Multi-Bit Cryptosystems based on Lattice Problems

Akinori Kawachi* Keisuke Tanaka* Keita Xagawa*

Abstract

We propose multi-bit versions of several single-bit cryptosystems based on lattice problems, the error-free version of the Ajtai-Dwork cryptosystem by Goldreich, Goldwasser, and Halevi [CRYPTO '97], the Regev cryptosystems [JACM 2004 and STOC 2005], and the Ajtai cryptosystem [STOC 2005]. We develop a universal technique derived from a general structure behind them for constructing their multi-bit versions without increase in the size of ciphertexts. By evaluating the trade-off between the decryption errors and the hardness of underlying lattice problems, it is shown that our multi-bit versions encrypt $O(\log n)$-bit plaintexts into ciphertexts of the same length as the original ones with reasonable sacrifices of the hardness of the underlying lattice problems. Our technique also reveals an algebraic property, named *pseudohomomorphism*, of the lattice-based cryptosystems.

Keyword: multi-bit public-key cryptosystems, lattice problems, pseudohomomorphism.

1 Introduction

Lattice-Based Cryptosystems. The lattice-based cryptosystems have been well-studied since Ajtai's seminal result [2] on a one-way function based on the worst-case hardness of lattice problems, which initiated the cryptographic use of lattice problems. Ajtai and Dwork first succeeded to construct public-key cryptosystems [6] based on the unique shortest vector problem (uSVP). After their results, a number of lattice-based cryptosystems have been proposed in the last decade by using cryptographic advantages of lattice problems [13, 10, 34, 4, 35].

We can roughly classify the lattice-based cryptosystems into two types: (A) those who are efficient on the size of their keys and ciphertexts and the speed of encryption/decryption procedures, but have no security proofs based on the hardness of well-known lattice problems, and (B) those who have security proofs based on the lattice problems but are inefficient.

For example, the GGH cryptosystem [14], NTRU [19] and their improvements [24, 31, 29, 18] belong to the type (A). These are efficient multi-bit cryptosystems related to lattices, but it is unknown whether their security is based on the hardness of well-known lattice problems. Actually, a few papers reported security issues of cryptosystems in this type [28, 11].

On the other hand, those in the type (B) have security proofs based on well-known lattice problems such as uSVP, the shortest vector problem (SVP) and the shortest linearly independent vectors problem (SIVP) [6, 34, 35]. (See Appendix E for their definitions and computational complexity.) In particular, the security of these cryptosystems can be guaranteed by the worst-case hardness of the lattice problems, i.e., breaking the cryptosystems on average is at least as hard as solving the lattice problems in the worst case. This

*Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, W8-55, 2-12-1 Ookayama Meguro-ku, Tokyo 152-8552, Japan. {kawachi,keisuke,xagawa5}@is.titech.ac.jp

attractive property of the average-case/worst-case connection has been also studied from a theoretical point of view [2, 27, 25, 32].

Aside from the interesting property, such cryptosystems generally have longer keys and ciphertexts than those of the cryptosystems in the type (A). To set their size practically reasonable, their security parameters must be small, which possibly makes the cryptosystems insecure in a practical sense [30]. Therefore, it is important to improve their efficiency for secure lattice-based cryptosystems in the type (B).

In recent years, several researchers actually considered more efficient lattice-based cryptosystems with security proofs. For example, Regev constructed an efficient lattice-based cryptosystem with shorter keys [35]. The security is based on the worst-case quantum hardness of certain approximation versions of SVP and SIVP, that is, his cryptosystem is secure if we have no polynomial-time quantum algorithm that solves the lattice problems in the worst case. Ajtai also constructed an efficient lattice-based cryptosystem with shorter keys by using a compact representation of special instances of uSVP [4], whose security is based on a certain Diophantine approximation problem.

Our Contributions. We continue to study efficient lattice-based cryptosystems with security proofs based on well-known lattice problems or other secure cryptosystems. In particular, we focus on the size of plaintexts encrypted by the cryptosystems in the type (B). To the best of the authors' knowledge, all those in this type are single-bit cryptosystems. We therefore obtain more efficient lattice-based cryptosystems with security proofs if we succeed to construct their multi-bit versions without increase in the size of ciphertexts.

In this paper, we consider multi-bit versions of the improved Ajtai-Dwork cryptosystem proposed by Goldreich, Goldwasser, and Halevi [13], the Regev cryptosystems given in [34] and in [35], and the Ajtai cryptosystem [4]. We develop a universal technique derived from a general structure behind them for constructing their multi-bit versions without increase in the size of ciphertexts.

Our technique requires precise evaluation of trade-offs between decryption errors and hardness of underlying lattice problems in the original lattice-based cryptosystems. We firstly give precise evaluation for the trade-offs to apply our technique to constructions of the multi-bit versions. This precise evaluation also clarifies a quantitative relationship between the security levels and the decryption errors in the lattice-based cryptosystems, which may be useful to improve the cryptosystems beyond our results.

Due to this evaluation of the cryptosystems, it is shown that our multi-bit versions encrypt $O(\log n)$-bit plaintexts into ciphertexts of the same length as the original ones with reasonable sacrifices of the hardness of the underlying lattice problems.

The ciphertexts of our multi-bit version are distributed in the same ciphertext space, theoretically represented with real numbers, as the original cryptosystem. To represent the real numbers in their ciphertexts, we have to round their fractional parts with certain precision. The size of ciphertexts then increases if we process the numbers with high precision. We stress that our technique does not need higher precision than that of the original cryptosystems, i.e., we take the same precision in our multi-bit versions as that of the original ones.

See Table 1 for the cryptosystems studied in this paper. (The problems in the "security" fields are defined in Appendix E.) We call the cryptosystems proposed in [13, 34, 35, 4] $\mathrm{AD}_{\mathrm{GGH}}$, R04, R05, and A05, respectively. We also call the corresponding multi-bit versions $\mathrm{mAD}_{\mathrm{GGH}}$, mR04, mR05, and mA05.

We also focus on the algebraic property we call *pseudohomomorphism* of the lattice-based cryptosystems. The homomorphism of ciphertexts is quite useful for many cryptographic applications. (See, e.g., [33].) In fact, the single-bit cryptosystems $\mathrm{AD}_{\mathrm{GGH}}$, R04, R05 and A05 implicitly have a similar property to the homomorphism. Let $E(x_1)$ and $E(x_2)$ be ciphertexts of x_1 and $x_2 \in \{0, 1\}$, respectively. Then, $E(x_1) + E(x_2)$ becomes a variant of $E(x_1 \oplus x_2)$. More precisely, $E(x_1) + E(x_2)$ does not obey the distribution

	Ajtai-Dwork		Regev'04	
cryptosystem	$\mathrm{AD}_{\mathrm{GGH}}$ [13]	$\mathrm{mAD}_{\mathrm{GGH}}$	R04 [34]	mR04
security	$O(n^{11})$-uSVP	$O(n^{11+\varepsilon})$-uSVP	$\tilde{O}(n^{1.5})$-uSVP	$\tilde{O}(n^{1.5+\varepsilon})$-uSVP
size of public key	$O(n^5 \log n)$	$O(n^5 \log n)$	$O(n^4)$	$O(n^4)$
size of private key	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(n^2)$
size of plaintext	1	$O(\log n)$	1	$O(\log n)$
size of ciphertext	$O(n^2 \log n)$	$O(n^2 \log n)$	$O(n^2)$	$O(n^2)$
rounding precision	2^{-n}	2^{-n}	2^{-8n^2}	2^{-8n^2}
	Regev'05		Ajtai	
cryptosystem	R05 [35]	mR05	A05 [4]	mA05
security	$\mathrm{SVP}_{\tilde{O}(n^{1.5})}$	$\mathrm{SVP}_{\tilde{O}(n^{1.5+\varepsilon})}$	DA′	A05
size of public key	$O(n^2 \log^2 n)$	$O(n^2 \log^2 n)$	$O(n^2 \log n)$	$O(n^2 \log n)$
size of private key	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$
size of plaintext	1	$O(\log n)$	1	$O(\log n)$
size of ciphertext	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$
rounding precision	2^{-n}	2^{-n}	$1/n$	$1/n$

Table 1: summary. (ε is any positive constant and $\tilde{O}(f(n))$ means $O(f(n)\,\mathrm{poly}(\log n))$.)

of the ciphertexts, but we can guarantee the same security level as that of the original cryptosystem and decrypt $E(x_1) + E(x_2)$ to $x_1 \oplus x_2$ by the original private key with a small decryption error. We refer to this property as the pseudohomomorphism. Goldwasser and Kharchenko actually made use of a similar property to construct the plaintext knowledge proof system for the Ajtai-Dwork cryptosystem [15].

Unfortunately, it is only over $\mathbb{Z}_2$ (and direct product groups of $\mathbb{Z}_2$ by concatenating the ciphertexts) that we can operate the addition of the plaintexts in the single-bit cryptosystems. It is unlikely that we can naively simulate the addition over large cyclic groups by concatenating ciphertexts in such single-bit cryptosystems.

In this paper, we present the pseudohomomorphic property of $\mathrm{mAD}_{\mathrm{GGH}}$, mR04, mR05, and (a slightly modified version mA05′ of) mA05 over larger cyclic groups. We believe that this property extends the possibility of the cryptographic applications of the lattice-based cryptosystems.

Main Idea for Multi-Bit Constructions and Their Security. We can actually find the following general structure behind the single-bit cryptosystems $\mathrm{AD}_{\mathrm{GGH}}$, R04, R05, and A05: Their ciphertexts of 0 are basically distributed according to a periodic Gaussian distribution and those of 1 are also distributed according to another periodic Gaussian distribution whose peaks are shifted to the middle of the period. We thus embed two periodic Gaussian distributions into the ciphertext space such that their peaks appear alternatively and regularly. (See the left side of Figure 1.)

Our technique is based on a generalization of this structure. More precisely, we regularly embed *multiple* periodic Gaussian distributions into the ciphertext space rather than only two ones. (See the right side of Figure 1.) Embedding p periodic Gaussian distributions as shown in this figure, the ciphertexts for a plaintext $i \in \{0, \ldots, p-1\}$ are distributed according the i-th periodic Gaussian distribution. This cyclic structure enables us not only to improve the efficiency of the cryptosystems but also to guarantee their security.

If we embed too many periodic Gaussian distributions, the decryption errors increase due to the overlaps of the distributions. We can then decrease the decryption errors by reducing their variance. However, it is known that smaller variance generally makes such cryptosystems less secure, as commented in [13]. We therefore have to evaluate the trade-offs in our multi-bit versions between the decryption errors and their

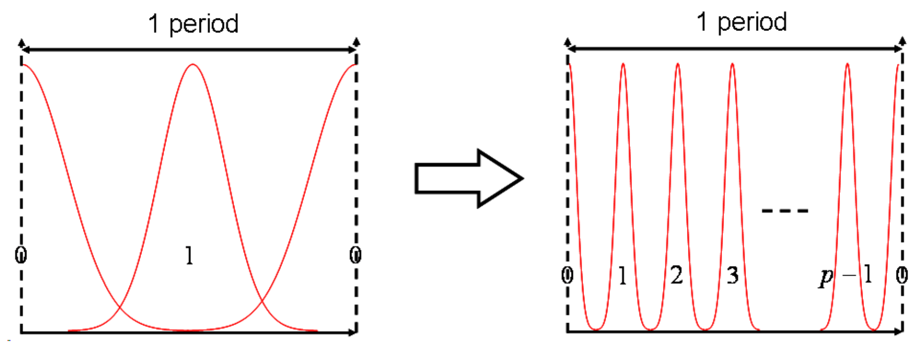

Figure 1: the embedding of periodic Gaussian distributions.

security, which depend on their own structures of the cryptosystems.

Once we evaluate their trade-offs, we can apply a general strategy based on the cyclic structure to the security proofs. The security of the original cryptosystems basically depends on the indistinguishability between a certain periodic Gaussian distribution Φ and a uniform distribution U since it is shown in their security proofs that we can construct an efficient algorithm for a certain hard lattice problem by employing an efficient distinguisher between Φ and U. The goal is thus to construct the distinguisher from an adversary against the multi-bit version.

We first assume that there exists an efficient adversary for distinguishing between two Gaussian distributions corresponding two kinds of ciphertexts in our multi-bit version with its public key. By the hybrid argument, the adversary can distinguish either between Φ_i and U or between Φ_j and U. We now suppose that it can distinguish between Φ_i and U. Note that we can slide Φ_i to Φ_0 corresponding to ciphertexts of 0 even if we do not know the private key by the cyclic property of the ciphertexts. Thus, we obtain an efficient distinguisher between Φ_0 and U. Φ_0 is in fact a variance-reduced version of the periodic Gaussian distribution Φ used in the original cryptosystem. We can guarantee the indistinguishability between such a version Φ_0 and U is based on the hardness of another lattice problem slightly easier than the original one. We can therefore guarantee the security of our multi-bit versions similarly to the original ones.

Encryption and Decryption in Multi-Bit Versions. We also exploit this cyclic structure for the correctness of encryption and decryption procedures. In the original cryptosystems except for R05, the private key is the period d of the periodic Gaussian distribution, and the public key consists of the information for generating the periodic Gaussian distribution corresponding to 0 and the information for shifting the distribution to the other distribution corresponding to 1. The latter information for the shift essentially is $k(d/2)$ for a random odd number k. Then, if we want to encrypt a plaintext 0, we generate the periodic Gaussian distribution corresponding to 0. Also, if we want to encrypt 1, we generate the distribution corresponding to 0 and then shift it using the latter information.

