - c_{2,1}) & \ldots & (c_{t,2} - c_{t,1}) \\ N & & & \\ & N & & \\ & & \ddots & \\ & & & N \end{pmatrix} = \begin{pmatrix} \boldsymbol{v}_2 - \boldsymbol{v}_1 \\ N\, I_{t \times t} \end{pmatrix}$$

Note that the length of the first row vector of $L$ is $\|\boldsymbol{v}_2 - \boldsymbol{v}_1\| \approx 2^{\eta}$. This is about the half size of $N \approx 2^{2\eta}$. On the other hand, our target solution is $(\bar{p_2}, *, \ldots, *) \cdot L = (e_1 Q - c_{1,1}, \ldots, e_t Q - c_{t,1}) = \boldsymbol{e} - \boldsymbol{v}_1$ where $\|\boldsymbol{e} - \boldsymbol{v}_1\| \approx 2^{\rho + \log Q} < N$.

For simplicity, let $L$ be a lattice generated by this matrix. Since our target vector is much longer than the shortest vector $\boldsymbol{v}_2 - \boldsymbol{v}_1$ in $L$, the best we can hope is that the second shortest vector in $L$ is $\boldsymbol{e} - \boldsymbol{v}_1$ (modulo $\boldsymbol{v}_2 - \boldsymbol{v}_1$). Using $\det L = N^{t-1} \approx 2^{2\eta(t-1)}$ and $\|\boldsymbol{v}_2 - \boldsymbol{v}_1\| \approx 2^{\eta}$, we expect that the length of the second shortest vector in $L$ would be $\ell_2 = (2^{2\eta(t-1) - \eta})^{1/(t-1)} = 2^{2\eta - \frac{\eta}{t-1}}$. Thus, we can expect to find $\boldsymbol{e} - \boldsymbol{v}_1$ in this lattice (modulo $\boldsymbol{v}_2 - \boldsymbol{v}_1$) if $\|\boldsymbol{e} - \boldsymbol{v}_1\| < \ell_2$. This yields the following condition on $t$:

$$t > \frac{\eta}{2\eta - \rho - \log Q} + 1. \tag{7}$$

By choosing $t$ satisfying (7) and using lattice basis reduction algorithms on a $t$-dimensional lattice $L$, we can obtain $\tilde{\boldsymbol{v}}$ which is close to $\boldsymbol{e} - \boldsymbol{v}_1$ when $t$ is small. Now, assuming that $\boldsymbol{e} - \boldsymbol{v}_1 = \tilde{\boldsymbol{v}} + x(\boldsymbol{v}_2 - \boldsymbol{v}_1)$ with a bound $|x| < 2^{\rho + \log Q - \eta}$, we can use the first component $\alpha, \beta$ of the first two non-zero shortest vectors in $L$ to construct a degree-2 polynomial $f(x) = (\alpha x + \beta)(\alpha x + \beta + c_{1,1} - c_{1,2})$. Finding a small root $x_0$ of $f(x)$ modulo $N$ using Coppersmith's method [3] and computing the greatest common divisor, $\gcd(N, \alpha x_0 + \beta)$, would factor $N$.

In experiments using Sage [11], $N$ is easily factored for the parameter $(\eta, \log Q) = (1536, 256)$ when $\rho$ is smaller than 2700.

## Acknowledgments

The authors thank Jung Hee Cheon, Changmin Lee, Jae Hong Seo, and Yong Soo Song for helpful discussions.

## References

1. J. Cheon, J.-S. Coron, J. Kim, M. Lee, T. Lepoint, M. Tibouchi, and A. Yun. Batch fully homomorphic encryption over the integers. In T. Johansson and P. Nguyen, editors, *Advances in Cryptology – EUROCRYPT 2013*, volume 7881 of *Lecture Notes in Computer Science*, pages 315–335. Springer Berlin Heidelberg, 2013.

2. J. H. Cheon, H. T. Lee, and J. H. Seo. A new additive homomorphic encryption based on the co-acd problem. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, CCS '14, pages 287–298, New York, NY, USA, 2014. ACM.
3. D. Coppersmith. Finding a small root of a univariate modular equation. In U. Maurer, editor, *Advances in Cryptology — EUROCRYPT '96*, volume 1070 of *Lecture Notes in Computer Science*, pages 155–165. Springer Berlin Heidelberg, 1996.
4. J.-S. Coron, T. Lepoint, and M. Tibouchi. Scale-invariant fully homomorphic encryption over the integers. In H. Krawczyk, editor, *Public-Key Cryptography – PKC 2014*, volume 8383 of *Lecture Notes in Computer Science*, pages 311–328. Springer Berlin Heidelberg, 2014.
5. M. v. Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan. Fully homomorphic encryption over the integers. In H. Gilbert, editor, *Advances in Cryptology – EUROCRYPT 2010*, volume 6110 of *Lecture Notes in Computer Science*, pages 24–43. Springer Berlin / Heidelberg, 2010.
6. M. Joye and B. Libert. Efficient cryptosystems from 2 k -th power residue symbols. In T. Johansson and P. Nguyen, editors, *Advances in Cryptology – EUROCRYPT 2013*, volume 7881 of *Lecture Notes in Computer Science*, pages 76–92. Springer Berlin Heidelberg, 2013.
7. J. Kim, M. S. Lee, A. Yun, and J. H. Cheon. Crt-based fully homomorphic encryption over the integers. Cryptology ePrint Archive, Report 2013/057, 2013. http://eprint.iacr.org/.
8. A. Lenstra, J. Lenstra, H.W., and L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261(4):515–534, 1982.
9. P. Nguyen and J. Stern. The hardness of the hidden subset sum problem and its cryptographic implications. In M. Wiener, editor, *Advances in Cryptology — CRYPTO' 99*, volume 1666 of *Lecture Notes in Computer Science*, pages 31–46. Springer Berlin Heidelberg, 1999.
10. P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In J. Stern, editor, *Advances in Cryptology — EUROCRYPT '99*, volume 1592 of *Lecture Notes in Computer Science*, pages 223–238. Springer Berlin Heidelberg, 1999.
11. W. Stein et al. *Sage Mathematics Software (Version 6.1)*. The Sage Development Team, 2014. `http://www.sagemath.org`.