Cryptanalysis of the Co-ACD Assumption

Pierre-Alain Fouque[1], Moon Sung Lee[2], Tancrède Lepoint[3], and Mehdi Tibouchi[4]

[1] Université de Rennes 1 and Institut Universitaire de France, fouque@irisa.fr
[2] Seoul National University (SNU), moolee@snu.ac.kr
[3] CryptoExperts, tancrede.lepoint@cryptoexperts.com
[4] NTT Secure Platform Laboratories, tibouchi.mehdi@lab.ntt.co.jp

Abstract. At ACM-CCS 2014, Cheon, Lee and Seo introduced a new number-theoretic assumption, the Co-Approximate Common Divisor (Co-ACD) assumption, based on which they constructed several cryptographic primitives, including a particularly fast additively homomorphic encryption scheme. For their proposed parameters, they found that their scheme was the "most efficient of those that support an additive homomorphic property".

In this paper, we analyze the security of the Cheon–Lee–Seo (CLS) homomorphic encryption scheme and of the underlying Co-ACD assumption, and present several lattice-based attacks that are effectively devastating for the proposed constructions. First, we prove that a few known plaintexts are sufficient to decrypt any ciphertext in the symmetric-key CLS scheme. This breaks the one-wayness of both the symmetric-key and the public-key variants of CLS encryption as well as the underlying decisional Co-ACD assumption for a very wide range of parameters. Then, we show that this attack can be heuristically extended to decrypt small messages without any known plaintext. And finally, we find that Coppersmith's theorem can even be used to solve the search variant of the Co-ACD problem, and mount a full key recovery on the public-key CLS scheme.

Concretely speaking, the parameters proposed by Cheon et al. and originally aiming at 128-bit security can be broken in a matter of seconds. And while it is possible to select parameters outside of the range in which our attacks run in polynomial time, they have to be so large as to render the proposed constructions severely uncompetitive (e.g. our asymptotic estimates indicate that 128 bits of security against our attacks require a modulus of at least 400,000 bits).

Keywords: Cryptanalysis, Lattice Reduction, Coppersmith Theorem, Homomorphic Encryption, Co-ACD Problem.

1 Introduction

At ACM-CCS 2014, Cheon, Lee and Seo [CLS14] introduced a new hardness assumption called the co-approximate common divisor assumption (Co-ACD). Informally, the decisional Co-ACD assumption states that it is hard to distinguish (without knowing the primes p_i's) between the uniform distribution over $\mathbb{Z}_{p_1} \times \cdots \times \mathbb{Z}_{p_n}$ and the distribution which outputs $(e \cdot Q \bmod p_i)_{i=1}^n$, where Q is a *known* value and e is some uniformly distributed noise in $(-2^\rho, 2^\rho)$. It is assumed that $\max\{p_1, \ldots, p_k\} < 2^\rho \cdot Q < \prod_i p_i$ to make the problem non trivial. The search Co-ACD assumption states that, given arbitrarily many samples from the distribution above, it is hard to recover the p_i's themselves. To validate the plausible hardnesses of these assumptions, the authors provided a cryptanalytic survey [CLS14, Sec. 4] based on known and new dedicated attacks: the algebraic approach due to Chen–Nguyen [CN12, CNT12], orthogonal lattices [NS98, NS99] and Coppersmith's theorem [Cop97, How01, CH12].

Based on the hardness of the decisional problem, the authors then proposed a very efficient additive homomorphic encryption scheme which outperformed competitors such as [Pai99, NLV11, JL13] by several orders of magnitude. In this scheme, a message $m \in \mathbb{Z}_Q$ is encrypted as $(c_1, c_2) =$

$\left(m + e \cdot Q \bmod p_1,\ m + e \cdot Q \bmod p_2\right)$ for large enough primes p_1, p_2 which form the secret key. The hope is that $eQ > p_1,\ p_2$ will hide the message m for each component c_1 and c_2 (which is indeed the case if the decisional Co-ACD assumption holds), while still allowing decryption using the Chinese Remainder Theorem for users who know the secret key, since $eQ < p_1 p_2$. This is a symmetric-key scheme, but can be converted to public-key using a transformation similar to the one from [DGHV10].

As the name suggests, the Co-ACD assumption has some similarity with the (extended) approximate common divisor (ACD) assumption, which has been used to construct various primitives including fully homomorphic encryption [DGHV10, CCK+13, CLT14]. In the ACD problem, the goal is to recover p given samples of the form $x = pq + r$ where q is uniformly distributed in $[0, 2^\gamma/p)$ and r is uniformly distributed in $(-2^\rho, 2^\rho)$. This problem has been introduced by Howgrave-Graham in [How01] and lattice reduction algorithms have been used to solve this problem using Coppersmith's theorem [Cop97, CH12], as well as other algebraic techniques [CN12, CNT12]. In view of that similarity, the parameter choice in [CLS14] seems rather bold: for example, the authors claim 128 bits of security with ciphertexts as small as 3000 bits, whereas even if we restrict the recent ACD-based fully homomorphic encryption scheme [CLT14] to homomorphic additions, ciphertext size for that scheme is still at least cubic in the security parameter, so ciphertexts have to be millions of bits long for 128 bits of security.

Our Contributions. In this paper, we present three new attacks, targeting the Cheon–Lee–Seo (CLS) encryption scheme (in both its symmetric-key and public-key incarnations) as well as the decisional and search variants of the Co-ACD assumption. Our new attacks severely reduce the security of all proposed constructions from [CLS14].

Our first attack, described in Section 3, in a known-plaintext attack against the symmetric-key CLS scheme. We establish that a few known plaintext-ciphertext pairs are sufficient to decrypt any ciphertext. This attack breaks the one-wayness of the symmetric-key CLS scheme, and it is straightforward to use it to break the one-wayness of the public-key CLS scheme and the decisional Co-ACD assumption as well. The algorithm is provable, runs in polynomial time on a wide range of parameters, and while it is possible to select parameters outside of that range, they need to be huge to achieve security, resulting in a scheme of little practical use.

Our second attack, discussed in Section 4, is a ciphertext-only attack on the symmetric-key CLS scheme. It allows an attacker to decrypt, without any known plaintext, a set of ciphertexts corresponding to small messages. Combined with the first attack, this makes it possible to decrypt arbitrary ciphertexts given only a few ciphertexts corresponding to small messages. This stronger attack uses the more advanced "doubly orthogonal lattice" technique of Nguyen–Stern, which makes it heuristic, but we find that it is very effective as well in practice.

Finally, our third attack, discussed in Section 5, solves the search variant of the Co-ACD problem in a wide range of parameters, and can in particular be used to factor the modulus of the public-key CLS scheme, revealing the entire private key. This attack combines a pure lattice step together with a generalization of Coppersmith's theorem due to Alexander May.

We present each of these attacks both in the case when $n = 2$ (i.e. the modulus N is a product of two primes), which is the one considered by Cheon et al. to construct their encryption schemes, and in the case of larger n, for which the Co-ACD assumptions are still defined, and the encryption schemes admit natural generalizations. We also provide extensive experiments that show that our attacks completely break the parameters proposed in [CLS14] in a very concrete way.

Related Work. Our attacks use orthogonal lattice techniques (as discussed e.g. in [NT12]), originally introduced by Nguyen and Stern in several attacks [NS97, NS98, NS99] against the Qu–Vanstone knapsack-based scheme [QV94], the Itoh–Okamoto–Mambo cryptosystem [IOM97] and the hidden subset sum problem. Similar techniques were also used in other cryptanalytic works, but work only with one modulus p_1 (either known but hard to factor [DGHV10, CNT10], or unknown [LT15]). In the original paper [CLS14], orthogonal lattice attacks were already considered to set the parameter $\rho = (n-1)\eta + 2\lambda$ for λ bits of security; we obtain better attacks, however, by considering different lattices as well as extended attack techniques.

Notation. For any integer n, we denote by $[n]$ the set $\{1, \ldots, n\}$ and by $\mathbb{Z}_n$ the ring of integers modulo n. We use $a \xleftarrow{\$} A$ to denote the operation of uniformly sampling an element a from a finite set A. If χ is a distribution, the notation $a \leftarrow \chi$ refers to sampling a according to the distribution χ. We let λ be the security parameter. We use bold letters for vectors and the inner product of two vectors $\boldsymbol{u}, \boldsymbol{v}$ is denoted by $\langle \boldsymbol{u}, \boldsymbol{v} \rangle$.

2 Preliminaries

In this section, we recall the additive homomorphic schemes (symmetric and public-key) proposed by Cheon–Lee–Seo (CLS) at ACM-CCS 2014 [CLS14], and its underlying security assumption called the co-approximate common divisor assumption (Co-ACD). We also give some background on lattices, orthogonal lattices and Coppersmith's algorithm to find small roots of modular polynomial equations.

2.1 CLS Somewhat Additively Homomorphic Encryption Schemes

Secret-Key Scheme. Given the security parameter λ, we use the following parameters: η the bit-length of the secret key elements p_i's, Q the size of the message space $\mathbb{Z}_Q$, ρ the bit-length of the error. The CLS scheme then consists of the following algorithms.

$\mathsf{CLS.KeyGen}(1^\lambda)$: Generate two random prime integers p_1, p_2 of η bits, and output $\mathsf{sk} = \{p_1, p_2\}$.