The private and public keys in our multi-bit versions are slightly different from those of the original ones. The major difference is the information for shifting the distribution. If the size of the plaintext space is p, the information for the shift is essentially $k(d/p)$, where the number k must be a coprime to p for unique decryption. We then interpret the number k as a generator of the "group" of periodic Gaussian distributions. We adopt a prime as the size of the plaintext space p for efficient public key generation in our constructions. The private key also contains this number k other than the period d. Therefore, we can construct correct encryption and decryption procedures using this information k.

In the cases of R05 and mR05, it is not necessary for keys to contain the information for the shift. We can actually obtain such information due to their own structures even if it is not given from the public key. Thus, p is not necessarily a prime in mR05.

Pseudohomomorphism in Multi-Bit Versions. The regular embedding of the periodic Gaussian distributions also gives our multi-bit cryptosystems the algebraic property named *pseudohomomorphism*. Recall that a Gaussian distribution has the following reproducing property: For two random variables X_1 and X_2 according to $N(m_1, s_1^2)$ and $N(m_2, s_2^2)$, where $N(m, s^2)$ is a Gaussian distribution with mean m and standard deviation s, the distribution of $X_1 + X_2$ is equal to $N(m_1 + m_2, s_1^2 + s_2^2)$. This property implies that the sum of two ciphertexts (i.e., the sum of two periodic Gaussian distributions) becomes a variant of a ciphertext (i.e., a periodic Gaussian distribution with larger variance). This sum can be moreover decrypted into the sum of two plaintexts with the private key of the multi-bit version, and has the indistinguishability based on the security of the multi-bit version. By precise analysis of our multi-bit versions, we estimate the upper bound of the number of the ciphertexts which can be summed without the change of the security and the decryption errors.

Organization. The rest of this paper is organized as follows. We describe basic notions and notations for lattice-based cryptosystems in Section 2. In Section 3, we first review the improved Ajtai-Dwork cryptosystem $\mathrm{AD}_{\mathrm{GGH}}$ and then describe the corresponding multi-bit version $\mathrm{mAD}_{\mathrm{GGH}}$ in detail. We put the description of the other multi-bit versions mR04, mR05 and mA05 to the appendices since the main idea of their constructions are based on the same universal technique and the difference among them is mainly the evaluation of the trade-offs in each of cryptosystems. We also give concluding remarks in Section 4.

2 Basic Notions and Notations

An n-dimensional lattice in $\mathbb{R}^n$ is the set $L(\mathbf{b}_1, \ldots, \mathbf{b}_n) = \{\sum_{i=1}^n \alpha_i \mathbf{b}_i : \alpha_i \in \mathbb{Z}\}$ of all integral combinations of n linearly independent vectors $\mathbf{b}_1, \ldots, \mathbf{b}_n$. The sequence of vectors $\mathbf{b}_1, \ldots, \mathbf{b}_n$ is called a *basis* of the lattice L. For clarity of notations, we represent a basis by the matrix $\mathbf{B} = (\mathbf{b}_1, \ldots, \mathbf{b}_n) \in \mathbb{R}^{n \times n}$. For any basis $\mathbf{B}$, we define the *fundamental parallelepiped* $\mathcal{P}(\mathbf{B}) = \{\sum_{i=1}^n \alpha_i \mathbf{b}_i : 0 \le \alpha_i < 1\}$. The vector $\mathbf{x} \in \mathbb{R}^n$ reduced modulo the parallelepiped $\mathcal{P}(\mathbf{B})$, denoted by $\mathbf{x} \bmod \mathcal{P}(\mathbf{B})$, is the unique vector $\mathbf{y} \in \mathcal{P}(\mathbf{B})$ such that $\mathbf{y} - \mathbf{x} \in L(\mathbf{B})$. The dual lattice L^* of a lattice L is the set $L^* = \{\mathbf{x} \in \mathbb{R}^n : \langle \mathbf{x}, \mathbf{y} \rangle \in \mathbb{Z} \text{ for all } \mathbf{y} \in L\}$. If L is generated by basis $\mathbf{B}$, then $(\mathbf{B}^T)^{-1}$ is a basis for the dual lattice, where $\mathbf{B}^T$ is the transpose of $\mathbf{B}$. For more details on lattices, see the textbook by Micciancio and Goldwasser [26].

The security parameter n of lattice-based cryptosystems is given by dimension of a lattice in the lattice problems on which security of the cryptosystems are based. Let $\lfloor x \rceil$ be the closest integer to $x \in \mathbb{R}$ (if there are two such integers, we choose the smaller.) and $\mathrm{frc}(x) = |x - \lfloor x \rceil|$ for $x \in \mathbb{R}$, i.e., $\mathrm{frc}(x)$ is the distance from x to the closest integer. We define $x \bmod y$ as $x - \lfloor x/y \rfloor y$ for $x, y \in \mathbb{R}$.

The length of a vector $\mathbf{x} = (x_1, \ldots, x_n)^T \in \mathbb{R}^n$, denoted by $\|\mathbf{x}\|$, is $(\sum_{i=1}^n x_i^2)^{1/2}$. The inner product of two vectors $\mathbf{x} = (x_1, \ldots, x_n)^T \in \mathbb{R}^n$ and $\mathbf{y} = (y_1, \ldots, y_n)^T \in \mathbb{R}^n$, denoted by $\langle \mathbf{x}, \mathbf{y} \rangle$, is $\sum_{i=1}^n x_i y_i$.

A function $f(n)$ is called negligible for sufficiently large n if $\lim_{n\to\infty} n^c f(n) = 0$ for any constant $c > 0$. We similarly call $f(n)$ a non-negligible function if there exists a constant $c > 0$ such that $f(n) > n^{-c}$ for sufficiently large n. We call probability p exponentially close to 1 if $p = 1 - 2^{-\Omega(n)}$. We represent a real number by rounding its fractional part. If the fractional part of $x \in \mathbb{R}$ is represented in m bits, the rounded number $\bar{x}$ has the precision of $1/2^m$, i.e., we have $|x - \bar{x}| \le 1/2^m$.

We say that an algorithm distinguishes between two distributions if the gap between the acceptance probability for their samples is non-negligible.

A Gaussian distribution $N(m, s^2)$ with mean m and standard derivation s is a distribution on $\mathbb{R}$ defined by the density function $\nu(l) = 1/(\sqrt{2\pi}s)\exp(-((l-m)/\sqrt{2}s)^2)$. We will make use of many variants of the Gaussian distribution in this paper. We define them when required.

3 A Multi-Bit Version of the Improved Ajtai-Dwork Cryptosystem

On behalf of four cryptosystems $\mathrm{AD_{GGH}}$, R04, R05, and A05, we discuss the improved Ajtai-Dwork cryptosystem $\mathrm{AD_{GGH}}$ given by Goldreich, Goldwasser, and Halevi [13] in detail and apply our technique to construction of its multi-bit version $\mathrm{mAD_{GGH}}$ in this section.

3.1 The Improved Ajtai-Dwork Cryptosystem and Its Multi-Bit Version

For understanding our construction intuitively, we first overview the protocol of $\mathrm{AD_{GGH}}$. Let $N = n^n = 2^{n \log n}$. We define an n-dimensional hypercube C and an n-dimensional ball B_r as $C = \{\mathbf{x} \in \mathbb{R}^n : 0 \le x_i < N, i = 1, \ldots, n\}$ and $B_r = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\| \le n^{-r}/4\}$ for any constant $r \ge 7$, respectively. For $\mathbf{u} \in \mathbb{R}^n$ and an integer i we define a hyperplane H_i as $H_i = \{\mathbf{x} \in \mathbb{R}^n : \langle \mathbf{x}, \mathbf{u} \rangle = i\}$.

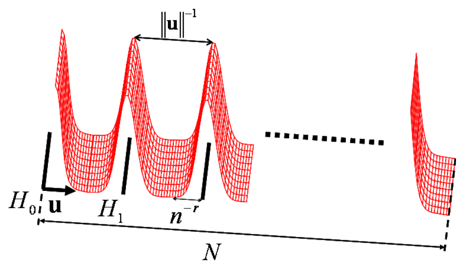

Figure 2: ciphertexts of 0 in $\mathrm{AD_{GGH}}$

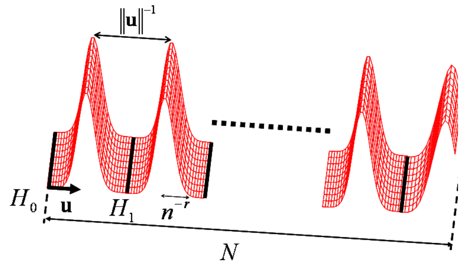

Figure 3: ciphertexts of 1 in $\mathrm{AD_{GGH}}$

Roughly speaking, $\mathrm{AD_{GGH}}$ encrypts 0 into a vector distributed closely around hidden $(n-1)$-dimensional parallel hyperplanes $H_0, H_1, H_2, \cdots$ for a normal vector $\mathbf{u}$ of H_0, and encrypts 1 into a vector distributed closely around their intermediate parallel hyperplanes $H_0 + \mathbf{u}/(2\|\mathbf{u}\|^2), H_1 + \mathbf{u}/(2\|\mathbf{u}\|^2), \cdots$. (See Figures 2 and 3.) Then, the private key is the normal vector $\mathbf{u}$. These distributions of ciphertexts can be obtained from its public key, which consists of vectors on the hidden hyperplanes and information i_1 for shifting a vector on the hyperplanes to another vector on the intermediate hyperplanes. If we know the normal vector, we can reduce the n-dimensional distribution to on the 1-dimensional one along the normal vector. Then, we can easily find whether a ciphertext distributed around the hidden hyperplanes or the intermediate ones.

We now describe the protocol of $\mathrm{AD_{GGH}}$ as follows. Our description slightly generalizes the original one by introducing a parameter r, which controls the variance of the distributions since we need to estimate a trade-off between the security and the size of plaintexts in our multi-bit version.

Preparation: All the participants agree with the security parameter n, the variance-controlling parameter r, and the precision 2^{-n} for rounding real numbers.

Key Generation: We choose $\mathbf{u}$ uniformly at random from the n-dimensional unit ball. Let $m = n^3$. Repeating the following procedure m times, we sample m vectors $\mathbf{v}_1, \ldots, \mathbf{v}_m$: (1) We choose $\mathbf{a}_i$ from $\{\mathbf{x} \in C : \langle \mathbf{x}, \mathbf{u} \rangle \in \mathbb{Z}\}$ uniformly at random, (2) choose $\mathbf{b}_1, \ldots, \mathbf{b}_n$ from B_r uniformly at random, (3) and output $\mathbf{v}_i = \mathbf{a}_i + \sum_{j=1}^{n} \mathbf{b}_j$ as a sample. We then take the minimum index i_0 satisfying that the width of $\mathcal{P}(\mathbf{v}_{i_0+1}, \ldots, \mathbf{v}_{i_0+n})$ is at least $n^{-2}N$, where width of a parallelepiped $\mathcal{P}(\mathbf{x}_1, \ldots \mathbf{x}_n)$ is defined as $\min_{i=1,\ldots,n} \mathrm{Dist}(\mathbf{x}_i, \mathrm{span}(\mathbf{x}_1, \ldots, \mathbf{x}_{i-1}, \mathbf{x}_{i+1}, \ldots, \mathbf{x}_n))$ for a distance function $\mathrm{Dist}(\cdot, \cdot)$ between a vector and an $(n-1)$-dimensional hyperplane.

Now let $\mathbf{w}_j = \mathbf{v}_{i_0+j}$ for every $j \in \{1, \ldots, n\}$, $V = (\mathbf{v}_1, \ldots, \mathbf{v}_m)$, and $W = (\mathbf{w}_1, \ldots, \mathbf{w}_n)$. We also choose an index i_1 uniformly at random from $\{i : \langle \mathbf{a}_i, \mathbf{u} \rangle \text{ is odd}\}$, where $\mathbf{a}_i$ is the vector appeared in the sampling procedure for $\mathbf{v}_i$. Note that there are such indices i_0 and i_1 with probability $1 - o(1)$. If such indices do not exist, we perform this procedure again. To guarantee the security, $\|\mathbf{u}\|$ should be in $[1/2, 1)$. The probability of this event is exponentially close to 1. If the condition is not satisfied, we sample the vector $\mathbf{u}$ again. Then, the private key is $\mathbf{u}$ and the public key is (V, W, i_1).

Encryption: Let S be a uniformly random subset of $\{1, 2, \ldots, m\}$. We encrypt a plaintext $\sigma \in \{0, 1\}$ to $\mathbf{x} = \frac{\sigma}{2}\mathbf{v}_{i_1} + \sum_{i \in S} \mathbf{v}_i \bmod \mathcal{P}(W)$.

Decryption: Let $\mathbf{x} \in \mathcal{P}(W)$ be a received ciphertext. We decrypt $\mathbf{x}$ to 0 if $\mathrm{frc}(\langle \mathbf{x}, \mathbf{u} \rangle) \leq 1/4$ and to 1 otherwise.

Carefully reading the results in [6, 13], we obtain the following theorem on the cryptosystem $\mathrm{AD_{GGH}}$.

Theorem 3.1 ([13]). *The cryptosystem* $\mathrm{AD_{GGH}}$ *encrypts a* 1*-bit plaintext into an* $n\lceil n(\log n + 1)\rceil$*-bit ciphertext with no decryption error. The security of* $\mathrm{AD_{GGH}}$ *is based on the worst case of* $O(n^{r+5})$-uSVP *for* $r \geq 7$*. The size of the public key is* $O(n^5 \log n)$ *and the size of the private key is* $O(n^2)$*.*

As commented in [9], we can actually improve the security of $\mathrm{AD_{GGH}}$ by a result in [9]. We give the precise proof in Appendix D.

Theorem 3.2. *The security of* $\mathrm{AD_{GGH}}$ *is based on the worst case of* $O(n^{r+4})$-uSVP *for* $r \geq 7$*.*

We next describe the multi-bit version $\mathrm{mAD_{GGH}}$ of $\mathrm{AD_{GGH}}$. Let p be a prime such that $2 \leq p \leq n^{r-7}$, where the parameter r controls a trade-off between the size of the plaintext space and the hardness of underlying lattice problems. In $\mathrm{mAD_{GGH}}$, we can encrypt a plaintext of $\log p$ bits into a ciphertext of the same size as $\mathrm{AD_{GGH}}$. The strategy of our construction basically follows the argument in Section 1. Note that the parameter r is chosen to keep our version error-free.

Preparation: All the participants agree with the parameters n, r and the precision 2^{-n} similarly to $\mathrm{AD_{GGH}}$, and additionally the size p of the plaintext space.

Key Generation: The key generation procedure is almost the same as that of $\mathrm{AD_{GGH}}$. We choose an index i'_1 uniformly at random from $\{i : \langle \mathbf{a}_i, \mathbf{u} \rangle \not\equiv 0 \bmod p\}$ instead of i_1 in the original key generation procedure. We set decryption information $k \equiv \langle \mathbf{a}_{i'_1}, \mathbf{u} \rangle \bmod p$. Note that there is such a k with probability $1 - (1/p)^m = 1 - o(1)$. Then, the private key is $(\mathbf{u}, k)$ and the public key is (V, W, i'_1).

Encryption: Let S be a uniformly random subset of $\{0, 1\}^m$. We encrypt $\sigma \in \{0, \ldots, p-1\}$ to $\mathbf{x} = \frac{\sigma}{p}\mathbf{v}_{i'_1} + \sum_{i \in S} \mathbf{v}_i \bmod \mathcal{P}(W)$.

Decryption: We decrypt a received ciphertext $\mathbf{x} \in \mathcal{P}(W)$ to $\lfloor p \langle \mathbf{x}, \mathbf{u} \rangle \rceil k^{-1} \bmod p$, where k^{-1} is the inverse of k in $\mathbb{Z}_p$.

Before evaluating the performance of $\mathrm{mAD_{GGH}}$ precisely, we give the summary of the results as follows.

Theorem 3.3 (security and decryption errors)**.** *Let $r \geq 7$ be any constant and let $p(n)$ be a prime such that $2 \leq p(n) \leq n^{r-7}$. The cryptosystem* $\mathrm{mAD_{GGH}}$ *encrypts a $\lfloor \log p(n) \rfloor$-bit plaintext into an $n\lceil n(\log n + 1)\rceil$-bit ciphertext without the decryption errors. The security of* $\mathrm{mAD_{GGH}}$ *is based on the worst case of $O(n^{r+4})$-*uSVP*. The size of the public key is the same as that of the original one. The size of the private key is $\lceil \log p(n) \rceil$ plus that of the original one.*

Theorem 3.4 (pseudohomomorphism)**.** *Let $r \geq 7$ be any constant. Also, let p be a prime and let κ be an integer such that $\kappa p \leq n^{r-7}$. Let E_{m} be the encryption function of* $\mathrm{mAD_{GGH}}$. *For any κ plaintexts $\sigma_1, \ldots, \sigma_\kappa$ $(0 \leq \sigma_i \leq p-1)$, we can decrypt the sum of κ ciphertexts $\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma_i) \bmod \mathcal{P}(W)$ into $\sum_{i=1}^{\kappa} \sigma_i \bmod p$ without decryption error. Moreover, if there exist two sequences of plaintexts $(\sigma_1, \ldots, \sigma_\kappa)$ and $(\sigma'_1, \ldots, \sigma'_\kappa)$, and a polynomial-time algorithm that distinguishes between $\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma_i) \bmod \mathcal{P}(W)$ and $\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma'_i) \bmod \mathcal{P}(W)$ with its public key, then there exists a polynomial-time algorithm that solves $O(n^{r+4})$-*uSVP *in the worst case with non-negligible probability.*

In what follows, we demonstrate the performance of $\mathrm{mAD_{GGH}}$ stated in the above theorems.

3.2 Decryption Errors of $\mathrm{mAD_{GGH}}$

We first evaluate the decryption error probability in $\mathrm{mAD_{GGH}}$. The following theorem can be proven by a similar argument to the analysis of [6, 13]. Since we generalize this theorem for analysis of the pseudo-homomorphism in $\mathrm{mAD_{GGH}}$ (Theorem 3.10), we here give a precise proof.

Theorem 3.5. *The cryptosystem* $\mathrm{mAD_{GGH}}$ *makes no decryption errors.*

Proof. Since the decryption error probability for any ciphertext can be estimated by sliding the distribution to that of the ciphertext of 0, we first estimate the decryption error probability for the ciphertext of 0.

Let $H := \{\mathbf{x} \in \mathbb{R}^n : \langle \mathbf{x}, \mathbf{u} \rangle \in \mathbb{Z}\}$. From the definition, $\mathrm{Dist}(\mathbf{v}_i, H) \leq n \cdot n^{-r}/4$ for $1 \leq i \leq m$. Thus, we can obtain $\mathrm{frc}\,(\langle \mathbf{v}_i, \mathbf{u} \rangle) \leq n^{1-r}/4$ and $\mathrm{frc}\,(\langle \sum_{i \in S} \mathbf{v}_i, \mathbf{u} \rangle) \leq n^{4-r}/4$. Next, we estimate an inner product between $\sum_{i \in S} \mathbf{v}_i \bmod \mathcal{P}(W)$ and $\mathbf{u}$. Let $\sum_{i \in S} \mathbf{v}_i = \mathbf{r} + \sum_{j=1}^{n} q_j \mathbf{w}_j$, where $\mathbf{r} \in \mathcal{P}(W)$. Since $\|\mathbf{w}_j\| \geq n^{-2}N$ and $p \leq n^{r-7}$, we have $|q_j| \leq n^5$ and

$$\mathrm{frc}\,(\langle \mathbf{r}, \mathbf{u} \rangle) \leq n \cdot n^5 \cdot \frac{1}{4} n^{1-r} + \frac{1}{4} n^{4-r} \leq \frac{5}{16} n^{7-r} \leq \frac{1}{2p}.$$

Therefore, we decrypt a ciphertext of 0 into 0 without decryption errors.

Now let ρ be a ciphertext of σ. Let $\mathbb{Z} \pm a := \{x \in \mathbb{R} : \mathrm{frc}\,(x) \leq a\}$ for $a \geq 0$ and $\mathbb{Z} + a \pm b := \{x \in \mathbb{R} : \mathrm{frc}\,(x - a) \leq b\}$ for $a, b \geq 0$. By a property of the key generation, we have $\left\langle \mathbf{v}_{i'_1}/p, \mathbf{u} \right\rangle \in \mathbb{Z} + k/p \pm n^{1-r}/4p$ and

$$\langle \rho, \mathbf{u} \rangle \in \mathbb{Z} + \frac{k}{p}\sigma \pm \frac{5}{16} n^{7-r} \pm \frac{1}{4p} n^{1-r} \sigma \pm \frac{1}{4} n^{4-r} \subset \mathbb{Z} + \frac{k}{p}\sigma \pm \frac{3}{8} n^{7-r}.$$

Therefore, we obtain $\langle \rho, \mathbf{u} \rangle \in \mathbb{Z} + k\sigma/p \pm 1/(2p)$ and decrypt ρ into σ without decryption errors. □

3.3 Security of $\mathrm{mAD_{GGH}}$

We next prove the security of $\mathrm{mAD_{GGH}}$. Let $U_{\mathcal{P}(W)}$ be a uniform distribution on $\mathcal{P}(W)$. We denote the encryption function of $\mathrm{AD_{GGH}}$ by E defined as a random variable $E(\sigma, (V, W, i_1))$ for a plaintext σ and a public key (V, W, i_1). If the public key is obvious, we abbreviate $E(\sigma, (V, W, i_1))$ to $E(\sigma)$. Similarly, the encryption function E_{m} is defined for $\mathrm{mAD_{GGH}}$.

First, we show that the indistinguishability between two certain distributions is based on the worst-case hardness of uSVP. The following lemma can be obtained by combining Theorem 3.2 and the results in [6] and [13] with our generalization.

Lemma 3.6 ([6, 13])**.** *If there exists a polynomial-time distinguisher between* $(E(0), (V, W, i_1))$ *and* $(U_{\mathcal{P}(W)}, (V, W, i_1))$*, there exists a polynomial-time algorithm for the worst case of* $O(n^{r+4})$-uSVP *for* $r \geq 7$.

We next present the indistinguishability between the ciphertexts of 0 in $\mathrm{mAD_{GGH}}$ and $U_{\mathcal{P}(W)}$.

Lemma 3.7. *If there exists a polynomial-time algorithm* $\mathcal{D}_1$ *that distinguishes between* $(E_{\mathrm{m}}(0), (V, W, i'_1))$ *and* $(U_{\mathcal{P}(W)}, (V, W, i'_1))$*, there exists a polynomial-time algorithm* $\mathcal{D}_2$ *that distinguishes between* $(E(0), (V, W, i_1))$ *and* $(U_{\mathcal{P}(W)}, (V, W, i_1))$*.*

Proof. We denote by $\varepsilon(n)$ the non-negligible gap of the acceptance probability of $\mathcal{D}_1$ between $E_{\mathrm{m}}(0)$ and $U_{\mathcal{P}(W)}$ with its public key. We will construct the distinguisher $\mathcal{D}_2$ from the given algorithm $\mathcal{D}_1$. To run $\mathcal{D}_1$ correctly, we first find the index i'_1 by estimating the gap of acceptance probability between $E_{\mathrm{m}}(0)$ and $U_{\mathcal{P}(W)}$ with the public key. If we can find i'_1, we output the result of $\mathcal{D}_1$ using i'_1 with the public key. Otherwise, we output a uniformly random bit. For random inputs of ciphertexts and public keys, the above procedure can distinguish between them.

We now describe the details of $\mathcal{D}_2$ as follows. We denote by $\mathbf{x}$ and (V, W, i_1) a ciphertext and a public key of $\mathrm{AD_{GGH}}$ given as an input for $\mathcal{D}_2$, respectively. Also, let $p_0 = \Pr[\mathcal{D}_1(E_{\mathrm{m}}(0), (V, W, j)) = 1]$ and $p_U = \Pr[\mathcal{D}_1(U_{\mathcal{P}(W)}, (V, W, j)) = 1]$, where the probability p_0 is taken over the inner random bits of the encryption procedure and p_U is taken over $U_{\mathcal{P}(W)}$.

(D1) For every $j \in \{1, \ldots, m\}$, we run $\mathcal{D}_1(E_{\mathrm{m}}(0), (V, W, j))$ and $\mathcal{D}_1(U_{\mathcal{P}(W)}, (V, W, j))$ $T = n/\varepsilon^2$ times. Let $x_0(j)$ and $x_U(j)$ be the number of 1 in the outputs of $\mathcal{D}_1$ for the ciphertexts of 0 and the uniform distribution with the index j, respectively.

(D2) If there exists the index j' such that $|x_0(j') - x_U(j')|/T > \varepsilon/2$, we take j' as the component of the public key.

(D3) We output $\mathcal{D}_1(\mathbf{x}, (V, W, j'))$ if we find j'. Otherwise, we output a uniformly random bit.

Note that we have $|p_0 - x_0(j')/T| \leq \varepsilon/4$ and $|p_U - x_U(j')/T| \leq \varepsilon/4$ with probability exponentially close to 1 by the Hoeffding bound [17]. Therefore, we succeed to choose the index j' with which $\mathcal{D}_1$ can distinguish between the target distributions with probability exponentially close to 1 if j' exists. By the above argument, $\mathcal{D}_1$ works correctly for a non-negligible fraction of all the inputs. $\square$

The next lemma can be proven by the hybrid argument.

Lemma 3.8. *If there exist* $\sigma_1, \sigma_2 \in \{0, \ldots, p-1\}$ *and a polynomial-time algorithm* $\mathcal{D}_3$ *that distinguishes between* $(E_{\mathrm{m}}(\sigma_1), (V, W, i'_1))$ *and* $(E_{\mathrm{m}}(\sigma_2), (V, W, i'_1))$*, there exists a polynomial-time algorithm* $\mathcal{D}_4$ *that distinguishes between* $(E_{\mathrm{m}}(0), (V, W, i'_1))$ *and* $(U_{\mathcal{P}(W)}, (V, W, i'_1))$*.*

Proof. By the hybrid argument, the distinguisher $\mathcal{D}_3$ can distinguish between $E_{\mathrm{m}}(\sigma_1)$ and $U_{\mathcal{P}(W)}$ or between $E_{\mathrm{m}}(\sigma_2)$ and $U_{\mathcal{P}(W)}$ with its public key. Without loss of generality, we can assume that $\mathcal{D}_3$ can distinguish between $E_{\mathrm{m}}(\sigma_1)$ and $U_{\mathcal{P}(W)}$ with its public key. Note that we have $E_{\mathrm{m}}(\sigma_1, (V, W, i'_1)) = E_{\mathrm{m}}(0, (V, W, i'_1)) + \frac{\sigma_1}{p}\mathbf{v}_{i'_1} \bmod \mathcal{P}(W)$ by the definition of E_{m}. Then, we can transform a given $\mathbf{x}$ from $E_{\mathrm{m}}(0, (V, W, i'_1))$ to another sample $\mathbf{y}$ from $E_{\mathrm{m}}(\sigma_1, (V, W, i'_1))$. We can therefore obtain the polynomial-time algorithm $\mathcal{D}_4$ that distinguishes between $(E_{\mathrm{m}}(0), (V, W, i'_1))$ and $(U_{\mathcal{P}(W)}, (V, W, i'_1))$. $\square$

By the above three lemmas, we obtain the security proof for our multi-bit version $\mathrm{mAD}_{\mathrm{GGH}}$.