$\mathsf{CLS.Encrypt}(\mathsf{sk}, m \in \mathbb{Z}_Q)$: Generate a random noise $e \xleftarrow{\$} (-2^\rho, 2^\rho)$, and output $\boldsymbol{c} = (c_1, c_2) = (m + e \cdot Q \bmod p_1, m + e \cdot Q \bmod p_2)$.

$\mathsf{CLS.Decrypt}(\mathsf{sk}, \boldsymbol{c})$: Parse $\boldsymbol{c} = (c_1, c_2)$. Compute $e' = c_1 + p_1 \cdot (p_1^{-1} \bmod p_2) \cdot (c_2 - c_1) \bmod (p_1 p_2)$ and output $e' \bmod Q$.

This completes the description of the scheme using Garner's formula to improve the decryption. When $2^\rho \cdot Q \leq 2^{2\eta-2} < p_1 p_2$, the previous scheme is obviously correct. As shown in [CLS14], this scheme is also somewhat additively homomorphic when adding the ciphertexts componentwise over $\mathbb{Z}$, i.e. a limited number of homomorphic additions (at least $2^{\eta-3-\rho-\lceil \log_2 Q \rceil}$) can be performed on fresh ciphertexts while preserving correctness.

Public-Key Variant. A public key variant of the latter scheme was also proposed in [CLS14]. The public key pk then consists of the public modulus $N = p_1 p_2$, and $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_\tau, \boldsymbol{b}_1, \boldsymbol{b}_2 \leftarrow \mathsf{CLS.Encrypt}(\mathsf{sk}, 0)$.

To encrypt a message $m \in \mathbb{Z}_Q$, one samples $(s_i)_{i=1}^\tau \xleftarrow{\$} \{0,1\}^\tau$ and $t_1, t_2 \leftarrow [0, 2^\nu)$ and outputs $\boldsymbol{c} = (m, m) + \sum_{i=0}^\tau s_i \cdot \boldsymbol{x}_i + t_1 \cdot \boldsymbol{b}_1 + t_2 \cdot \boldsymbol{b}_2$. The parameters are chosen so that $(\tau + 2^{\nu+1}) \cdot 2^\rho \cdot Q < 2^{\eta-2}$ to ensure that $\boldsymbol{c}$ decrypts correctly with $\mathsf{CLS.Decrypt}$. The $\boldsymbol{x}_i$'s, $\boldsymbol{b}_1$, and $\boldsymbol{b}_2$ are specially crafted (using

Table 1: Parameters for the CLS scheme for $\lambda = 128$ bits of security

Parameters	Q	η	ρ	τ	ν
Set-I	2^{256}	1536	1792	3328	142
Set-II	2^{256}	2194	2450	4645	142
Set-III	2^{256}	2706	2962	5669	142

rejection sampling) in order to apply the leftover hash lemma over lattices of Coron, Lepoint and Tibouchi [CLT13, Sec. 4]; this step gives conditions on ν and τ but is irrelevant for our purposes—we refer to [CLS14] for a rigorous description.

Practical Parameters. Some specific parameters (and implementation results) are proposed by Cheon et al. The parameters are chosen from their cryptanalysis survey [CLS14, Sec. 4] and aim at a security level of 128 bits. We recall these parameters in Table 1.

2.2 The Co-ACD Assumptions

Definition 1 (Co-ACD). *Let $n, Q, \eta, \rho \geq 1$ and denote π the uniform distribution over the η-bit prime integers. The Co-ACD distribution for a given $\boldsymbol{p} = (p_1, \ldots, p_n) \in \mathbb{Z}^n$ is the set of tuples $(e \cdot Q \bmod p_1, \ldots, e \cdot Q \bmod p_n)$ where $e \xleftarrow{\$} (-2^\rho, 2^\rho)$.*

- *The search-Co-ACD problem is: For a vector $\boldsymbol{p} \leftarrow \pi^n$ and given arbitrarily many samples from the Co-ACD distribution for $\boldsymbol{p}$, to compute $\boldsymbol{p}$.*
- *The decisional-Co-ACD problem is: For some fixed vector $\boldsymbol{p} \leftarrow \pi^n$ and given arbitrarily many samples from $\mathbb{Z}_{p_1} \times \cdots \times \mathbb{Z}_{p_n}$, to distinguish whether the samples are distributed uniformly or whether they are distributed as the Co-ACD distribution for $\boldsymbol{p}$.*

We sometimes use notation like $(\rho, \eta, n; Q)$-Co-dACD to mean the assumption that no polynomial-time adversary solve the decisional Co-ACD problem with these parameters. In [CLS14, Th. 1 and Th. 3], the authors prove that the (somewhat) additive homomorphic encryption schemes of Section 2.1 are semantically secure under the $(\rho, \eta, 2; Q)$-Co-dACD assumption when $Q < 2^{\eta - 3 - 2\lambda}$.

2.3 Background on Lattices

In this section, we recall some useful facts about lattices, orthogonal lattices and Coppersmith's technique. We refer to [MR09, NS01, Ngu10] for more details.

Lattices. A d-dimensional Euclidean lattice $L \subset \mathbb{Z}^t$ is the set of all integer linear combinations of some linearly independent vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d \in \mathbb{Z}^t$, and we write

$$L = \Big\{ \sum_{i \leq d} x_i \boldsymbol{b}_i : (x_i)_{i \leq d} \in \mathbb{Z}^d \Big\} = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d \,.$$

We denote $\mathrm{vol}(L)$ the volume of the lattice, defined as $\mathrm{vol}(L) = |\det(BB^t)^{1/2}|$ for any basis matrix $B = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$. (The determinant of a lattice is well-defined since it is independent of the choice of the basis). A theorem due to Minkowski bounds the length of the shortest vector in L in terms of $\mathrm{vol}(L)$.

Theorem 1 ([Ngu10, Cor. 3]). *Any d-dimensional lattice L in $\mathbb{Z}^t$ contains a nonzero vector $\boldsymbol{x}$ such that $\|\boldsymbol{x}\| \leq \sqrt{d} \cdot \mathrm{vol}(L)^{1/d}$.*

And in fact, in most lattices, that is close to the right order of magnitude not just for the length $\lambda_1(L)$ of the shortest vector, but for all *successive minima* $\lambda_i(L)_{1 \leq i \leq d}$: according to the *Gaussian heuristic*, one expects $\lambda_i(L) \approx \frac{d}{2\pi e} \mathrm{vol}(L)^{1/d}$ for all i.

Among all the bases of a lattice L, some are "better" than others, in the sense that they consist of shorter and "more orthogonal" vectors. Shortening the vectors of a lattice basis and making them more orthogonal is the purpose of lattice reduction algorithm. The LLL algorithm of Lenstra, Lenstra and Lovász [LLL82] is the best known such algorithm; it runs in polynomial time and returns a basis $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d)$ such that $\|\boldsymbol{b}_i\| \leq 2^{\chi \cdot d} \cdot \lambda_i(L)$ for some absolute constant χ. More generally, with other reduction algorithms such as BKZ, one can achieve better approximation factors.

Orthogonal Lattices. For any vectors $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{Z}^t$, we say that $\boldsymbol{u}$ and $\boldsymbol{v}$ are orthogonal if $\langle \boldsymbol{u}, \boldsymbol{v} \rangle = 0$, and we denote it $\boldsymbol{u} \perp \boldsymbol{v}$. For any lattice $L \subset \mathbb{Z}^t$, we denote by $L^\perp$ its orthogonal lattice, i.e. the set of vectors in $\mathbb{Z}^t$ orthogonal to the points in L: $L^\perp = \{\boldsymbol{v} \in \mathbb{Z}^t \mid \forall \boldsymbol{u} \in L, \langle \boldsymbol{u}, \boldsymbol{v} \rangle = 0\}$. Note that $\dim(L) + \dim(L^\perp) = t$. We have the following theorem [NS97]:

Theorem 2. *There exists an algorithm which, given any basis $\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d\}$ of a lattice L in $\mathbb{Z}^t$ of dimension d, outputs an LLL-reduced basis of the orthogonal lattice $L^\perp$, and whose running time is polynomial with respect to t, d and any upper bound on the bit-length of the $\|\boldsymbol{b}_j\|$'s.*

Remark 1. A simple algorithm for Theorem 2 consists in a single call to LLL: to compute an LLL-reduced basis $\{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{t-d}\}$ of the orthogonal lattice $L^\perp \subset \mathbb{Z}^t$ to $L = \mathbb{Z}\boldsymbol{b}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{b}_d \subset \mathbb{Z}^t$, one applies LLL to the lattice in $\mathbb{Z}^{d+t}$ generated by the rows of the following matrix:

$$\begin{pmatrix} \gamma \cdot b_{1,1} & \cdots & \gamma \cdot b_{d,1} & 1 & & 0 \\ \vdots & & \vdots & & \ddots & \\ \gamma \cdot b_{1,t} & \cdots & \gamma \cdot b_{d,t} & 0 & & 1 \end{pmatrix},$$

where $\boldsymbol{b}_i = (b_{i,j})_{j=1}^t$ and γ is a suitably large constant, and keeps only the t last coefficients of each resulting vector.

Coppersmith's Technique. In [Cop96, Cop97], Coppersmith presents a method based on lattice reduction to find small roots of univariate modular polynomials. In this paper, we use the more general formulation of his theorem due to May [May03, Th. 7].