Theorem 3.9. *If there exist plaintexts $\sigma_1, \sigma_2 \in \{0, \dots, p-1\}$ and a polynomial-time algorithm that distinguishes between the ciphertexts of σ_1 and σ_2 of* $\mathrm{mAD}_{\mathrm{GGH}}$ *with its public key, there exists a polynomial-time algorithm for the worst-case of $O(n^{r+4})$*-uSVP *for $r \geq 7$.*

3.4 Pseudohomomorphism of $\mathrm{mAD}_{\mathrm{GGH}}$

As stated in Theorem 3.4, $\mathrm{mAD}_{\mathrm{GGH}}$ has the pseudohomomorphic property. To demonstrate this property, we have to evaluate the decryption errors for sum of ciphertexts and prove its security.

Decryption Errors for Sum of Ciphertexts. First, we evaluate the decryption errors when we apply the decryption procedure to the sum of ciphertexts in $\mathrm{mAD}_{\mathrm{GGH}}$. Recall that $\mathbb{Z} \pm a := \{x \in \mathbb{R} : \mathrm{frc}\,(x) \leq a\}$ for $a \geq 0$ and $\mathbb{Z} + a \pm b := \{x \in \mathbb{R} : \mathrm{frc}\,(x - a) \leq b\}$ for $a, b \geq 0$.

Theorem 3.10. *Let $r \geq 7$ be any constant. Also let p be a prime and κ be an integer such that $\kappa p \leq n^{r-7}$. For any κ plaintexts $\sigma_1, \dots, \sigma_\kappa$ $(0 \leq \sigma_i \leq p-1)$, we can decrypt the sum of κ ciphertexts $\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma_i) \bmod \mathcal{P}(W)$ into $\sum_{i=1}^{\kappa} \sigma_i \bmod p$ without the decryption errors.*

Proof. We define $\rho_1, \dots, \rho_\kappa$ as ciphertexts of $\sigma_1, \dots, \sigma_\kappa$, respectively. We will show that we can decrypt $\rho := \sum_{i=1}^{\kappa} \rho_i \bmod \mathcal{P}(W)$ into $\sum_{i=1}^{\kappa} \sigma_i \bmod p$. From the proof of Theorem 3.5, we have

$$\langle \rho_i, \mathbf{u} \rangle \in \mathbb{Z} + \frac{k}{p}\sigma_i \pm \frac{3}{8} n^{7-r}.$$

Hence, we obtain

$$\left\langle \sum_{i=1}^{\kappa} \rho_i, \mathbf{u} \right\rangle \in \mathbb{Z} + \frac{k}{p} \sum_{i=1}^{\kappa} \sigma_i \pm \frac{3}{8} \kappa n^{7-r}.$$

Combining with the fact $\rho_i \in \mathcal{P}(W)$ and $\kappa p \leq n^{r-7}$, we have

$$\langle \rho, \mathbf{u} \rangle \in \mathbb{Z} + \frac{k}{p} \sum_{i=1}^{\kappa} \sigma_i \pm \frac{3}{8} \kappa n^{7-r} \pm \frac{1}{4} \kappa n^{2-r} \subset \mathbb{Z} + \frac{k}{p} \sum_{i=1}^{\kappa} \sigma_i \pm \frac{1}{2} \kappa n^{7-r} \subset \mathbb{Z} + \frac{k}{p} \sum_{i=1}^{\kappa} \sigma_i \pm \frac{1}{2p}.$$

Therefore, we correctly decrypt ρ into $\sum_{i=1}^{\kappa} \sigma_i \bmod p$. □

Security for Sum of Ciphertexts. We can also give the security proof for the sum of ciphertexts in $\mathrm{mAD}_{\mathrm{GGH}}$. The security proof obeys so general framework that we can apply the same argument to the security of sum of ciphertexts in the other multi-bit versions mR04, mR05, and mA05′. For convenience of the other multi-bit versions, we here present an abstract security proof for sum of ciphertexts. We denote the encryption function of our multi-bit cryptosystems by E_{m}, also regarded as a random variable $E_{\mathrm{m}}(\sigma, pk)$ for a plaintext σ and a public key pk. If the public key is obvious, we abbreviate $E_{\mathrm{m}}(\sigma, pk)$ to $E_{\mathrm{m}}(\sigma)$. Let C be the ciphertext space and U_C be the uniform distribution on C.

We first show that it is hard to distinguish between the sum of ciphertexts and the uniform distribution if it is hard to distinguish between κ samples from $E_{\mathrm{m}}(0)$ and those from U_C.

Lemma 3.11. *If there exist two sequences of plaintexts* $(\sigma_1, \ldots, \sigma_\kappa)$ *and* $(\sigma'_1, \ldots, \sigma'_\kappa)$ *and a polynomial-time algorithm* $\mathcal{D}_1$ *that distinguishes between* $(\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma_i), pk)$ *and* $(\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma'_i), pk)$*, then there exists a polynomial-time algorithm* $\mathcal{D}_2$ *that distinguishes between* κ *ciphertexts and its public key* $(E_{\mathrm{m}}(0, pk), \ldots, E_{\mathrm{m}}(0, pk), pk)$ *and uniformly random* κ *ciphertexts and the public key* $(U_C, \ldots, U_C, pk)$*.*

Proof. By the hybrid argument, the distinguisher $\mathcal{D}_1$ can distinguish between $\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma_i)$ and U_C or between $\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma'_i)$ and U_C with its public key. Without loss of generality, we can assume that $\mathcal{D}_1$ can distinguish between $(\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma_i), pk)$ and (U_C, pk). By $(\sigma_1, \ldots, \sigma_\kappa)$, we can transform $(E_{\mathrm{m}}(\sigma_1), \ldots, E_{\mathrm{m}}(\sigma_\kappa), pk)$ into $(\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma_i), pk)$. This shows the polynomial-time distinguisher $\mathcal{D}_2$. □

As already stated in Section 1 (and Lemma 3.7 in the case of $\mathrm{AD}_{\mathrm{GGH}}$), the original security proofs of $\mathrm{AD}_{\mathrm{GGH}}$, R04, R05 and A05 show that we have efficient algorithms for certain lattice problems if there is an efficient distinguisher between $E_m(0)$ and U_C with its public key. By the similar argument to that in original proofs, we also have such algorithms from efficient distinguisher $\mathcal{D}_2$ between $(E_m(0), \ldots, E_m(0), pk)$ and $(U_C, \ldots, U_C, pk)$. Thus, we obtain from $\mathcal{D}_2$ in Lemma 3.11 a probabilistic polynomial-time algorithm $\mathcal{A}$ that solve the worst case of $O(n^{r+4})$-uSVP in the case of $\mathrm{mAD}_{\mathrm{GGH}}$.

By combining the above discussion with Lemma 3.11, we guarantee the security of the sum of ciphertexts in $\mathrm{mAD}_{\mathrm{GGH}}$.

Theorem 3.12. *If there exist two sequences of plaintext* $(\sigma_1, \ldots, \sigma_\kappa)$ *and* $(\sigma'_1, \ldots, \sigma'_\kappa)$ *and a polynomial-time algorithm* $\mathcal{D}_1$ *that distinguishes between* $(\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma_i), pk)$ *and* $(\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma'_i), pk)$*, then there exists a probabilistic polynomial-time algorithm* $\mathcal{A}$ *that solves the worst case of* $O(n^{r+4})$-uSVP *in the case of* $\mathrm{mAD}_{\mathrm{GGH}}$.

4 Concluding Remarks

We have developed a universal technique for constructing multi-bit versions of lattice-based cryptosystems using periodic Gaussian distributions and revealed their pseudohomomorphism. In particular, we have showed the details of the multi-bit version of the improved Ajtai-Dwork cryptosystem in Section 3.

Although our technique achieved only logarithmic improvements on the length of plaintexts, we also obtained precise evaluation of the trade-offs between decryption errors and the hardness of underlying lattice problems in the single-bit cryptosystems. We believe that our evaluation is useful for further improvements of such single-bit cryptosystems.

Another direction of research on lattice-based cryptosystems is to find interesting cryptographic applications by their algebraic properties such as the pseudohomomorphism. Number-theoretic cryptosystems can provide a number of applications due to their algebraic structures, whereas lattice-based ones have few applications currently. For demonstration of the cryptographic advantages of lattice problems, it is important to develop the algebraic properties and their applications such as [15].

References

[1] Dorit Aharonov and Oded Regev. Lattice problems in NP ∩ coNP. *Journal of the ACM*, 52(5):749–765, 2005.

[2] Miklós Ajtai. Generating hard instances of lattice problems (extended abstract). In *STOC ’96*, pages 99–108, 1996.

[3] Miklós Ajtai. The shortest vector problem in L_2 is NP-hard for randomized reductions (extended abstract). In *STOC '98*, pages 10–19, 1998.

[4] Miklós Ajtai. Representing hard lattices with $O(n \log n)$ bits. In *STOC 2005*, pages 94–103, 2005.

[5] Miklós Ajtai and Cynthia Dwork. A public-key cryptosystem with worst-case/average-case equivalence. *ECCC*, TR96-065, 1996.

[6] Miklós Ajtai and Cynthia Dwork. A public-key cryptosystem with worst-case/average-case equivalence. In *STOC '97*, pages 284–293, 1997. Also available at ECCC TR96-065.

[7] Wojciech Banaszczyk. New bounds in some transference theorems in the geometry of numbers. *Mathematische Annalen*, 4(296):625–635, 1993.

[8] Jin-Yi Cai. A relation of primal-dual lattices and the complexity of shortest lattice vector problem. *Theoretical Computer Science*, 207(1):105–116, 1998.

[9] Jin-Yi Cai. A new transference theorem in the geometry of numbers and new bounds for Ajtai's connection factor. *Discrete Applied Mathematics*, 126(1):9–31, 2003.

[10] Jin-Yi Cai and Thomas W. Cusick. A lattice-based public-key cryptosystem. *Information and Computation*, 151(1-2):17–31, 1999.

[11] Craig Gentry. Key recovery and message attacks on NTRU-composite. In *EUROCRYPT 2001*, pages 182–194, 2001.

[12] Oded Goldreich and Shafi Goldwasser. On the limits of nonapproximability of lattice problems. *Journal of Computer and System Sciences*, 60(3):540–563, 2000.

[13] Oded Goldreich, Shafi Goldwasser, and Shai Halevi. Eliminating decryption errors in the Ajtai-Dwork cryptosystem. In *CRYPTO '97*, pages 105–111, 1997. Also available at ECCC TR97-018.

[14] Oded Goldreich, Shafi Goldwasser, and Shai Halevi. Public-key cryptosystems from lattice reduction problems. In *CRYPTO '97*, pages 112–131, 1997.

[15] Shafi Goldwasser and Dmitriy Kharchenko. Proof of plaintext knowledge for the Ajtai-Dwork cryptosystem. In *TCC 2005*, pages 529–555, 2005.

[16] Martin Grötschel, László Lovász, and Alexander Schrijver. *Geometric algorithms and combinational optimization*. Springer-Verlag, 1988.

[17] Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

[18] Nick Howgrave-Graham, Phong Q. Nguyen, David Pointcheval, John Proos, Joseph H. Silverman, Ari Singer, and William Whyte. The impact of decryption failures on the security of NTRU encryption. In *CRYPTO 2003*, pages 226–246, 2003.

[19] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. NTRU: A ring-based public key cryptosystem. In *ANTS-III*, pages 267–288, 1998.

[20] Subhash Khot. Hardness of approximating the shortest vector problem in lattices. In *FOCS 2004*, pages 126–135, 2004.

[21] S. Ravi Kumar and D. Sivakumar. On the unique shortest lattice vector problem. *Theoretical Computer Science*, 255(1-2):641–648, 2001.

[22] Jeffrey C. Lagarias. The computational complexity of simultaneous Diophantine approximation problems. *SIAM Journal of Computing*, 14(1):196–209, 1985.

[23] Arjen K. Lenstra, Hendrik W. Lenstra, and László Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261(4):513–534, 1982.

[24] Daniele Micciancio. Improving lattice based cryptosystems using the Hermite normal form. In *CaLC 2001*, pages 126–145, 2001.

[25] Daniele Micciancio. Generalized compact knapsacks, cyclic lattices, and efficient one-way functions from worst-case complexity assumptions. *ECCC*, TR2004-095, 2004.

[26] Daniele Micciancio and Shafi Goldwasser. *Complexity of Lattice Problems: a cryptographic perspective*. Kluwer Academic Publishers, 2002.

[27] Daniele Micciancio and Oded Regev. Worst-case to average-case reductions based on Gaussian measures. In *FOCS 2004*, pages 372–381, 2004.

[28] Phong Q. Nguyen. Cryptanalysis of the Goldreich-Goldwasser-Halevi cryptosystem from Crypto '97. In *CRYPTO '99*, pages 288–304, 1999.

[29] Phong Q. Nguyen and David Pointcheval. Analysis and improvements of NTRU encryption paddings. In *CRYPTO 2002*, pages 210–225, 2002.

[30] Phong Q. Nguyen and Jacques Stern. Cryptanalysis of the Ajtai-Dwork cryptosystem. In *CRYPTO '98*, pages 223–242, 1998.

[31] Seong-Hun Paeng, Bae Eun Jung, and Kil-Chan Ha. A lattice based public key cryptosystem using polynomial representations. In *PKC 2003*, pages 292–308, 2003.

[32] Chris Peikert and Alon Rosen. Efficient collision-resistant hashing from worst-case assumptions on cyclic lattices. In *TCC 2006*, pages 145–166, 2006.

[33] Dörte Rappe. Homomorphic cryptosystems and their applications. Ph.D. Thesis, University of Dortmund, 2004. Also available at http://eprint.iacr.org/2006/001.

[34] Oded Regev. New lattice based cryptographic constructions. *Journal of the ACM*, 51(6):899–942, 2004.

[35] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *STOC 2005*, pages 84–93, 2005.

[36] Mårten Trolin. The shortest vector problem in lattices with many cycles. In *CaLC 2001*, pages 194–205, 2001.

A A Multi-Bit Version of the Regev'04 Cryptosystem

A.1 The Regev'04 Cryptosystem and Its Multi-Bit Version

In this section, we consider the Regev cryptosystem R04 proposed in [34]. Roughly speaking, the ciphertexts of 0 and 1 approximately corresponds to two periodic Gaussian distributions in R04. (See Figures 4 and 5.) We now denote the distributions of the ciphertexts of 0 and 1 as Φ_0 and Φ_1, respectively. Note that every peak in Φ_1 is regularly located in the middle of two peaks in Φ_0. A parameter h is approximately equal to the number of peaks in Φ_0, and a private key d, obtained from h, corresponds to length of the period. A public key is of the form $(a_1, \dots, a_m, i_0)$, where $a_1, \dots, a_m$ are samples from Φ_0 to make a ciphertext of 0 by summing up randomly chosen elements from the samples and a certain index $i_0 \in \{1, \dots, m\}$ is used to shift a ciphertext of 0 to that of 1 by adding $a_{i_0}/2$ to a ciphertext of 0. One can easily see that we can distinguish between Φ_0 and Φ_1 with d. It however seems hard to distinguish them only with polynomially many samples of Φ_0 and i_0. Actually, it is shown in [34] that breaking R04 is at least as hard as the worst case of a certain uSVP.

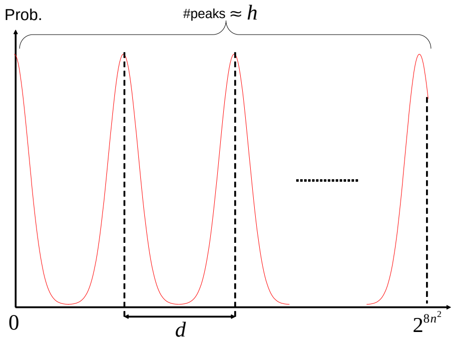

Figure 4: ciphertexts of 0 in R04

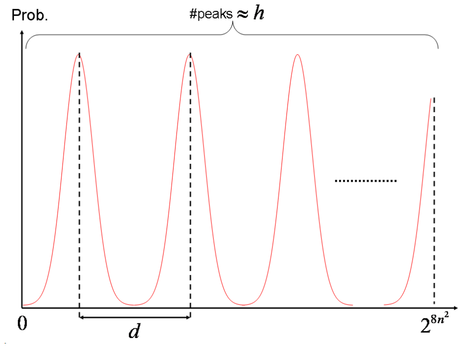

Figure 5: ciphertexts of 1 in R04

In what follows, we precisely describe the original R04. We begin with the definition of a folded Gaussian distribution Ψ_α whose density function is $\Psi_\alpha(l) = \sum_{k \in \mathbb{Z}} (1/\alpha) \exp(-\pi((l-k)/\alpha)^2)$. This distribution is obtained by "folding" a Gaussian distribution $N(0, \alpha^2/(2\pi))$ on $\mathbb{R}$ into the interval $[-1/2, 1/2)$. Note that this folded Gaussian distribution is equivalent with the fractional part of $N(0, \alpha^2/(2\pi))$. Based on this distribution, R04 makes use of a periodic distribution $\Phi_{h,\alpha}$ defined by the following density function: $\Phi_{h,\alpha}(l) = \Psi_\alpha(lh \bmod 1)$. We can sample values according to this distribution by using samples from Φ_α, as shown in [34]: (1) We sample $x \in \{0, \dots, \lceil h \rceil\}$ uniformly at random and then (2) sample y according to Ψ_α. (3) If $0 \le (x+y)/h < 1$, we then take the value as a sample. Otherwise, we repeat (1) and (2).

Let $N = 2^{8n^2}$, $m = c_0 n^2$ for a sufficiently large constant c_0, and $\gamma(n) = \omega(n\sqrt{\log n})$, specifying the size of the ciphertext space, the size of the public keys, and the variance of the folded Gaussian distribution, respectively. In this section, we require precision of $1/2^{8n^2} = 1/N$ for rounding real numbers.

Preparation: All the participants agree with the security parameter n and the precision 2^{-8n^2}.

Key Generation: Let $H = \{h \in [\sqrt{N}, 2\sqrt{N}) : \mathrm{frc}(h) < 1/(16m)\}$. We choose $h \in H$ uniformly at random and set $d = N/h$. The private key is the number d. Choosing $\alpha \in [2/\gamma(n), (2\sqrt{2})/\gamma(n))$, we sample m values $z_1, \dots, z_m$ from the distribution $\Phi_{h,\alpha}$, where $z_i = (x_i + y_i)/h$ $(i = 1, \dots, m)$ according to the above sampling procedure. Let $a_i = \lceil N z_i \rfloor$ for every $i \in \{1, \dots, m\}$. Note that we have an index i_0 such that x_{i_0} is odd with a probability exponentially close to 1. Then, the public key is $(a_1, \dots, a_m, i_0)$.

Encryption: We choose a uniformly random subset S of $\{1,\dots,m\}$. The ciphertext is $\sum_{i\in S} a_i \bmod N$ if the plaintext is 0, and $(\sum_{i\in S} a_i + \lfloor a_{i_0}/2\rfloor) \bmod N$ if it is 1.

Decryption: We decrypt a received ciphertext $w \in \{0,\dots,N-1\}$ to 0 if $\operatorname{frc}(w/d) < 1/4$ and to 1 otherwise.

Summarizing the results in [34] on the size of plaintexts, ciphertexts, and keys, the decryption errors, and the security of R04, Regev proved the following theorem.

Theorem A.1 ([34])**.** *The cryptosystem* R04 *encrypts a* 1*-bit plaintext into an* $8n^2$*-bit ciphertext with decryption error probability at most* $2^{-\Omega(\gamma^2(n)/m)} + 2^{-\Omega(n)}$*. The security of* R04 *is based on the worst case of* $O(\gamma(n)\sqrt{n})$-uSVP*. The size of the public key is* $O(n^4)$ *and the size of the private key is* $O(n^2)$*.*

We next propose a multi-bit version mR04 of the cryptosystem R04. Let p be a prime such that $2 \le p \le n^r$ and $\delta(n) = \omega(n^{1+r}\sqrt{\log n})$ for any constant $r > 0$, where the parameter r controls the trade-off between the decryption errors (or the size of plaintext space) and the hardness of underlying lattice problems. Our cryptosystem mR04 can encrypt one of p plaintexts in $\{0,\dots,p-1\}$ into a ciphertext of the same size as one of R04.

As mentioned above, R04 relates the ciphertexts to two periodic Gaussian distributions Φ_0 and Φ_1 such that each of them has one peak in a period of length d. Our construction follows the argument in Section 1. The idea of our cryptosystem is embedding of p periodic Gaussian distributions $\Phi_0,\dots,\Phi_{p-1}$ corresponding to the plaintexts $\{0,\dots,p-1\}$ into the same period of length d. We also adjust the parameter α, which affects the variance of the Gaussian distributions, to bound the decryption errors. Note that $\operatorname{frc}(h)$ also affects the decryption errors. Therefore, adjusting the set H simultaneously with α, we have to reduce the decryption errors by $\operatorname{frc}(h)$. Based on the above idea, we describe our cryptosystem mR04 as follows.

Preparation: All the participants agree with the parameters n and r, the precision 2^{-8n^2}, and the size p of the plaintext space.

Key Generation: Let $H_r = \{h \in [\sqrt{N}, 2\sqrt{N}) : \operatorname{frc}(h) < 1/(8n^r m)\}$. We choose $h \in H_r$ uniformly at random and set $d = N/h$. Choosing $\alpha \in [2/\delta(n), (2\sqrt{2})/\delta(n))$, we sample m values $z_1,\dots,z_m$ from the distribution $\Phi_{h,\alpha}$, where $z_i = (x_i+y_i)/h$ $(i = 1,\dots,m)$ according to the above sampling procedure. Let $a_i = \lceil N z_i \rceil$ for every $i \in \{1,\dots,m\}$. Additionally, we choose an index i'_0 uniformly at random from $\{i : x_i \not\equiv 0 \bmod p\}$. Then, we compute $k \equiv x_{i'_0} \bmod p$. The private key is (d,k) and the public key is $(a_1,\dots,a_m,i'_0)$.

Encryption: Let $\sigma \in \{0,\dots,p-1\}$ be a plaintext. We choose a uniformly random subset S of $\{1,\dots,m\}$. The ciphertext is $\left(\sum_{i\in S} a_i + \left\lfloor \sigma a_{i'_0}/p \right\rceil\right) \bmod N$.

Decryption: For a received ciphertext $w \in \{0,\dots,N-1\}$, we compute $\tau = w/d \bmod 1$. We decrypt the ciphertext w to $\lfloor p\tau \rceil k^{-1} \bmod p$, where k^{-1} is the inverse of k in $\mathbb{Z}_p$.

Before evaluating the performance of mR04 precisely, we give the summary of the results as follows.

Theorem A.2. *For any constant* $r > 0$*, let* $\delta(n) = \omega(n^{1+r}\sqrt{\log n})$ *and let* $p(n)$ *be a prime such that* $2 \le p(n) \le n^r$*. The cryptosystem* mR04 *encrypts a* $\lfloor \log p(n) \rfloor$*-bit plaintext into an* $8n^2$*-bit ciphertext with decryption error probability at most* $2^{-\Omega\left(\delta^2(n)/(n^{2r}m)\right)} + 2^{-\Omega(n)}$*. The security of* mR04 *is based on the worst case of* $O(\delta(n)\sqrt{n})$-uSVP*. The size of a public key is the same as that of the original one. The size of a private key is* $\lceil \log p(n) \rceil$ *plus that of the original one.*

For example, setting $\delta(n) = n^{1+r}\log n$ for any constant $r > 0$, we obtain an $\lfloor r \log n \rfloor$-bit cryptosystem with negligible decryption error, whose security is based on the worst-case of $O(n^{1.5+r}\log n)$-uSVP.

Theorem A.3 (pseudohomomorphism)**.** *Let $\delta(n) = \omega(n^{1+r}\sqrt{\log n})$. Also let $p(n)$ be a prime and κ an integer such that $\kappa p \leq n^r$ for any constant $r > 0$. Let E_{m} be the encryption function of* mR04*. For any κ plaintexts $\sigma_1, \ldots, \sigma_\kappa$ $(0 \leq \sigma_i \leq p-1)$, we can decrypt the sum of κ ciphertexts $\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma_i) \bmod N$ into $\sum_{i=1}^{\kappa} \sigma_i \bmod p$ with decryption error probability at most $2^{-\Omega((\delta(n))^2/n^{2r}m)}$. Moreover, if there exist two sequences of plaintexts $(\sigma_1, \ldots, \sigma_\kappa)$ and $(\sigma'_1, \ldots, \sigma'_\kappa)$, and a polynomial-time algorithm that distinguishes between $\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma_i) \bmod N$ and $\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma'_i) \bmod N$ with its public key, then there exists a polynomial-time algorithm that solves $O(\delta(n)\sqrt{n})$-*uSVP *in the worst case with non-negligible probability.*

In what follows, we demonstrate the performance of mR04 stated in the above theorems.

A.2 Decryption Errors of mR04

We first give the analysis of the decryption errors.

Theorem A.4. *The probability of the decryption errors in* mR04 *is at most $2^{-\Omega\left(\delta^2(n)/(n^{2r}m)\right)} + 2^{-\Omega(n)}$.*

We omit the proof of the decryption errors since it can be done by a quite similar analysis to [34] and we will prove the generalized theorem (Theorem A.9) in Appendix A.4.

A.3 Security of mR04

In what follows, we evaluate the security of our cryptosystem mR04. We first mention the result in [34] that the indistinguishability of two certain distributions is guaranteed by the hardness of a certain uSVP. Let U_N and U_1 be the uniform distributions over $\{0, \ldots, N-1\}$ and $[0, 1)$, respectively.

Lemma A.5 ([34])**.** *If there exists a polynomial-time distinguisher between $\Phi_{h,\alpha}$ and U_1 over uniformly random choices of $h \in [\sqrt{N}, 2\sqrt{N})$ and $\alpha \in [2/\delta(n), 2\sqrt{2}/\delta(n))$, there exists a polynomial-time algorithm for the worst case of $O(\delta(n)\sqrt{n})$-*uSVP.

Thus, our task is to prove the security of our cryptosystem mR04 from this indistinguishability. For convenience of the proof, we introduce a parameterized version R04′ of the cryptosystem R04. In the key generation procedure of R04′, we sample h from $H_r = \{h \in [\sqrt{N}, 2\sqrt{N}) : \mathrm{frc}(h) < 1/(8n^r m)\}$ and α from $[2/\delta, 2\sqrt{2}/\delta)$ uniformly at random. The other procedures in R04′ are the same as R04. Similarly to the case of R04, we can show that the indistinguishability between the ciphertexts of 0 in R04′ and U_N can be guaranteed by the indistinguishability between $\Phi_{h,\alpha}$ and U_N.

Lemma A.6. *For any constant $r > 0$, let p be a prime such that $2 \leq p \leq n^r$ and $\delta(n) = \omega(n^{1+r}\sqrt{\log n})$. If there exists a polynomial-time algorithm that distinguishes between ciphertexts of* 0 *in* R04′ *and U_N with its public key, there exists a polynomial-time algorithm between $\Phi_{h,\alpha}$ and U_1 over uniformly random choices of $h \in [\sqrt{N}, 2\sqrt{N})$ and $\alpha \in [2/\delta(n), 2\sqrt{2}/\delta(n))$.*

This lemma can be proven by the same way as [34] using the fact that $8n^r m \in \mathrm{poly}(n)$. By the same technique as the security proof of $\mathrm{mAD_{GGH}}$, we obtain the following lemma.