Theorem 3 (Coppersmith). *Let N be an integer of unknown factorization, which has a divisor $b \geq N^\beta$. Let $f(x)$ be a univariate polynomial of degree δ. Then, we can find all solutions x_0 to the equation $f(x) \equiv 0 \bmod b$ which satisfy $|x_0| \leq N^{\beta^2/\delta}$ in time polynomial in $(\log N, \delta)$.*

3 Known Plaintext Attack against the CLS Scheme

Let $m_1, \ldots, m_t$ be t messages with their respective ciphertexts $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_t$, where $\boldsymbol{c}_i \leftarrow \mathsf{Encrypt}(\mathsf{sk}, m_i)$. Throughout this section, we assume that the first message m_1 is unknown while the other $(t-1)$ messages $m_2, \ldots, m_t$ are known, and we show that if t is large enough, m_1 can be recovered efficiently. This means that the symmetric-key CLS scheme is not one-way against known-message attacks

(not OW-KPA-secure), and as a direct corollary, it follows that the public-key CLS scheme is not one-way either.

Moreover, this also implies that we can solve the decisional Co-ACD problem efficiently for suitable parameters. We can either see this from the original security reduction for the symmetric-key [CLS14, Th. 1], or much more directly: if we have samples from a distribution which is either the Co-ACD distribution or uniformly random, we can use these samples to "encrypt" randomly chosen messages $m_1, \ldots, m_t$, and then apply our attack. It will recover the correct value of m_1 with significant probability if the distribution was Co-ACD, but returns a random element of $\mathbb{Z}_Q$ for the uniform distribution, so that we solve the decisional Co-ACD problem with significant advantage.

We present our attack in the sections below: we first present the attack in Section 3.1 in the case when $n = 2$, i.e. $N = p_1 p_2$, as in the original CLS encryption schemes. Then, we show in Section 3.2 how it generalizes naturally to higher values of n, thus breaking the decisional Co-ACD assumption for a wide range of parameters.

3.1 Message Recovery Using Known Plaintexts for $N = p_1 p_2$

Denote $\boldsymbol{c} = (\boldsymbol{c}_1, \ldots, \boldsymbol{c}_t)$. We can write for each $i \in [t]$:

$$\begin{cases} c_{i,1} = m_i + e_i \cdot Q + k_{i,1} \cdot p_1 \\ c_{i,2} = m_i + e_i \cdot Q + k_{i,2} \cdot p_2 \end{cases},$$

where $|e_i| < 2^\rho$ and $k_{i,1}$ (resp. $k_{i,2}$) is the quotient in the Euclidean division of $m_i + e_i \cdot Q$ by p_1 (resp. p_2). If we write $\boldsymbol{e} = (e_i)_{i\in[t]}$, $\boldsymbol{k}_j = (k_{i,j})_{i\in[t]}$ and $\boldsymbol{C}_j = (c_{i,j})_{i\in[t]}$ for $j = 1, 2$, we have:

$$\begin{cases} \boldsymbol{C}_1 = \boldsymbol{m} + \boldsymbol{e} \cdot Q + p_1 \cdot \boldsymbol{k}_1 \\ \boldsymbol{C}_2 = \boldsymbol{m} + \boldsymbol{e} \cdot Q + p_2 \cdot \boldsymbol{k}_2 \end{cases}. \tag{1}$$

In particular, this yields the following equation:

$$\boldsymbol{C}_1 - \boldsymbol{C}_2 = p_1 \cdot \boldsymbol{k}_1 - p_2 \cdot \boldsymbol{k}_2. \tag{2}$$

Since only $\boldsymbol{C}_1 - \boldsymbol{C}_2$ is known, Equation (2)) can be seen as a variant of the hidden subset sum problem as considered by Nguyen and Stern [NS99]. However, while in the hidden subset sum setting the hidden vectors are random independent binary vectors, in our case the unknown vectors $\boldsymbol{k}_1$ and $\boldsymbol{k}_2$ are nearly parallel and have entries of roughly $(\rho + \log Q - \eta)$ bits. As a result, it turns out that $\boldsymbol{k}_1$ and $\boldsymbol{k}_2$ cannot be obtained directly from the much shorter reduced basis of the lattice they generate, and therefore we do not know how to recover the secret primes p_1, p_2 from the ciphertext difference $\boldsymbol{C}_1 - \boldsymbol{C}_2$ alone. Nevertheless, we can still obtain the unknown message m_1 in two steps:

(1) Find a short vector $\boldsymbol{u}$ in the orthogonal lattice $L^\perp$ to the lattice $L = \mathbb{Z}(\boldsymbol{C}_1 - \boldsymbol{C}_2)$ generated by $\boldsymbol{C}_1 - \boldsymbol{C}_2$. If $\boldsymbol{u}$ is short enough, we get $\langle \boldsymbol{u}, \boldsymbol{k}_1 \rangle = \langle \boldsymbol{u}, \boldsymbol{k}_2 \rangle = 0$.
(2) Reducing the linear equation $\langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q - \boldsymbol{C}_1 \rangle = \langle \boldsymbol{u}, -p_1 \cdot \boldsymbol{k}_1 \rangle = -p_1 \langle \boldsymbol{u}, \boldsymbol{k}_1 \rangle = 0$ modulo Q, we eliminate $\boldsymbol{e}$, and recover the message from $\langle \boldsymbol{u}, \boldsymbol{m} - \boldsymbol{C}_1 \rangle \equiv 0 \bmod Q$.

For the first step, we use the algorithm of Nguyen and Stern to obtain a basis of the orthogonal lattice $L^\perp \subset \mathbb{Z}^t$ of rank $t - 1$, where $L = \mathbb{Z}(\boldsymbol{C}_1 - \boldsymbol{C}_2)$ (see Theorem 2). Let $\boldsymbol{u}$ be a vector in $L^\perp$. Then, the following holds:

$$\begin{aligned} \langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q - \boldsymbol{C}_1 \rangle &\equiv \langle \boldsymbol{u}, \boldsymbol{C}_1 - \boldsymbol{C}_1 \rangle = 0 \pmod{p_1}, \\ \langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q - \boldsymbol{C}_1 \rangle &\equiv \langle \boldsymbol{u}, \boldsymbol{C}_2 - \boldsymbol{C}_1 \rangle = 0 \pmod{p_2}. \end{aligned}$$

Thus, we get the following equation:

$$\langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q - \boldsymbol{C}_1 \rangle \equiv 0 \pmod{N}.$$

Now if $\|\boldsymbol{u}\|$ is less than $N/\|\boldsymbol{m} + \boldsymbol{e} \cdot Q - \boldsymbol{C}_1\| \approx 2^{2\eta - \rho - \log Q}$, then

$$\|\langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q - \boldsymbol{C}_1 \rangle\| \leq \|\boldsymbol{u}\| \cdot \|\boldsymbol{m} + \boldsymbol{e} \cdot Q - \boldsymbol{C}_1\| < N,$$

which implies that the inner product $\langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q - \boldsymbol{C}_1 \rangle$ is actually zero over the integers. Finally, $\boldsymbol{u}$ satisfies $\langle \boldsymbol{u}, \boldsymbol{k}_1 \rangle = 0$ from Eq. (1), and similarly we obtain that $\langle \boldsymbol{u}, \boldsymbol{k}_2 \rangle = 0$.

In the second step, we actually recover the message m_1. Using the vector $\boldsymbol{u} = (u_1, \ldots, u_t) \in \Lambda^\perp$ obtained in the first step, we have $\langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q - \boldsymbol{C}_1 \rangle = 0$. Viewing this equation modulo Q, we obtain

$$\langle \boldsymbol{u}, \boldsymbol{m} - \boldsymbol{C}_1 \rangle = 0 \pmod{Q}. \tag{3}$$

Solving the equation (3) modulo Q reveals m_1 completely as soon as $\gcd(u_1, Q) = 1$, which happens with significant probability $\phi(Q)/Q = \Omega(1/\log\log Q)$ if we assume that u_1 is randomly distributed modulo Q. More generally, we always obtain m_1 modulo $(Q/\gcd(u_1, Q))$, which gives $\gcd(u_1, Q)$ candidates for m_1 (this is usually small, and polynomially bounded on average for a random Q by [Bro01, Theorem 4.3]), and we can of course obtain more information on m_1 with a different, independent short vector $\boldsymbol{u}$ or with more known plaintexts, making recovery very fast in practice.

We now discuss the value of t needed to find a short enough vector $\boldsymbol{u}$. Since the volume of $L^\perp$ is $\mathrm{vol}(L^\perp) = \mathrm{vol}(L) = \|\boldsymbol{C}_1 - \boldsymbol{C}_2\| \approx 2^\eta$ and it has rank $t - 1$, Minkowski's theorem guarantees that it contains a vector $\boldsymbol{u}$ of length at most $\sqrt{t-1} \cdot \mathrm{vol}(L^\perp)^{1/(t-1)} \approx 2^{\eta/(t-1)}$. Such a vector is short enough to carry out our attack provided that:

$$\eta/(t-1) < 2\eta - \rho - \log Q \iff t > 1 + \frac{\eta}{2\eta - \rho - \log Q}.$$

Setting $t = 4$ is enough for all proposed parameters in [CLS14]. For such a small lattice dimension, it is straightforward to find the actual shortest vector of the lattice, and we can easily recover m_1 in practice in a fraction of a second, even accounting for occasional repetitions when more than one candidate is found.

We can also analyze the attack asymptotically as follows. For large lattice dimensions, a lattice reduction algorithm may not find the shortest vector of $L^\perp$, but only an approximation within a factor $2^{\chi \cdot (t-1)}$, where the value χ depends on the algorithm; we can achieve a given value of χ in time $2^{\Theta(1/\chi)}$ using BKZ-reduction with block size $\Theta(1/\chi)$. With such an algorithm, a short enough vector $\boldsymbol{u}$ will be found provided that:

$$\chi \cdot (t-1) + \frac{\eta}{t-1} < 2\eta - \rho - \log Q.$$

The left-hand side is minimal for $t - 1 = \sqrt{\eta/\chi}$, and is then equal to $2\sqrt{\chi\eta}$. Moreover, the right-hand side is a lower bound on the additive homomorphicity of the encryption scheme (denoted by $\log_2 A$ in [CLS14]), and should thus be at least as large as the security parameter λ for the scheme to be of interest. The condition to find $\boldsymbol{u}$ then becomes $\chi < \lambda^2/4\eta$. Thus, we obtain an attack with complexity $2^{\Omega(\eta/\lambda^2)}$, which means that our algorithm runs in provable polynomial time for parameters such that $\eta = \widetilde{O}(\lambda^2)$, and that we should have at least $\eta = \Omega(\lambda^3)$ to achieve λ bits of security, making the scheme quite inefficient.

More concretely, 128-bit security roughly corresponds to $2^\chi \approx 1.007$ [CN11, vdPS13]. Hence, a conservative choice of η for 128 bits of security should satisfy:

$$\eta \gtrsim \frac{128^2}{4 \cdot \log_2(1.007)} > 400{,}000,$$

making the scheme quite impractical!

3.2 Generalization to $n \geq 2$

The original CLS scheme is instantiated with a modulus $N = p_1 p_2$ which the product of two primes, for efficiency reasons. However, it can naturally be extended to the case when $N = p_1 \cdots p_n$ is the product of any number $n \geq 2$ of primes, in which case the security is reduced to the Co-ACD assumption with the corresponding n. It is clear as well, however, that our attack strategy extends to this case too.

Indeed, consider such a modulus $N = \prod_{j=1}^n p_j$ (which may be kept secret). Using the same notation as before, we have:

$$\boldsymbol{C}_j = \boldsymbol{m} + \boldsymbol{e} \cdot Q + p_j \cdot \boldsymbol{k}_j, \text{ for all } j \in [n].$$

Recall that we know the plaintexts $m_2, \ldots, m_t$ where $\boldsymbol{m} = (m_1, m_2, \ldots, m_t)$. Our goal is to find $m_1 \in \mathbb{Z}_Q$.

We first prove that a vector orthogonal to $\boldsymbol{C}_j - \boldsymbol{C}_1$ for all $j \in [n]$ is either large, or orthogonal to $\boldsymbol{k}_j$ for all $j \in [n]$.

Lemma 1. *Let $\boldsymbol{u} \in \mathbb{Z}^t$. If $\boldsymbol{u} \perp (\boldsymbol{C}_j - \boldsymbol{C}_1)$ for all $j \in [n]$, then it verifies one of the following condition:*

(1) $\boldsymbol{u} \perp \boldsymbol{k}_j$ for all $j \in [n]$;
(2) $\|\boldsymbol{u}\| \geq 2^{n(\eta-1)}/(Q \cdot 2^{\rho+1} \cdot t^{1/2})$.

Proof. Let $\boldsymbol{u} \in \mathbb{Z}^t$ such that $\boldsymbol{u} \perp (\boldsymbol{C}_j - \boldsymbol{C}_1)$ for all $j \in [n]$, and which does not verify condition (2). Now for all $j \in [n]$,

$$0 = \langle \boldsymbol{u}, \boldsymbol{C}_j - \boldsymbol{C}_1 \rangle = \langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q + p_j \cdot \boldsymbol{k}_j - \boldsymbol{C}_1 \rangle = \langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q - \boldsymbol{C}_1 \rangle + p_j \cdot \langle \boldsymbol{u}, \boldsymbol{k}_j \rangle.$$

In particular, $N = \prod_{j=1}^n p_j > 2^{n(\eta-1)}$ divides $\langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q - \boldsymbol{C}_1 \rangle$. Now the Cauchy-Schwarz inequality yields

$$|\langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q - \boldsymbol{C}_1 \rangle| \leq \|\boldsymbol{u}\| \cdot (\|\boldsymbol{m}\| + Q \cdot \|\boldsymbol{e}\| + \|\boldsymbol{C}_1\|) < \|\boldsymbol{u}\| \cdot Q \cdot 2^{\rho+1} \cdot t^{1/2} < 2^{n(\eta-1)},$$

which implies $\langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q - \boldsymbol{C}_1 \rangle = 0$, and thus $\langle \boldsymbol{u}, \boldsymbol{k}_j \rangle = 0$ for all $j \in [n]$. □

Let $L^\perp$ be the orthogonal lattice to the lattice $L = \mathbb{Z}(\boldsymbol{C}_2 - \boldsymbol{C}_1) \oplus \cdots \oplus \mathbb{Z}(\boldsymbol{C}_n - \boldsymbol{C}_1)$ generated by the vectors $\boldsymbol{C}_j - \boldsymbol{C}_1$. As before, we can use lattice reduction to find a short vector $\boldsymbol{u}$ in $L^\perp$, and by Lemma 1, if $\boldsymbol{u}$ is sufficiently short it must satisfy $\langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q - \boldsymbol{C}_1 \rangle = 0$. Reducing that linear relation modulo Q, we obtain:

$$\langle \boldsymbol{u}, \boldsymbol{m} - \boldsymbol{C}_1 \rangle = 0 \bmod Q,$$

which can be used to recover the message m_1, provided that $\gcd(u_1, Q) = 1$ (which again happens with significant probability).

As before, let us estimate the condition on t for such a short vector $\boldsymbol{u}$ to exist. Since $L^\perp$ is of rank $m = t - n + 1$ and volume $\mathrm{vol}(L^\perp) = \mathrm{vol}(L) \leq \prod_{i=2}^{n} \|\boldsymbol{C}_i - \boldsymbol{C}_1\| \approx 2^{(n-1)\eta}$, Minkowski's theorem ensures that it contains a vector of length at most $\sqrt{m} \cdot \mathrm{vol}(L^\perp)^{1/m} \approx 2^{(n-1)\eta/m}$. Taking logarithms and ignoring logarithmic factors, the condition can be written as:

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

Again, if we can find the shortest vector in a t-dimensional lattice for some t satisfying (4), we can break the CLS scheme (and the decisional Co-ACD assumption) for the corresponding choice of parameters. For the parameters suggested in [CLS14], where the authors take $\rho = (n-1)\eta + 2\lambda$, the required t is quite small: it suffices to choose $t \approx 3n$ if $2\lambda + \log Q < \eta/2$. Therefore, it is easy to break such parameters for small values of n.

More generally, we can mimic the asymptotic analysis of the previous section to take larger parameters into account. Lattice reduction will again approximate the shortest vector in $L^\perp$ within a factor $2^{\chi \cdot m}$ in time $2^{\Theta(1/\chi)}$, and the resulting vector is short enough when:

$$\chi \cdot m + \frac{(n-1)\eta}{m} < n(\eta - 1) - \log Q - \rho. \tag{5}$$

The left-hand side is minimal for $m = \sqrt{(n-1)\eta/\chi}$, in which case it evaluates to $2\sqrt{\chi \cdot (n-1)\eta}$. Moreover, the right-hand side is again a bound on additive homomorphicity, and should be taken at least as large as λ. Thus, the vector will be short enough for $\chi < \frac{\lambda^2}{4(n-1)\eta}$, hence an attack on the scheme (and the corresponding Co-ACD assumption) in time $2^{\Omega(n\eta/\lambda^2)}$. Therefore, $n\eta$ (the size in bits of the modulus N) should be chosen as $\Omega(\lambda^3)$ for λ bits of security.

And again, the numerical estimate for 128 bits of security, corresponding to $2^\chi \approx 1.007$, indicates that N should be chosen greater than 400,000 bits.

3.3 Experimental Results

We implemented our attack on the parameters proposed by [CLS14] (see Table 1), and on other sets of parameters for $n \geq 2$. The reduced basis for $L^\perp$ is computed using the Nguyen–Stern algorithm (cf. Remark 1), and we choose $\boldsymbol{u}$ among short enough vectors in the reduced basis such that $\gcd(u_1, Q)$ is minimal. As reported in Table 2, the attack takes *much less than a second* for $n = 2$, and under 40 seconds even for $n = 5$ and a much larger ρ. On average, the number of candidates for m_1 is always less than 2.

4 Ciphertext-Only Attack against the CLS Scheme

We now present a somewhat stronger attack against the symmetric-key CLS encryption scheme, which works without any known plaintext. We assume that we obtain the ciphertexts $\boldsymbol{c}_i \leftarrow \mathsf{Encrypt}(\mathsf{sk}, m_i)$ corresponding to t messages $m_1, \ldots, m_t$ that are unknown but *small*, and we show that all the m_i's can be recovered efficiently.