Lemma A.7. *If there exist plaintexts $\sigma_1, \sigma_2 \in \{0, \ldots, p-1\}$ and a polynomial-time algorithm that distinguishes between the ciphertexts of σ_1 and σ_2 in* mR04 *with its public key, there exists a polynomial-time algorithm that distinguishes between the ciphertexts of* 0 *in* R04′ *and U_N with its public key.*

By the above lemmas, we can show the security of mR04 based on the hardness of uSVP.

Theorem A.8. *If there exist plaintexts $\sigma_1, \sigma_2 \in \{0, \ldots, p-1\}$ and a polynomial-time algorithm that distinguishes between the ciphertexts of σ_1 and σ_2 in* mR04 *with its public key, there exists a polynomial-time algorithm for the worst-case of $O(\delta(n)\sqrt{n})$-*uSVP.

A.4 Pseudohomomorphism of mR04

Decryption Errors for Sum of Ciphertexts.

Theorem A.9 (mR04)**.** *Let $\delta(n) = \omega(n^{1+r}\sqrt{\log n})$. Also let $p(n)$ be a prime and κ be an integer such that $\kappa p \leq n^r$ for any constant $r > 0$. For any κ plaintexts $\sigma_1, \ldots, \sigma_\kappa$ $(0 \leq \sigma_i \leq p-1)$, we can decrypt the sum of κ ciphertexts $\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma_i) \bmod N$ into $\sum_{i=1}^{\kappa} \sigma_i \bmod p$ with decryption error probability at most $2^{-\Omega((\delta(n))^2/n^{2r}m)}$.*

Before the proof, we need the following lemma given in [34] to bound the tails of Gaussian distributions.

Lemma A.10 ([34])**.** *The probability that the distance of a normal variable with variance σ^2 from its mean is more than t is at most $\sqrt{\frac{2}{\pi}}\frac{\sigma}{t}\exp\left(-\frac{t^2}{2\sigma^2}\right)$, i.e.,*

$$\Pr_{X \sim N(\mu, \sigma^2)}[|X-\mu| > t] \leq \sqrt{\frac{2}{\pi}}\frac{\sigma}{t}\exp\left(-\frac{t^2}{2\sigma^2}\right).$$

By Lemma A.10, one can see easily that if $\sigma \leq 1/\sqrt{n}$, the probability $\Pr_{X\sim N(0,\sigma^2)}[|X| > 1/2]$ is exponentially small in n.

Proof. The proof is similar to the estimation of the decryption errors in [34]. First, we show the case that we have κ ciphertexts of 0, $\rho_1, \ldots, \rho_\kappa$. The probabilities are taken over the choices of the private and public keys and the inner random bits of the encryption procedure. Let $S_1, \ldots, S_\kappa$ denote the subsets of indices used in the encryption procedure, i.e., $\rho_i = \sum_{j\in S_i} a_j \bmod N$. Let $\rho := \sum_{i=1}^{\kappa} \rho_i \bmod N$. Thus,

$$\left|\rho - \left(\sum_{i=1}^{\kappa}\left(\sum_{j\in S_i} a_j \bmod d\lfloor h\rceil\right) \bmod d\lfloor h\rceil\right)\right| \leq m\kappa\,|N - d\lfloor h\rceil| = m\kappa d \cdot \mathrm{frc}\,(h) < \frac{\kappa}{8n^r}d.$$

Similarly to the argument for evaluation of the decryption errors in [34], we obtain

$$\begin{aligned}
\mathrm{frc}\left(\frac{\rho}{d}\right) &< \frac{\kappa}{8n^r} + \mathrm{frc}\left(\frac{\sum_{i=1}^{\kappa}\left(\sum_{j\in S_i} a_i \bmod d\lfloor h\rceil\right) \bmod d\lfloor h\rceil}{d}\right) \\
&= \frac{\kappa}{8n^r} + \mathrm{frc}\left(\frac{\sum_{i=1}^{\kappa}\sum_{j\in S_i} a_j}{d}\right) \\
&< \frac{\kappa}{8n^r} + \frac{m\kappa}{d} + \mathrm{frc}\left(\frac{N}{d}\sum_{i=1}^{\kappa}\sum_{j\in S_i} z_j\right).
\end{aligned}$$

Since $z_j = (x_j + y_j)/h$ and $d = N/h$,

$$\mathrm{frc}\left(\frac{N}{d}\sum_{i=1}^{\kappa}\sum_{j\in S_i} z_j\right) = \mathrm{frc}\left(\sum_{i=1}^{\kappa}\sum_{j\in S_i}(x_j + y_j)\right) = \mathrm{frc}\left(\sum_{i=1}^{\kappa}\sum_{j\in S_i} y_j\right).$$

Hence, we have

$$\operatorname{frc}\left(\frac{\rho}{d}\right) < \frac{\kappa}{8n^r} + \frac{m\kappa}{d} + \operatorname{frc}\left(\sum_{i=1}^{\kappa}\sum_{j\in S_i} y_j\right) < \frac{3\kappa}{16n^r} + \operatorname{frc}\left(\sum_{i=1}^{\kappa}\sum_{j\in S_i} y_j\right),$$

where we used the fact that $d = 2^{\Theta(4n^2)}$ is much larger than $m = c_0 n^2$. All x_i are strictly less than $\lceil h \rceil - 1$ with probability exponentially close to 1. Conditioned on that, $y_1, \ldots, y_m$ are distributed according to Ψ_α. Therefore, we have

$$\Pr\left[\operatorname{frc}\left(\sum_{i=1}^{\kappa}\sum_{j\in S_i} y_j\right) > \frac{1}{16p}\right] \le \Pr\left[\operatorname{frc}\left(\sum_{j=1}^{m} \kappa y_j\right) > \frac{1}{16p}\right].$$

The distribution of $\sum_{j=1}^{m} \kappa y_j$ mod 1 is $\Psi_{\sqrt{m\kappa}\alpha}$. Since $\sqrt{m\kappa}\alpha = O(\frac{\sqrt{m\kappa}}{\delta(n)})$, we obtain

$$\Pr\left[\operatorname{frc}\left(\sum_{i=1}^{\kappa}\sum_{j\in S_i} y_j\right) > \frac{1}{16p}\right] \le 2^{-\Omega((\delta(n))^2/m\kappa p^2)} \le 2^{-\Omega((\delta(n))^2/n^{2r}m)}$$

by Lemma A.10. We thus obtain $\operatorname{frc}(\rho/d) < 1/(4p)$, which implies that we can decrypt ρ to 0 with decryption error probability at most $2^{-\Omega((\delta(n))^2/mn^{2r})}$.

Next, we consider κ ciphertexts $\rho'_1, \ldots, \rho'_\kappa$ of plaintexts $\sigma_1, \ldots, \sigma_\kappa$ respectively and set $\rho' := \sum_{i=1}^{\kappa} \rho_i$ mod N. From the encryption procedure, $\rho'_i = \rho_i + \lfloor \sigma_i a_{i'_0}/p \rceil$ mod N. By using the fact that $k \equiv x_{i'_0}$ mod p and that with probability exponentially close to 1, $y_{i'_0} \in \mathbb{Z} \pm 1/(8n^r)$, we get $\lfloor a_{i'_0}/p \rceil / d \in \mathbb{Z} + k/p \pm 1/(8pn^r) \pm 2/d$. Hence, we have $\lfloor \sigma_i a_{i'_0}/p \rceil / d \in \mathbb{Z} + \sigma_i k/p \pm 1/(8n^r) \pm 2/d$. This implies that

$$\sum_{i=1}^{\kappa} \frac{\lfloor \sigma_i a_{i'_0}/p \rceil}{d} \in \mathbb{Z} + \frac{k}{p}\sum_{i=1}^{\kappa}\sigma_i \pm \frac{\kappa}{8n^r} \pm \frac{2\kappa}{d}.$$

Since $\operatorname{frc}(\rho/d) < 1/(4p)$, we obtain

$$\frac{\rho'}{d} \in \mathbb{Z} + \frac{k}{p}\sum_{i=1}^{\kappa}\sigma_i \pm \frac{1}{4p} \pm \frac{\kappa}{8n^r} \pm \frac{\kappa+1}{8n^r m} \pm \frac{2\kappa}{d} \subset \mathbb{Z} + \frac{k}{p}\sum_{i=1}^{\kappa}\sigma_i \pm \frac{1}{2p}$$

with the probability at most $2^{-\Omega((\delta(n))^2/mn^{2r})}$, which completes the proof. □

Security for Sum of Ciphertexts. By similar argument in Section 3.4, we obtain the following theorem.

Theorem A.11. *If there exist two sequences of plaintext $(\sigma_1, \ldots, \sigma_\kappa)$ and $(\sigma'_1, \ldots, \sigma'_\kappa)$ and a polynomial-time algorithm $\mathcal{D}_1$ that distinguishes between $(\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma_i), pk)$ and $(\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma'_i), pk)$, then there exists a probabilistic polynomial-time algorithm $\mathcal{A}$ that solves the worst case of $O(\delta(n)\sqrt{n})$-*uSVP *in the case of* mR04.

B A Multi-Bit Version of the Regev'05 Cryptosystem

B.1 The Regev'05 Cryptosystem and Its Multi-Bit Version

The cryptosystem R05 proposed in 2005 [35] is also constructed by using a variant of Gaussian distributions. A folded Gaussian distribution Ψ_α over $[-1/2, 1/2)$ is given by a density function $\Psi_\alpha(l) =$

$\sum_{k \in \mathbb{Z}} (1/\alpha) \exp(-\pi((l-k)/\alpha)^2)$. Let $m = 5(n+1)(2\log n + 1) = \Theta(n \log n)$ and $q(n) \in [n^2, 2n^2]$ be a prime. The parameter $\alpha = \alpha(n)$ satisfying conditions that $\alpha(n) = o(1/(\sqrt{n} \log n))$ and $\alpha(n)q(n) > 2\sqrt{n}$ is used to control the variance of the distribution Ψ_α. (In [35], α is set to $1/(\sqrt{n}\log^2 n)$.) We also describe the discretized distribution on $\mathbb{Z}_q$ from Ψ_α. The Gaussian distribution Φ_α on $\mathbb{Z}_q$ is obtained by sampling from Ψ_α, multiplying q, and rounding the closest integer modulo q. The distribution can be formally defined as $\Phi_\alpha(l) = \int_{(l-1/2)/q}^{(l+1/2)/q} \Psi_\alpha(x)dx$.

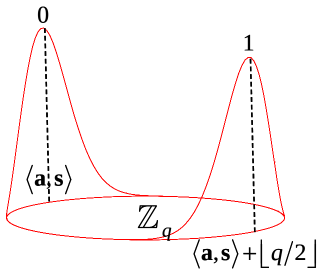

Figure 6: cryptosystem R05

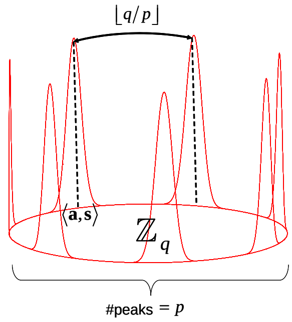

Figure 7: multi-bit version of R05

In R05, the ciphertexts of 0 and 1 are vectors in $\mathbb{Z}_q^n$ obtained from some Gaussian distributions, which are specified by vectors $\mathbf{a}_1, \ldots, \mathbf{a}_m$ shared among all the participants in the preparation procedure. Every coordinate i of the ciphertext of 0 corresponds to a Gaussian distribution on $\mathbb{Z}_q$ with mean $\langle \mathbf{a}_i, \mathbf{s} \rangle$ for the private key $\mathbf{s}$. On the other hand, the ciphertext of 1 corresponds to the “opposite” Gaussian distribution. (See Figure 6.)

Preparation: All the participants agree with the security parameter n, the variance-controlling parameter α, and the precision 2^{-n}. They also share m vectors $\mathbf{a}_1, \ldots, \mathbf{a}_m$ chosen from $\mathbb{Z}_q^n$ uniformly at random.

Key Generation: The private key $\mathbf{s}$ is chosen uniformly at random from $\mathbb{Z}_q^n$. We also choose $e_1, \ldots, e_m$ according to the distribution Φ_α. Let $b_i = \langle \mathbf{a}_i, \mathbf{s} \rangle + e_i$ for every $i \in \{1, \ldots, m\}$. The public key is $\{(\mathbf{a}_i, b_i)\}_{i=1,\ldots,m}$.

Encryption: We choose a uniformly random subset S of $\{1, \ldots, m\}$. The ciphertext is $(\sum_{i \in S} \mathbf{a}_i, \sum_{i \in S} b_i)$ if the plaintext is 0, and $(\sum_{i \in S} \mathbf{a}_i, \lfloor q/2 \rfloor + \sum_{i \in S} b_i)$ if it is 1.

Decryption: We decrypt a received ciphertext $(\mathbf{a}, b) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ into 0 if $|(b - \langle \mathbf{a}, \mathbf{s} \rangle) \bmod q| < q/4$, and into 1 otherwise, where $|\cdot|$ is the absolute value function on $\mathbb{Z}_q$, i.e., $|x| = \min\{x, q-x\}$ for any $x \in \mathbb{Z}_q$.

Note that the security reduction of R05 is done by a polynomial-time quantum algorithm. In other word, if R05 is insecure, there exists a polynomial-time quantum algorithm for certain lattice problems. As shown in [35], the cryptosystem R05 has the following performance.