Combining this attack with the one from the previous section, this means that we can break the one-wayness of the symmetric-key CLS scheme without any known plaintexts, as long as we

Table 2: Known Plaintext Attack on the CLS scheme with message space $\mathbb{Z}_{2^{256}}$ using $(t-1)$ plaintext-ciphertext pairs (average value over 100 experiments using Sage [S+14] on a single 2.8Ghz Intel CPU).

(a) Attack against the proposed parameters claiming 128 bits of security

Parameters	t	Time in seconds	Success rate	Average # of candidates
Set-I	4	0.005s	100%	1.21
Set-II	4	0.006s	100%	1.52
Set-III	4	0.007s	100%	1.33

(b) Various parameters for $n \geq 2$ with $\eta = 1536$

n	2		3		4		5	
ρ	1792	2688	3328	4224	4864	5760	6400	7296
t	4	14	6	28	8	42	11	58
Time in seconds	0.005s	0.122s	0.027s	1.95s	0.081s	10.8s	0.22s	39.1s
Success rate	100%	100%	100%	95%	100%	95%	100%	92%
Average # of candidates	1.21	1	1.08	1	1.07	1	1.03	1

get a few ciphertexts associated with small messages (a very common situation in a homomorphic setting!).

From a technical standpoint, this stronger attack is still based on Nguyen–Stern orthogonal lattices, but uses the "doubly orthogonal" technique introduced in [NS97]. This makes the attack heuristic, in contrast with the one from Section 3, which is fully provable.

We present our attack in the sections below: we first present the attack in Section 4.1 in the case when $n = 2$, i.e. $N = p_1 p_2$, as in the original CLS encryption schemes. Then, we explain in Section 4.2 that it generalizes naturally to higher values of n.

4.1 (Small) Message Recovery Using Known Ciphertexts for $N = p_1 p_2$

We use the same notation as in Section 3.1. Our attack proceeds in two steps:

(1) Find $t-3$ short vectors $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{t-3}$ in the orthogonal lattice $L^\perp$ to the lattice $L = \mathbb{Z}\boldsymbol{C}_1 \oplus \mathbb{Z}\boldsymbol{C}_2$. If the $\boldsymbol{u}_i$ are short enough, we will get that $\langle \boldsymbol{u}_i, \boldsymbol{m} + \boldsymbol{e} \cdot Q \rangle = 0$.
(2) Rewriting $\langle \boldsymbol{u}_i, \boldsymbol{m} + \boldsymbol{e} \cdot Q \rangle = \langle \boldsymbol{u}_i, \boldsymbol{m} \rangle + Q \cdot \langle \boldsymbol{u}, \boldsymbol{e} \rangle = 0$ and reducing modulo Q, we get that $\langle \boldsymbol{u}_i, \boldsymbol{m} \rangle = 0 \bmod Q$. If $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{t-4}$ are short enough, the previous equation holds over $\mathbb{Z}$ and $\boldsymbol{m} \in (L')^\perp$ where $L' = \mathbb{Z}\boldsymbol{u}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{u}_{t-4}$. One should recover the small vector $\boldsymbol{m}$ as the shorter vector of $(L')^\perp$.

For the first step, we once again use the algorithm of Nguyen and Stern to obtain a basis $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{t-2}$ of $L^\perp \subset \mathbb{Z}^t$ of rank $t-2$. Similarly to Lemma 1, we have that a vector $\boldsymbol{u}_i$ orthogonal to both $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ is either large, or orthogonal to $\boldsymbol{k}_1, \boldsymbol{k}_2$ and $\boldsymbol{m} + \boldsymbol{e} \cdot Q$.

Lemma 2. *Let $\boldsymbol{u} \in \mathbb{Z}^t$. If $\boldsymbol{u} \perp \boldsymbol{C}_1$ and $\boldsymbol{u} \perp \boldsymbol{C}_2$, then it verifies one of the following condition:*

(1) $\boldsymbol{u} \perp (\boldsymbol{m} + \boldsymbol{e} \cdot Q)$, $\boldsymbol{u} \perp \boldsymbol{k}_1$ and $\boldsymbol{u} \perp \boldsymbol{k}_2$;

(2) $\|\boldsymbol{u}\| \geq 2^{2(\eta-1)}/(Q \cdot 2^{\rho+1} \cdot t^{1/2})$.

Proof. Let $\boldsymbol{u} \in \mathbb{Z}^t$ such that $\boldsymbol{u} \perp \boldsymbol{C}_1$ and $\boldsymbol{u} \perp \boldsymbol{C}_2$, and which does not verify condition (2). Now

$$\begin{cases} 0 = \langle \boldsymbol{u}, \boldsymbol{C}_1 \rangle = \langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q \rangle + p_1 \cdot \langle \boldsymbol{u}, \boldsymbol{k}_1 \rangle \\ 0 = \langle \boldsymbol{u}, \boldsymbol{C}_2 \rangle = \langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q \rangle + p_2 \cdot \langle \boldsymbol{u}, \boldsymbol{k}_2 \rangle \end{cases}.$$

In particular, $p = p_1 \cdot p_2 > 2^{2(\eta-1)}$ divides $\langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q \rangle$. Now the Cauchy–Schwarz inequality yields

$$|\langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q \rangle| \leq \|\boldsymbol{u}\| \cdot (\|\boldsymbol{m}\| + Q \cdot \|\boldsymbol{e}\|) < \|\boldsymbol{u}\| \cdot Q \cdot 2^{\rho+1} \cdot t^{1/2} < 2^{2(\eta-1)}\,,$$

which implies $\langle \boldsymbol{u}, \boldsymbol{m} + \boldsymbol{e} \cdot Q \rangle = 0$, and thus $\langle \boldsymbol{u}, \boldsymbol{k}_1 \rangle = \langle \boldsymbol{u}, \boldsymbol{k}_2 \rangle = 0$. □

In particular, if $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{t-3}$ do not verify condition (2) of Lemma 2, they are such that $\langle \boldsymbol{u}_i, \boldsymbol{m} + \boldsymbol{e} \cdot Q \rangle = 0$.

For the second step, we similarly prove that if a vector $\boldsymbol{u}$ is orthogonal to $\boldsymbol{m} + \boldsymbol{e} \cdot Q$, then it is either large or orthogonal to both $\boldsymbol{m}$ and $\boldsymbol{e}$.

Lemma 3. *Let $\boldsymbol{u} \in \mathbb{Z}^t$. If $\boldsymbol{u} \perp (\boldsymbol{m} + \boldsymbol{e} \cdot Q)$, then it verifies one of the following condition:*

(1) $\boldsymbol{u} \perp \boldsymbol{m}$ *and* $\boldsymbol{u} \perp \boldsymbol{e} \cdot Q$*;*
(2) $\|\boldsymbol{u}\| \geq Q/(2^{\mu} \cdot t^{1/2})$.

In particular, if $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{t-4}$ do not verify condition (2) of Lemma 3, then $\boldsymbol{m}, \boldsymbol{e}, \boldsymbol{k}_1$ and $\boldsymbol{k}_2$ are in $(L')^{\perp}$ where $L' = \mathbb{Z}\boldsymbol{u}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{u}_{t-4}$. By applying Nguyen and Stern technique, one can hope to recover $\boldsymbol{m}$ as the shortest vector of $(L')^{\perp}$.

We now discuss the conditions so that

(a) $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{t-3}$ do not verify condition (2) of Lemma 2,
(b) $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{t-4}$ do not verify condition (2) of Lemma 3.

Let us start with (a). For linearly independent $\boldsymbol{m} + \boldsymbol{e} \cdot Q$, $\boldsymbol{k}_1$ and $\boldsymbol{k}_2$, the first condition of Lemma 2 cannot hold for all $\boldsymbol{u}_k$ with $k \in [t-2]$ (for reasons of dimensions). In particular, the largest $\boldsymbol{u}_k$, say $\boldsymbol{u}_{t-2}$, must satisfy $\|\boldsymbol{u}_{t-2}\| \geq 2^{2(\eta-1)}/(Q \cdot 2^{\rho+1} \cdot t^{1/2})$. Now the other vectors form a lattice $L = \mathbb{Z}\boldsymbol{u}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{u}_{t-3}$ of rank $t-3$ and of volume

$$V = \mathrm{vol}(L) \approx \frac{\mathrm{vol}(\Lambda^{\perp})}{\|\boldsymbol{u}_{t-2}\|} \leq \frac{\|\boldsymbol{C}_1\| \cdot \|\boldsymbol{C}_2\|}{2^{2(\eta-1)}/(Q \cdot 2^{\rho+1} \cdot t^{1/2})} \leq Q \cdot 2^{\rho+3} \cdot t^{3/2}\,.$$

Heuristically, we can expect L to behave as a random lattice; assuming the Gaussian heuristic, we should have $\|\boldsymbol{u}_k\| \approx \sqrt{t-3} \cdot V^{1/(t-3)})$. Thus, the condition for all the $\boldsymbol{u}_k$'s to be orthogonal to $\boldsymbol{k}_1$, $\boldsymbol{k}_2$ and $\boldsymbol{m} + \boldsymbol{e} \cdot Q$ becomes

$$t^{-1/2} \cdot 2^{-2} \cdot \left(Q \cdot 2^{\rho+3} \cdot t^{3/2}\right)^{1+1/(t-3)} \ll 2^{2\cdot(\eta-1)} \leq 2^{2\cdot\eta}\,.$$

Taking logarithms and ignoring logarithmic factors, this means:

$$t \gtrsim 3 + \frac{\rho + 3 + \log Q}{2\eta - \log Q - \rho - 3} = 3 + \frac{\alpha}{1-\alpha} \quad \text{where} \quad \alpha = \frac{\rho + 3 + \log Q}{2\eta}. \tag{6}$$

In the following, we assume that condition (6) is satisfied; therefore the vectors $\boldsymbol{m} + \boldsymbol{e} \cdot Q$, $\boldsymbol{k}_1$ and $\boldsymbol{k}_2$ belong to $L^\perp$.

Next, let us focus on (b). Similarly, for linearly independent $\boldsymbol{m}$, $\boldsymbol{e}$, $\boldsymbol{k}_1$ and $\boldsymbol{k}_2$, condition (2) of Lemma 3 cannot hold for all $k \in [t-3]$, and therefore $\|\boldsymbol{u}_{t-3}\| \geq Q/(2^\mu \cdot t^{1/2})$. Now we want to select t large enough so that all the $\|\boldsymbol{u}_k\|$ for $k \leq t-4$ verifies condition (1) of Lemma 3. We have that

$$\|\boldsymbol{u}_k\| = \mathcal{O}(t^2 \cdot Q^{1/(t-3)} \cdot 2^{(\rho+3)/(t-3)}),$$

so the $\boldsymbol{u}_k$'s do not verify condition (2) of Lemma 3 (and therefore verify condition (1)) when

$$t^{5/2} \cdot Q^{1/(t-3)} \cdot 2^{(\rho+3)/(t-3)} \cdot 2^\mu \ll Q\,.$$

Taking logarithms and ignoring logarithmic factors, this means:

$$t \gtrsim 3 + \frac{\log Q + \rho + 3}{\log Q - \mu}\,. \tag{7}$$

Finally, assuming condition (7) is satisfied, $\boldsymbol{m}$ is a really short vector (of norm $\approx 2^\mu \cdot t^{1/2}$) orthogonal to $\boldsymbol{u}_k$ for all $k \in [t-4]$. It follows that one should recover $\boldsymbol{m}$ as the first vector of the reduced basis of $(L')^\perp$, at least in the case of small lattice dimensions. Our experiments, presented in Section 4.3, show that this is condition is well verified in practice.

Moreover, one can carry out an asymptotic analysis as in Section 3 to take larger lattice dimensions into account. The computations are very similar, but due to the heuristic nature of the present attack, they are less meaningful.

4.2 Generalization to $n \geq 2$

Once again, our technique generalizes directly to $n \geq 2$. The steps of the generalized attack are similar:

(1) Find $t-n-1$ short vectors $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{t-n-1}$ in the orthogonal lattice $L^\perp$ to the lattice $L = \mathbb{Z}\boldsymbol{C}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{C}_n$. If the $\boldsymbol{u}_i$ are short enough, we will get that $\langle \boldsymbol{u}_i, \boldsymbol{m} + \boldsymbol{e} \cdot Q \rangle = 0$ (and $\langle \boldsymbol{u}_i, \boldsymbol{k}_j \rangle = 0$ for all $j \in [n]$).

(2) Rewriting $\langle \boldsymbol{u}_i, \boldsymbol{m} + \boldsymbol{e} \cdot Q \rangle = \langle \boldsymbol{u}_i, \boldsymbol{m} \rangle + Q \cdot \langle \boldsymbol{u}, \boldsymbol{e} \rangle = 0$ and reducing modulo Q, we get that $\langle \boldsymbol{u}_i, \boldsymbol{m} \rangle = 0 \bmod Q$. If $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{t-n-2}$ are short enough, the previous equation holds over $\mathbb{Z}$ and $\boldsymbol{m} \in (L')^\perp$ where $L' = \mathbb{Z}\boldsymbol{u}_1 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{u}_{t-n-2}$. One should recover the small vector $\boldsymbol{m}$ as the shorter vector of $(L')^\perp$.

Condition (7) becomes:

$$t \gtrsim n + 1 + \frac{\log Q + \rho + n + 1}{\log Q - \mu}\,.$$

4.3 Experimental Results

We ran our attacks against the parameters of Table 1. Once again, our attack is really efficient; it amounts to applying LLL twice (cf. Remark 1) and runs in a matter of seconds. Results are collected in Table 3.

Table 3: Attack of Section 4.1 on the CLS scheme with message space $\mathbb{Z}_{2^{256}}$ (average value over 500 experiments using Sage [S+14] on a single 3.4Ghz Intel Core i7 CPU).

(a) **Parameter Set-I**

μ	0	16	32	64	128	192	224
Minimal t from Eq. (7)	12	12	13	14	20	36	68
Minimal t in practice	12	12	13	14	20	39	80
Running time (in seconds)	0.16s	0.16s	0.21s	0.28s	1.10s	13.9s	169s
Success rate	100%						

(b) **Parameter Set-II**

μ	0	16	32	64	128	192	224
Minimal t from Eq. (7)	14	15	16	18	25	46	88
Minimal t in practice	14	15	16	18	25	47	98
Running time (in seconds)	0.38s	0.49s	0.62s	0.99s	3.65s	37.1s	521s
Success rate	100%						72.8%

5 Breaking the Search Co-ACD Assumption

In this section, we break the search Co-ACD assumption when $N = \prod_i p_i$ and Q are known (as in the public-key CLS scheme): given a few samples $\{(e_i \cdot Q \bmod p_1, \ldots, e_i \cdot Q \bmod p_n)\}_i$ from the Co-ACD distribution, we show that one can recover the p_i's in heuristic polynomial time, at least for certain ranges of parameters. In particular, in the public-key CLS encryption scheme, the private key can be recovered from the public key alone!

5.1 Description of the Attack

For simplicity, we first consider the case $n = 2$ (as in the CLS scheme). We use the same notation as in Section 4 with $\boldsymbol{m} = 0$, and assume that $N = p_1 p_2$ is known. Hence, we have that

$$(\boldsymbol{C}_1 - \boldsymbol{e} \cdot Q) \cdot (\boldsymbol{C}_2 - \boldsymbol{e} \cdot Q) = \boldsymbol{0} \bmod N\,, \tag{8}$$

where the multiplication $\cdot$ is done componentwise. We start from the following equation:

$$\boldsymbol{e} \cdot Q = (\boldsymbol{C}_1 - \boldsymbol{C}_2) \cdot \bar{p}_1 + \boldsymbol{C}_2 \pmod N\,, \tag{9}$$

where $\bar{p}_1 = p_2 \cdot (p_2^{-1} \bmod p_1) \bmod N$ is the first CRT coefficient for (p_1, p_2). Multiplying $Q^{-1} \bmod N$, we obtain

$$\boldsymbol{e} = (\boldsymbol{C}_1 - \boldsymbol{C}_2) \cdot \bar{p}_1 Q^{-1} + \boldsymbol{C}_2 \cdot Q^{-1} \pmod N.$$

Similar to the lattice used against the ACD problem in [DGHV10], considering the above equation, we construct a lattice L generated by the rows of the following $(t+2) \times (t+1)$ matrix:

$$\begin{pmatrix} \boldsymbol{C}_1 - \boldsymbol{C}_2 & 0 \\ \boldsymbol{C}_2 \cdot Q^{-1} \bmod N & 2^\rho \\ N \cdot I_{t \times t} & \boldsymbol{0} \end{pmatrix}$$

The lattice L contains the following short distinguished vectors:

$$\boldsymbol{v}_1 = (\boldsymbol{C}_1 - \boldsymbol{C}_2, 0) \text{ and } \boldsymbol{v}_2 = (\boldsymbol{e}, 2^\rho), \tag{10}$$

of respective norms $\|\boldsymbol{v}_1\| \approx 2^\eta$ and $\|\boldsymbol{v}_2\| \approx 2^\rho$, and when t is large enough, we expect those vectors to be much shorter than other independent vectors in L (see the discussion below). As a result, if $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are the first two vectors of a reduced basis of the lattice L, we expect to have, up to some explicit sign change, $\boldsymbol{v}_1 = \boldsymbol{x}_1$ and:

$$(\boldsymbol{e}, 2^\rho) = \boldsymbol{v}_2 = \boldsymbol{x}_2 + \alpha \boldsymbol{x}_1 \tag{11}$$

for some unknown integer coefficient $\alpha \in \mathbb{Z}$.

Now, plugging the previous equality into Equation (8) and considering the first components of the corresponding vectors, we obtain:

$$\big(c_{1,1} - Q(x_{2,1} + \alpha \cdot x_{1,1})\big) \cdot \big(c_{2,1} - Q(x_{2,1} + \alpha \cdot x_{1,1})\big) = 0 \bmod N\,.$$

This yields a univariate quadratic equation modulo N which admits α as a solution. Moreover, that solution α is short, in the sense that

$$|\alpha| = \frac{\|\boldsymbol{v}_2 - \boldsymbol{x}_2\|}{\|\boldsymbol{x}_1\|} \leq \frac{\|\boldsymbol{v}_2\| + \|\boldsymbol{x}_2\|}{\|\boldsymbol{x}_1\|} \lesssim 2^{1+\rho-\eta} < \sqrt{N}.$$

As a result, we can use (the original, univariate version of) Coppersmith's theorem [Cop96, Cop97] to solve this equation in polynomial time, obtain α, and recover $\boldsymbol{e}$ from (11). It is then straightforward to factor N by computing $\gcd(\boldsymbol{C}_1 - \boldsymbol{e} \cdot Q, N)$ componentwise.