Theorem B.1 ([35]). *The cryptosystem* R05 *encrypts a* 1*-bit plaintext into an* $(n+1)\lceil \log q \rceil$*-bit ciphertext with decryption error probability at most* $2^{-\Omega\left(1/(m\alpha^2(n))\right)} + 2^{-\Omega(n)}$. *The security of* R05 *is based on the worst case of* $\mathrm{SVP}_{\tilde{O}(n/\alpha(n))}$ *and* $\mathrm{SIVP}_{\tilde{O}(n/\alpha(n))}$ *for polynomial-time quantum algorithms. The size of the public key is* $O(n \log^2 n)$ *and the size of the private key is* $O(n \log n)$.

We now give our cryptosystem mR05 based on R05. (See Figure 7.) Let $r \in (0,1)$ be any constant, which controls the trade-off between the size of plaintext space and the hardness of underlying lattice problems, and p be an integer such that $p \le n^r = o(n)$, which is the size of the plaintext space in mR05. mR05 can encrypt a plaintext in $\{0,\ldots,p-1\}$ into a ciphertext of the same size as R05. We use the same parameters m and q as R05 and introduce a parameter $\beta = \beta(n) = \alpha(n)/n^r = o(1/(n^{0.5+r}\log n))$ to control the distribution instead of α in R05. The parameter $\beta(n)$ must satisfy $\beta(n)q(n) > 2\sqrt{n}$.

Preparation: All the participants agree with the parameters n,β, the precision 2^{-n}, and the size p of the plaintext space. They also share m vectors $\mathbf{a}_1,\ldots,\mathbf{a}_m$ chosen from $\mathbb{Z}_q^n$ uniformly at random.

Key Generation: This procedure is the same as R05 except that we sample $e_1,\ldots,e_m$ from Φ_β.

Encryption: We choose a uniformly random subset S of $\{1,\ldots,m\}$. For a plaintext $\sigma \in \{0,\ldots,p-1\}$, the ciphertext is $(\sum_{i\in S}\mathbf{a}_i, \lfloor \sigma q/p \rceil + \sum_{i\in S} b_i)$.

Decryption: We decrypt a received ciphertext $(\mathbf{a},b)$ to $\lfloor (b - \langle \mathbf{a},\mathbf{s}\rangle)p/q \rceil \bmod p$.

Before evaluating the performance of mR05 precisely, we give the summary of the results as follows.

Theorem B.2. *Let $p = p(n)$ be an integer such that $p(n) \le n^r$ for any constant $0 < r < 1$. The cryptosystem* mR05 *encrypts a $\lfloor \log p(n) \rfloor$-bit plaintext into an $(n+1)\lceil \log q \rceil$-bit ciphertext with decryption error probability at most $2^{-\Omega\left(1/(m\beta^2(n)n^{2r})\right)} + 2^{-\Omega(n)}$. The security of* mR05 *is based on the worst case of $\mathrm{SVP}_{\tilde{O}(n/\beta(n))}$ and $\mathrm{SIVP}_{\tilde{O}(n/\beta(n))}$ for polynomial-time quantum algorithms. The size of the public key and private key is the same as that of the original one.*

For example, by setting $p(n) = n^r$ for a constant $0 < r < 1$ and $\beta(n) = 1/(n^{0.5+r}\log^2 n)$, we obtain a $\lfloor r \log n \rfloor$-bit cryptosystem with negligible decryption error whose security is based on $\mathrm{SVP}_{\tilde{O}(n^{1.5+r})}$ and $\mathrm{SIVP}_{\tilde{O}(n^{1.5+r})}$.

Theorem B.3 (pseudohomomorphism)**.** *Let $p(n)$ be an integer and κ be an integer such that $\kappa p \le n^r$ for any constant $0 < r < 1$. Let E_{m} be the encryption function of* mR05*. For any κ plaintexts $\sigma_1,\ldots,\sigma_\kappa$ $(0 \le \sigma_i \le p-1)$, we can decrypt the sum of κ ciphertexts $\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma_i)$ into $\sum_{i=1}^{\kappa}\sigma_i \bmod p$ with decryption error probability at most $2^{-\Omega(1/(m\beta^2(n)n^{2r}))}$, where the addition is defined over $\mathbb{Z}_q^n \times \mathbb{Z}_q$. Moreover, if there exist two sequences of plaintexts $(\sigma_1,\ldots,\sigma_\kappa)$ and $(\sigma'_1,\ldots,\sigma'_\kappa)$, and a polynomial-time algorithm that distinguishes between $\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma_i)$ and $\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma'_i)$ with its public key, then there exist polynomial-time quantum algorithms that solve $\mathrm{SVP}_{\tilde{O}(n/\beta(n))}$ and $\mathrm{SIVP}_{\tilde{O}(n/\beta(n))}$ in the worst case with non-negligible probability.*

In what follows, we demonstrate the performance of mR05 stated in the above theorems.

B.2 Decryption Errors of mR05

We first estimate the decryption errors in our cryptosystem mR05. By replacing the parameter α in R05 to the parameter β in mR05, we immediately obtain the evaluation of the decryption errors from Theorem B.1. The generalization of this theorem (Theorem B.8) is also given in Appendix B.4.

Theorem B.4. *The probability of the decryption errors in* mR05 *is at most $2^{-\Omega\left(1/(m\beta^2(n)n^{2r})\right)} + 2^{-\Omega(n)}$.*

B.3 Security of mR05

We next discuss the security of our cryptosystem mR05. Let U_{R05} be the uniform distribution over the ciphertext space $\mathbb{Z}_q^n \times \mathbb{Z}_q$ of R05 (and mR05). The strategy of the security proof for mR05 is similar to mR04. We first mention the result in [35] that the indistinguishability between the ciphertexts of 0 in R05 and U_{R05} is guaranteed by the worst-case hardness of certain lattice problems.

Lemma B.5 ([35])**.** *If there exists a polynomial-time algorithm that distinguishes between the ciphertexts of* 0 *in* R05 *and* U_{R05} *with its public key, there exists a polynomial-time quantum algorithm for the worst case of* $\mathrm{SVP}_{\tilde{O}(n/\alpha(n))}$ *and* $\mathrm{SIVP}_{\tilde{O}(n/\alpha(n))}$.

We now consider a slightly modified version R05′ with the distribution parameter $\beta(n) = \alpha(n)/n^r = o(1/(n^{0.5+r}\log n))$ instead of $\alpha(n)$ in R05. Since the trade-off between the decryption error and the security of R05′ is obtained by Theorem B.1, we can show the following lemma by the same technique as the security proof of $\mathrm{mAD}_{\mathrm{GGH}}$.

Lemma B.6. *If there exist plaintexts* $\sigma_1, \sigma_2 \in \{0, \ldots, p-1\}$ *and a polynomial-time algorithm that distinguishes between the ciphertexts of* σ_1 *and* σ_2 *in* mR05 *with its public key, there exists a polynomial-time algorithm that distinguishes between the ciphertexts of* 0 *in* R05′ *and* U_{R05} *with its public key.*

By these lemmas, we can obtain the security of our cryptosystem mR05.

Theorem B.7. *If there exist plaintexts* $\sigma_1, \sigma_2 \in \{0, \ldots, p-1\}$*, and a polynomial-time algorithm that distinguishes between the ciphertext of* σ_1 *and* σ_2 *in* mR05 *with its public key, there exists a polynomial-time quantum algorithm for the worst-case of* $\mathrm{SVP}_{\tilde{O}(n/\beta(n))}$ *and* $\mathrm{SIVP}_{\tilde{O}(n/\beta(n))}$.

We omit the proof of the security since it is quite similar to $\mathrm{mAD}_{\mathrm{GGH}}$.

B.4 Pseudohomomorphism of mR05

Decryption Errors for Sum of Ciphertexts.

Theorem B.8 (mR05)**.** *Let* $\beta(n) = o(1/(n^{0.5+r}\log n))$*. Also let* $p(n)$ *be an integer and* κ *be an integer such that* $\kappa p \leq n^r$ *for any constant* $0 < r < 1$*. For any* κ *plaintexts* $\sigma_1, \ldots, \sigma_\kappa$ $(0 \leq \sigma_i \leq p-1)$*, we can decrypt the sum of* κ *ciphertexts* $\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma_i)$ *into* $\sum_{i=1}^{\kappa}\sigma_i \bmod p$ *with decryption error probability at most* $2^{-\Omega(1/(m\beta^2(n)n^{2r}))}$*, where the addition is defined over* $\mathbb{Z}_q^n \times \mathbb{Z}_q$.

Proof. The proof is similar to [35]. First, we estimate the decryption errors for the sum of κ ciphertexts of 0, $(\rho_1, \upsilon_1), \ldots, (\rho_\kappa, \upsilon_\kappa)$. The probabilities are taken over the choices of the private and public keys and the randomness of the encryption procedure. Let $S_1, \ldots, S_\kappa$ denote the subsets of indices used in the encryption procedure, i.e., $(\rho_i, \upsilon_i) = (\sum_{j\in S_i}\mathbf{a}_j, \sum_{j\in S_i} b_j)$. Let $(\rho, \upsilon) = (\sum_{i=1}^{\kappa}\rho_i, \sum_{i=1}^{\kappa}\upsilon_i)$. Recall that we obtain $\sum_{i=1}^{\kappa}\sum_{j\in S_i} e_j = \upsilon - \langle\rho, \mathbf{s}\rangle$ in the key generation. We will show

$$\Pr\left[\left|\sum_{i=1}^{\kappa}\sum_{j\in S_i} e_i \bmod q\right| > \frac{\lfloor q/p\rfloor}{4}\right] < 2^{-\Omega(1/(m\beta^2 n^{2r}))}, \tag{1}$$

where $e_1, \ldots, e_\kappa$ are samples from the distribution Φ_β and $|x| := \min\{x, q-x\}$ for $x \in [0, q-1)$. A sample from Φ_β can be obtained by sampling x_i from Ψ_β and outputting $\lfloor qx_i\rceil \bmod q$. Notice that $\sum_{i=1}^{\kappa}\sum_{j\in S_i}\left\lfloor qx_j\right\rceil \bmod q$ is at most $m\kappa < q/(16p)$ away from $\sum_{i=1}^{\kappa}\sum_{j\in S_i} qx_i \bmod q$ for sufficiently large n. Therefore, it is sufficient to show

$$\Pr\left[\left|\sum_{i=1}^{\kappa}\sum_{j\in S_i} qx_i\right| > \frac{q}{16p}\right] < 2^{-\Omega(1/(m\beta^2 n^{2r}))},$$

where $x_1, \ldots, x_\kappa$ are independently distributed according to Ψ_β. That is, it is sufficient to show

$$\Pr\left[\mathrm{frc}\left(\sum_{i=1}^{\kappa}\sum_{j\in S_i} x_i\right) > \frac{1}{16p}\right] < 2^{-\Omega(1/(m\beta^2 n^{2r}))}.$$

Similarly to the argument in Theorem A.9, we obtain

$$\Pr\left[\mathrm{frc}\left(\sum_{i=1}^{\kappa}\sum_{j\in S_i} x_i\right) > \frac{1}{16p}\right] \le \Pr\left[\mathrm{frc}\left(\sum_{j=1}^{m}\kappa x_i\right) > \frac{1}{16p}\right] \le 2^{-\Omega(1/m\kappa p^2\beta^2)} \le 2^{-\Omega(1/m\beta^2 n^{2r})}.$$

It follows that we can decrypt (ρ, υ) into 0 with decryption error probability at most $2^{-\Omega(1/(m\beta^2 n^{2r}))}$.

Next, we consider κ ciphertexts $(\rho'_1, \upsilon'_1), \ldots, (\rho'_\kappa, \upsilon'_\kappa)$ of plaintexts $\sigma_1, \ldots, \sigma_\kappa$ respectively. We now set $(\rho', \upsilon') := (\sum_{i=1}^{\kappa}\rho'_i, \sum_{i=1}^{\kappa}\upsilon'_i)$. By the encryption procedure, $\upsilon'_i = \upsilon_i + \lfloor \sigma_i q/p \rceil$. Therefore, we have $\upsilon' - \langle \rho', \mathbf{s}\rangle = \sum_{i=1}^{\kappa}\sum_{j\in S_i} e_j + \sum_{i=1}^{\kappa}\lfloor \sigma_i q/p \rceil$. Combining the equation (1) and the fact that $\left|\sum_{i=1}^{\kappa}\lfloor \sigma_i q/p\rceil - \sum_{i=1}^{\kappa}\sigma_i q/p\right| \le \kappa < \lfloor q/p \rfloor/4$, we decrypt (ρ', υ') into $\sum_{i=1}^{\kappa}\sigma_i \bmod p$ with decryption error probability at most $2^{-\Omega(1/(m\beta^2 n^{2r}))}$. □

Security for Sum of Ciphertexts. By similar argument in Section 3.4, we obtain the following theorem.

Theorem B.9. *If there exist two sequences of plaintext $(\sigma_1, \ldots, \sigma_\kappa)$ and $(\sigma'_1, \ldots, \sigma'_\kappa)$ and a polynomial-time algorithm $\mathcal{D}_1$ that distinguishes between $(\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma_i), pk)$ and $(\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma'_i), pk)$, thenthere exists a polynomial-time quantum algorithm for the worst case of* $\mathrm{SVP}_{\tilde{O}(n/\alpha(n))}$ *and* $\mathrm{SIVP}_{\tilde{O}(n/\alpha(n))}$ *in the case of* mR05.