Finally, we analyze how t should be chosen. Since our target vector $\boldsymbol{v}_2$ is much longer than the shortest vector $\boldsymbol{v}_1$, the best we can hope is that the second shortest vector in L is $\boldsymbol{v}_2$ (modulo $\boldsymbol{v}_1$). Using $\det L = 2^\rho \cdot N^{t-1} \approx 2^{\rho+2\eta(t-1)}$, we expect that the length of the second shortest vector in L to be $\ell = (2^{\rho+2\eta(t-1)-\eta})^{1/t} = 2^{2\eta+\frac{\rho-3\eta}{t}}$. Thus, we can expect to find $\boldsymbol{v}_2$ in L (modulo $\boldsymbol{v}_1$) if $\|\boldsymbol{v}_2\| < \ell$. This yields the following condition on t:

$$t > \frac{3\eta - \rho}{2\eta - \rho}. \tag{12}$$

By choosing t satisfying (12), the above described attack finds factors of N easily which is verified by the experiments.

5.2 Extension to $n \geq 3$

In this section, we extend the attack against search Co-ACD assumption to the case $n \geq 3$. Unlike the case $n = 2$, we will see that this extended attack is only applicable in a certain range for ρ, but it always breaks non trivial instances of the search Co-ACD problem.

Similar to the case $n = 2$, we start from the following equation:

$$\boldsymbol{e} \cdot Q = (\boldsymbol{C}_1 - \boldsymbol{C}_n) \cdot \bar{p}_1 + \cdots + (\boldsymbol{C}_{n-1} - \boldsymbol{C}_n) \cdot \bar{p}_{n-1} + \boldsymbol{C}_n \pmod N\,, \tag{13}$$

where the $\bar{p}_i$'s are the CRT coefficients $\bar{p}_i = \frac{N}{p_i} \cdot \big(\frac{p_i}{N} \bmod p_i\big)$. Multiplying $Q^{-1} \bmod N$, we again get

$$\boldsymbol{e} \cdot Q = (\boldsymbol{C}_1 - \boldsymbol{C}_n) \cdot \bar{p}_1 Q^{-1} + \cdots + (\boldsymbol{C}_{n-1} - \boldsymbol{C}_n) \cdot \bar{p}_{n-1} Q^{-1} + \boldsymbol{C}_n \cdot Q^{-1} \pmod N.$$

Therefore, if we consider the lattice L generated by the rows of the following matrix:

$$\begin{pmatrix} \boldsymbol{C}_1 - \boldsymbol{C}_n & 0 \\ \vdots & \vdots \\ \boldsymbol{C}_{n-1} - \boldsymbol{C}_n & 0 \\ \boldsymbol{C}_n \cdot Q^{-1} \bmod N & 2^\rho \\ N \cdot I_{t\times t} & \boldsymbol{0} \end{pmatrix}$$

it is full rank and contains the following short distinguished vectors: $\boldsymbol{v}_i = (\boldsymbol{C}_i - \boldsymbol{C}_n, 0)$ for $i = 1, \ldots, n-1$, which are all of norm $\approx 2^\eta$, and $\boldsymbol{v}_n = (\boldsymbol{e}, 2^\rho)$ of norm $\approx 2^\rho$. With high probability, these vectors are linearly independent, and when t is large enough, we expect them to be much shorter than other independent vectors in the lattice (see the discussion below).

As a result, and since $\boldsymbol{v}_n$ is much longer than the $\boldsymbol{v}_i$'s for $i < n$, applying lattice reduction to L should yield a reduced basis $(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{t+1})$ such that $\bigoplus_{i=1}^r \mathbb{Z}\boldsymbol{x}_i = \bigoplus_{i=1}^r \mathbb{Z}\boldsymbol{v}_i$ for $r = n-1$ and $r = n$. In particular, $(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{n-1}, \boldsymbol{x}_n, \ldots, \boldsymbol{x}_{t+1})$ is a basis of L, and writing $\boldsymbol{v}_n$ over that basis yields:

$$(\boldsymbol{e}, 2^\rho) = \boldsymbol{v}_n = \alpha \boldsymbol{v}_1 + \boldsymbol{y}$$

for some $\alpha \in \mathbb{Z}$ and $\boldsymbol{y} \in \mathbb{Z}\boldsymbol{v}_2 \oplus \cdots \oplus \mathbb{Z}\boldsymbol{v}_{n-1} \oplus \mathbb{Z}\boldsymbol{x}_n$. Plugging that relation into Equation (13) gives:

$$(\alpha \boldsymbol{v}_1 + \boldsymbol{y}) \cdot Q \equiv \bar{p}_1 \boldsymbol{v}_1 + \cdots + \bar{p}_{n-1} \boldsymbol{v}_{n-1} + \boldsymbol{w} \pmod N$$

where $\boldsymbol{w} = (\boldsymbol{C}_n, 2^\rho Q)$. Now choose a vector $\boldsymbol{u} \in \mathbb{Z}^{t+1}$ orthogonal to $\boldsymbol{v}_2, \ldots, \boldsymbol{v}_{n-1}, \boldsymbol{x}_n$ but not to $\boldsymbol{v}_1$ modulo N (such a vector exists with overwhelming probability, and when it does, it can be found in deterministic polynomial time using the Nguyen–Stern algorithm [NS97]). Taking the inner product with $\boldsymbol{u}$ yields:

$$Q\alpha \cdot \langle \boldsymbol{v}_1, \boldsymbol{u} \rangle + 0 \equiv \bar{p}_1 \langle \boldsymbol{v}_1, \boldsymbol{u} \rangle + 0 + \cdots + 0 + \langle \boldsymbol{w}, \boldsymbol{u} \rangle \pmod N,$$

or equivalently:

$$\bar{p}_1 \equiv Q\alpha + \omega \pmod N \quad \text{where} \quad \omega = -\frac{\langle \boldsymbol{w}, \boldsymbol{u} \rangle}{\langle \boldsymbol{v}_1, \boldsymbol{u} \rangle} \bmod N. \tag{14}$$

Moreover, α is still small compared to N, of size about $\rho - \eta$ bits. Therefore, we can proceed as before and deduce a polynomial relation from (14) so as to apply Coppersmith's theorem to recover α. We propose two ways of doing so. Note that once α is found, we obtain a non trivial factor of N straight away by computing $\gcd(Q\alpha + \omega, N) = N/p_1$.

One first approach to computing α is to observe that $\bar{p}_1$ is an idempotent element of $\mathbb{Z}_N$: it satisfies $\bar{p}_1^2 \equiv \bar{p}_1 \bmod N$. It follows that α is a root of the quadratic polynomial $F_2(X) = (Q \cdot X + \omega)^2 - (Q \cdot X + \omega)$ modulo N. It is thus possible to compute α in polynomial time using Coppersmith's theorem when $2^{\rho-\eta} < \sqrt{N} \approx 2^{n\eta/2}$, i.e. $\rho < \frac{n+2}{2} \cdot \eta$. Since we already know that $\rho > (n-1)\eta$ for security, that condition is only non trivial for $n = 2$ (providing a slightly different formulation of the attack from the previous section) and $n = 3$ (in which case we can break parameters $\rho < 5\eta/2$).

A second approach is to see that Equation (14) implies:

$$Q\alpha + \omega \equiv 0 \pmod{N/p_1}.$$

Therefore, α is a small root of the linear polynomial $F_1(X) = Q \cdot X + \omega$ modulo some large unknown factor of N of size $\approx N^{1-1/n}$. Alexander May's extension of Coppersmith's theorem guarantees that we can then recover α in deterministic polynomial time provided that $2^{\rho-\eta} < N^{(1-1/n)^2} \approx 2^{(n-2+1/n)\eta}$, i.e. $\rho < (n-1+1/n)\eta$. That condition is always non trivial, and thus we obtain an attack for all values of n. For $n = 3$, however, the previous approach should be preferred as it gives a better bound for ρ ($5\eta/2$ instead of $7\eta/3$).

Finally, let us evaluate the condition on t for the attack to succeed. As before, the condition says that the n-th minimum of the lattice L should be at least 2^ρ, while the first $n-1$ minima are at most 2^η. The volume of L is $\mathrm{vol}(L) = 2^\rho \cdot N^{t-n+1}$, and the expected n-th minimum is roughly $\ell = \left(\mathrm{vol}(L)/2^{(n-1)\eta}\right)^{1/(t+1-(n-1))}$. Thus, the condition can be written as: $(t-n+2)\cdot\rho < \rho + (t-n+1)\cdot n\eta - (n-1)\eta$, or equivalently:

$$t > \frac{(n+1)\eta - \rho}{n\eta - \rho} \cdot (n-1),$$

which is a direct generalization of Condition (12). For $n \geq 4$, since our best attack only works for $\rho < (n-1+1/n)\eta$, this condition simplifies to $t > \frac{(n+1)-(n-1+1/n)}{n-(n-1+1/n)}(n-1) = 2n-1$, i.e. $t \geq 2n$.

5.3 Experimental Results

We have implemented the attack of Section 5.1 in Sage. Timings are reported in Table 4. The initial lattice reduction step is very fast, and the Coppersmith computation, where most of the CPU time is spent, also takes on the order of seconds at most for practically all parameters we tested (despite the fact that Sage's `small_roots` command is relatively poorly optimized compared to more recent implementation efforts such as [BCF+14]).

We also implemented the attack for larger n, and found for example that N can be factored in a few seconds with only 5 samples for $(n, \eta, \rho) = (3, 1000, 2300)$.

ρ	1792	2192	2592	2792	2892	2992
Minimal t from Eq. (12)	3	3	5	7	10	21
Minimal t in practice	3	3	5	7	10	22
Running time of the attack (in seconds)	0.31s	0.26s	1.07s	1.07s	17.3s	1886s
Success rate	100%				99%	86%

(a) $\eta = 1536$ ($\rho = 1792$ for 128 bits of security)

ρ	2450	2950	3450	3700	3950	4200
Minimal t from Eq. (12)	3	3	4	5	7	13
Minimal t in practice	3	3	4	5	7	14
Running time of the attack (in seconds)	0.57s	0.55s	0.41s	2.0s	2.1s	203s
Success rate	100%					

(b) $\eta = 2194$ ($\rho = 2450$ for 128 bits of security)

Table 4: Attack of Section 5.1 on the search Co-ACD assumption with $Q = 2^{256}$ (average value over 100 experiments using Sage [S+14] on a single 2.8Ghz Intel CPU).

Acknowledgments

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

References

[BCF+14] Jingguo Bi, Jean-Sébastien Coron, Jean-Charles Faugère, Phong Q. Nguyen, Guénaël Renault, and Rina Zeitoun. Rounding and chaining LLL: finding faster small roots of univariate polynomial congruences. In Hugo Krawczyk, editor, *PKC*, volume 8383 of *LNCS*, pages 185–202. Springer, 2014.

[Bro01] Kevin A. Broughan. The gcd-sum function. *Journal of Integer Sequences*, 4, 2001. Article 01.2.2.

[CCK+13] Jung Hee Cheon, Jean-Sébastien Coron, Jinsu Kim, MoonSung Lee, Tancrède Lepoint, Mehdi Tibouchi, and Aaram Yun. Batch fully homomorphic encryption over the integers. In Thomas Johansson and PhongQ. Nguyen, editors, *EUROCRYPT*, volume 7881 of *LNCS*, pages 315–335. Springer Berlin Heidelberg, 2013.

[CH12] Henry Cohn and Nadia Heninger. Approximate common divisors via lattices. In *ANTS X*, 2012.

[CLS14] Jung Hee Cheon, Hyung Tae Lee, and Jae Hong Seo. A new additive homomorphic encryption based on the co-ACD problem. In Gail-Joon Ahn, Moti Yung, and Ninghui Li, editors, *ACM CCS*, pages 287–298. ACM, 2014.

[CLT13] Jean-Sébastien Coron, Tancrède Lepoint, and Mehdi Tibouchi. Practical multilinear maps over the integers. In Ran Canetti and Juan A. Garay, editors, *CRYPTO, Part I*, volume 8042 of *LNCS*, pages 476–493. Springer, 2013.

[CLT14] Jean-Sébastien Coron, Tancrède Lepoint, and Mehdi Tibouchi. Scale-invariant fully homomorphic encryption over the integers. In Hugo Krawczyk, editor, *PKC*, volume 8383 of *LNCS*, pages 311–328. Springer Berlin Heidelberg, 2014.

[CN11] Yuanmi Chen and Phong Q. Nguyen. BKZ 2.0: Better lattice security estimates. In *ASIACRYPT*, pages 1–20, 2011.

[CN12] Yuanmi Chen and Phong Q. Nguyen. Faster algorithms for approximate common divisors: Breaking fully-homomorphic-encryption challenges over the integers. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT*, volume 7237, pages 502–519. Springer, 2012.

[CNT10] Jean-Sébastien Coron, David Naccache, and Mehdi Tibouchi. Fault attacks against EMV signatures. In Josef Pieprzyk, editor, *CT-RSA*, volume 5985 of *LNCS*, pages 208–220. Springer, 2010.

[CNT12] Jean-Sébastien Coron, David Naccache, and Mehdi Tibouchi. Public key compression and modulus switching for fully homomorphic encryption over the integers. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT*, volume 7237 of *Lecture Notes in Computer Science*, pages 446–464. Springer, 2012.

[Cop96] Don Coppersmith. Finding a small root of a univariate modular equation. In Ueli M. Maurer, editor, *EUROCRYPT*, volume 1070 of *LNCS*, pages 155–165. Springer, 1996.

[Cop97] Don Coppersmith. Small solutions to polynomial equations, and low exponent RSA vulnerabilities. *J. Cryptology*, 10(4):233–260, 1997.

[DGHV10] Marten van Dijk, Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. Fully homomorphic encryption over the integers. In Henri Gilbert, editor, *EUROCRYPT*, volume 6110 of *LNCS*, pages 24–43. Springer Berlin / Heidelberg, 2010.

[How01] Nick Howgrave-Graham. Approximate integer common divisors. In Joseph H. Silverman, editor, *CaLC*, volume 2146 of *LNCS*, pages 51–66. Springer, 2001.

[IOM97] Kouichi Itoh, Eiji Okamoto, and Masahiro Mambo. Proposal of a fast public key cryptosystem. In Carlisle Adams and Mike Just, editors, *SAC*, pages 224–230, 1997.

[JL13] Marc Joye and Benoît Libert. Efficient cryptosystems from 2^k-th power residue symbols. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT*, volume 7881 of *LNCS*, pages 76–92. Springer, 2013.

[LLL82] Arjen K. Lenstra, Hendrik W. Lenstra Jr., and László Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261(4):515–534, 1982.

[LT15] Tancrède Lepoint and Mehdi Tibouchi. Cryptanalysis of a (somewhat) additively homomorphic encryption scheme used in PIR. In *WAHC*, 2015.

[May03] Alexander May. *New RSA Vulnerabilities Using Lattice Reduction Methods*. PhD thesis, University of Paderborn, 2003. Available from `http://www.cs.uni-paderborn.de/uploads/tx_sibibtex/bp.pdf`.

[MR09] Daniele Micciancio and Oded Regev. Lattice-based cryptography. In Daniel J. Bernstein, Johannes Buchmann, and Erik Dahmen, editors, *Post-Quantum Cryptography*, pages 147–191. Springer Berlin Heidelberg, 2009.

[Ngu10] Phong Q. Nguyen. Hermite's constant and lattice algorithms. In Phong Q. Nguyen and Brigitte Vallée, editors, *The LLL Algorithm*, Information Security and Cryptography, pages 19–69. Springer, 2010.

[NLV11] Michael Naehrig, Kristin E. Lauter, and Vinod Vaikuntanathan. Can homomorphic encryption be practical? In Christian Cachin and Thomas Ristenpart, editors, *ACM CCSW*, pages 113–124. ACM, 2011.

[NS97] Phong Q. Nguyen and Jacques Stern. Merkle-Hellman revisited: A cryptoanalysis of the Qu-Vanstone cryptosystem based on group factorizations. In Burton S. Kaliski Jr., editor, *CRYPTO*, volume 1294 of *LNCS*, pages 198–212. Springer, 1997.

[NS98] Phong Q. Nguyen and Jacques Stern. Cryptanalysis of a fast public key cryptosystem presented at SAC '97. In Stafford E. Tavares and Henk Meijer, editors, *SAC*, volume 1556 of *LNCS*, pages 213–218. Springer, 1998.

[NS99] Phong Nguyen and Jacques Stern. The hardness of the hidden subset sum problem and its cryptographic implications. In Michael Wiener, editor, *CRYPTO*, volume 1666 of *LNCS*, pages 31–46. Springer Berlin Heidelberg, 1999.

[NS01] Phong Q. Nguyen and Jacques Stern. The two faces of lattices in cryptology. In Joseph H. Silverman, editor, *CaLC*, volume 2146 of *LNCS*, pages 146–180. Springer, 2001.

[NT12] Phong Q. Nguyen and Mehdi Tibouchi. Lattice-based fault attacks on signatures. In Marc Joye and Michael Tunstall, editors, *Fault Analysis in Cryptography*, Information Security and Cryptography, pages 201–220. Springer, 2012.

[Pai99] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In Jacques Stern, editor, *EUROCRYPT*, volume 1592 of *LNCS*, pages 223–238. Springer, 1999.

[QV94] Minghua Qu and Scott A. Vanstone. The knapsack problem in cryptography. In *Finite Fields: Theory, Applications and Algorithms*, volume 168 of *Contemporary Mathematics*, pages 291–308. AMS, 1994.

[S+14] W. A. Stein et al. *Sage Mathematics Software (Version 6.4)*. The Sage Development Team, 2014. `http://www.sagemath.org`.

[vdPS13] Joop van de Pol and Nigel P. Smart. Estimating key sizes for high dimensional lattice-based systems. In Martijn Stam, editor, *IMACC*, volume 8308 of *LNCS*, pages 290–303. Springer, 2013.