C A Multi-Bit Version of the Ajtai Cryptosystem

C.1 The Ajtai Cryptosystem and Its Multi-Bit Version

Let b be a uniformly random string of $O(n^2 \log n)$ bits and t be a random string of $O(n \log n)$ bits specified later. We denote by $\nu_s^{(n)}$ a Gaussian distribution on an n-dimensional Euclidean space with mean $\mathbf{0}$ and standard deviation s. The density function is given by $\nu_s^{(n)}(\mathbf{x}) = s^{-n}\exp(-\pi\|\mathbf{x}/s\|^2)$. Note that, given an orthonormal basis for $\mathbb{R}^n$, $\nu_s^{(n)}$ can be written as the sum of n orthogonal 1-dimensional Gaussian distributions along one of the basis vectors. For instance, given a basis $\{\mathbf{e}_1, \ldots, \mathbf{e}_n\}$, $\nu_s^{(n)}(\mathbf{x}) = \prod_{i=1}^{n}(1/s)\exp(-\pi(x_i/s)^2)$ for any $\mathbf{x} = \sum_{i=1}^{n} x_i\mathbf{e}_i$.

Ajtai showed how to generate a certain class of efficiently representable lattices related to hard problems in [4]. He also succeeded to construct two lattice-based cryptosystems based on the original Ajtai-Dwork cryptosystem [6] and the improved Ajtai-Dwork cryptosystem [13]. The latter one reduces decryption error from the former one by the idea of [13]. In this section, we only describe the former one, which is related to security of our cryptosystem.

In the Ajtai cryptosystem A05, we make use of a periodic Gaussian distribution on $\mathbb{R}^n$ such that its peaks are located on the points of the dual lattice spanned by a basis F of an instance $L(b, t)$ of uSVP obtained in the preparation procedure. Then, the periodic Gaussian distribution looks like a “wave” going along the shortest vector $\mathbf{u}$ of $L(b, t)$ since the dual lattice of $L(b, t)$, which is an instance of uSVP, has a much longer interval between two $(n-1)$-dimensional sublattices orthogonal to $\mathbf{u}$ than others. (See Figure 8.) Then, the ciphertexts of 0 correspond to the periodic Gaussian distribution modulo $\mathcal{P}(F)$ and those of 1 correspond to the uniform distribution on $\mathcal{P}(F)$ in the cryptosystem A05. Similarly to the previous cryptosystems, if we

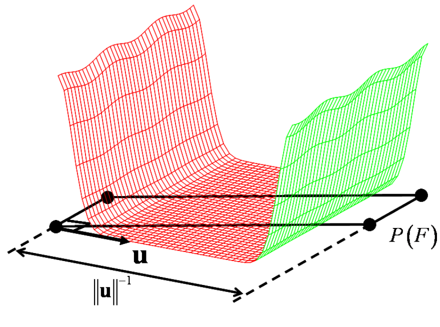

Figure 8: ciphertexts of 0 in A05

know $\mathbf{u}$, we can easily decrypt a received ciphertext by the inner product between the ciphertext and $\mathbf{u}$ with high probability.

We now describe the details of the Ajtai cryptosystem A05. All the participants share a probabilistic polynomial-time algorithm $\mathcal{D}$, a deterministic polynomial-time algorithm $\mathcal{B}$, and a uniformly random string b. In the preparation procedure, $\mathcal{D}$ generates a random string t and a vector $\mathbf{u}$ in a lattice $L(b,t)$ from b. Also, $\mathcal{B}$ generates a basis $B(b,t)$ of the lattice $L(b,t)$ if strings b and t are given. Then, the probability that $L(b,t)$ is an instance of $n^{1/2+r}$-uSVP and $\mathbf{u}$ is its unique shortest vector such that $n^{-r/2} \leq \|\mathbf{u}\| \leq n^{-r/3}$ is exponentially close to 1. Now let $F = (\mathbf{f}_1,\dots,\mathbf{f}_n)$ be a basis of the dual lattice of $L(b,t)$. We also denote by $U_{\mathcal{P}(F)}$ the uniform distribution on $\mathcal{P}(F)$.

Preparation: All the participants agree with the security parameter n, and share the algorithms $\mathcal{B},\mathcal{D}$ and the random string b.

Key Generation: We give b to the procedure $\mathcal{D}$, and then obtain t and $\mathbf{u}$. Then, the private key is $\mathbf{u}$ and the public key is t.

Encryption: Let $\sigma \in \{0,1\}$ be an encrypted plaintext. If $\sigma = 0$, we choose $\mathbf{z}$ from a Gaussian distribution on the n-dimensional Euclidean space given by the density function $\nu^{(n)}(\mathbf{x}) = \exp(-\pi\|\mathbf{x}\|^2)$. We then set $\mathbf{y} = (y_1,\dots,y_n)^T = \mathbf{z} \bmod \mathcal{P}(F)$. If $\sigma = 1$, we choose $\mathbf{y}$ from the uniform distribution $U_{\mathcal{P}(F)}$. These operations for real numbers are done with precision $2^{-n\log n}$. The ciphertext $\bar{\mathbf{y}} = (\bar{y}_1,\dots,\bar{y}_n)^T$ is obtained by rounding $\mathbf{y}$ with precision of $1/n$, i.e., we have $|\bar{y}_i - y_i| \leq 1/n$ for every $i \in \{1,\dots,n\}$.

Decryption: We decrypt a received ciphertext $\bar{\mathbf{y}}$ to 0 if $\operatorname{frc}(\langle\bar{\mathbf{y}},\mathbf{u}\rangle) \leq \tilde{c}\sqrt{\log n}\,\|\mathbf{u}\|$ and to 1 otherwise, where $\tilde{c}$ is a constant given in [4]. This operation is also done with precision $2^{-n\log n}$.

Summarizing the results on A05, he mentioned the following theorem in [4]. Since the ciphertexts of A05 are rounded with precision of $1/n$ and use a compact representation of lattices, the ciphertexts and the keys can be represented by $O(n\log n)$ bits. For the definition of the underlying problem DA′, see Appendix E.

Theorem C.1 ([4]). *The cryptosystem* A05 *encrypts a* 1*-bit plaintext into an* $O(n\log n)$*-bit ciphertext with decryption error probability at most* $\tilde{O}(n^{-r/3})$*. The security of* A05 *is based on the average case of* DA′*. The size of the public key and the private key is* $O(n\log n)$*.*

We show the multi-bit cryptosystem mA05 as follows. Let λ be the length of the unique non-zero shortest vector $\mathbf{u}$, i.e., $\lambda = \|\mathbf{u}\|$. We generalized the standard deviation of n-dimensional Gaussian distribution in encryption procedure for the sake of a discuss of a pseudohomomorphism. We use $\nu_s^{(n)}(\mathbf{x}) = s^{-n}\exp(-\pi\|\mathbf{x}/s\|^2)$ instead of $\nu^{(n)}$ in the original cryptosystem. If we set $s = 1$, the security of our cryptosystem is based on the security of the original one. We suppose that $\eta(n) = \omega(\sqrt{\log n})$ is a parameter to control a trade-off between decryption errors and size of plaintexts and $1/n$ is the precision of rounding in the encryption procedure as same as in the original. To guarantee the decryption errors, we suppose that $s > \sqrt{\lambda}/\eta(n)$. Let a prime p be the size of plaintext space such that $p < n^{r/6}/(4s\eta(n))$. Note that $p \le 1/(4\sqrt{\lambda}s\eta(n))$.

Preparation: All the participants agree with the parameters n and s, and the size p of the plaintext space. They also share the algorithms $\mathcal{B}, \mathcal{D}$ and the random string b.

Key Generation: This procedure is the same as that of A05 except that we add an index i_1 chosen uniformly at random from $\{i : \langle \mathbf{f}_i, \mathbf{u}\rangle \not\equiv 0 \bmod p\}$ to the public key and $k \equiv \langle \mathbf{f}_{i_1}, \mathbf{u}\rangle \bmod p$ to the private key. Thus, the private key is $(\mathbf{u}, k)$ and the public key is (t, i_1).

Encryption: Let $\sigma \in \{0, \ldots, p-1\}$ be a plaintext. We choose $\mathbf{z}$ from the Gaussian distribution $\nu_s^{(n)}$. Then, the ciphertext $\bar{\mathbf{y}}$ is obtained by rounding $\mathbf{y} = \frac{\sigma}{p}\mathbf{f}_{i_1} + \mathbf{z} \bmod \mathcal{P}(F)$ with the precision of $1/n$, i.e., we have $|\bar{y}_i - y_i| \le 1/n$ for every $i \in \{1, \ldots, n\}$.

Decryption: We decrypt a received ciphertext $\bar{\mathbf{y}}$ into $\lceil p\langle \bar{\mathbf{y}}, \mathbf{u}\rangle \rfloor k^{-1} \bmod p$, where k^{-1} is the inverse of k in $\mathbb{Z}_p$.

Before evaluating the performance of mA05 precisely, we give the summary of the results as follows.

Theorem C.2. *The cryptosystem* mA05 *encrypts a $\lfloor \log p(n)\rfloor$-bit plaintext into an $O(n\log n)$-bit ciphertext with decryption error probability at most $2^{-\Omega(\eta^2(n))}$, where $p < n^{r/6}/(4s\eta(n))$ and $s > \sqrt{\lambda}/\eta(n)$. The security of* mA05 *is based on the security of* A05*. The size of the public key is the same as that of the original one. The size of the private key is $\lceil \log p\rceil$ plus that of the original one.*

Setting $\eta(n) = \log n$, we obtain an $O(\log n)$-bit cryptosystem with negligible decryption errors.

Finally, we discuss the pseudohomomorphic property of mA05. We consider a modified version mA05′ of our multi-bit mA05 is the same cryptosystem as mA05 except that the precision is $2^{-n\log n}$ for its ciphertexts instead of $1/n$. This modified version mA05′ actually has the pseudohomomorphism. We denote by E_{m}^s the encryption function of mA05′ such that we use the Gaussian distribution with standard deviation s in the encryption procedure.

Theorem C.3 (pseudohomomorphism)**.** *Let p be a prime and κ be an integer such that $\kappa p < n^{r/6}/(4\eta(n))$ for any constant $r > 0$. We can decrypt the sum of κ ciphertexts $\sum_{i=1}^{\kappa} E_{\mathrm{m}}^1(\sigma_i) \bmod \mathcal{P}(F)$ into $\sum_{i=1}^{\kappa}\sigma_i \bmod p$ with decryption error probability at most $2^{-\Omega(\eta^2(n))}$. Moreover, if there exist two sequences of plaintexts $(\sigma_1, \ldots, \sigma_\kappa)$ and $(\sigma'_1, \ldots, \sigma'_\kappa)$, and a polynomial-time algorithm that distinguishes between $\sum_{i=1}^{\kappa} E_{\mathrm{m}}^1(\sigma_i) \bmod \mathcal{P}(F)$ and $\sum_{i=1}^{\kappa} E_{\mathrm{m}}^1(\sigma'_i) \bmod \mathcal{P}(F)$ with its public key, then there exists a polynomial-time algorithm that solves* DA′ *with non-negligible probability.*

In what follows, we demonstrate the performance of mA05 and mA05′ stated in the above theorems.

C.2 Decryption Errors of mA05

We now give the decryption errors of our multi-bit version mA05.

Theorem C.4. *The probability of the decryption errors in* mA05 *is at most* $2^{-\Omega(\eta^2(n))}$.

Proof. Let $\bar{\mathbf{y}}$ be a ciphertext of a plaintext σ. It is enough to show

$$\Pr\left[\mathrm{frc}\left(\langle\bar{\mathbf{y}},\mathbf{u}\rangle - \frac{k\sigma}{p}\right) > \frac{1}{2p}\right] \leq 2^{-\Omega(\eta^2(n))}.$$

Since $p < 1/(4\sqrt{\lambda}s\eta(n))$ and $\sqrt{\lambda}s\eta(n) > \lambda$,

$$\begin{aligned}\Pr\left[\mathrm{frc}\left(\langle\bar{\mathbf{y}},\mathbf{u}\rangle - \frac{k\sigma}{p}\right) > \frac{1}{2p}\right] &\leq \Pr\left[\mathrm{frc}\left(\langle\bar{\mathbf{y}},\mathbf{u}\rangle - \frac{k\sigma}{p}\right) > 2\sqrt{\lambda}s\eta(n)\right] \\ &\leq \Pr\left[\mathrm{frc}\left(\langle\bar{\mathbf{y}},\mathbf{u}\rangle - \frac{k\sigma}{p}\right) > \sqrt{\lambda}s\eta(n) + \lambda\right].\end{aligned}$$

By the rounding precision of $1/n$, we also have $|\langle(\bar{\mathbf{y}} - \mathbf{y}),\mathbf{u}\rangle| \leq \lambda$. Therefore, we have

$$\begin{aligned}\Pr\left[\mathrm{frc}\left(\langle\bar{\mathbf{y}},\mathbf{u}\rangle - \frac{k\sigma}{p}\right) > \sqrt{\lambda}s\eta(n) + \lambda\right] &\leq \Pr\left[\mathrm{frc}\left(\langle\mathbf{y},\mathbf{u}\rangle - \frac{k\sigma}{p}\right) > \sqrt{\lambda}s\eta(n)\right] \\ &\leq \Pr_{\mathbf{z}\sim\nu_s^{(n)}}\left[\mathrm{frc}\left(\langle\mathbf{z},\mathbf{u}\rangle\right) > \sqrt{\lambda}s\eta(n)\right] + 2^{-\Omega(n)}.\end{aligned}$$

(In the last inequality, we use the fact that $\mathbf{y} = \mathbf{z} + \frac{\sigma}{p}\mathbf{f}_{i'_0} \bmod \mathcal{P}(F)$ and $k \equiv \left\langle\mathbf{f}_{i'_0},\mathbf{u}\right\rangle \bmod p$.) Notice that the fractional part of $\langle\mathbf{z},\mathbf{u}\rangle$ then has a folded Gaussian distribution $\Psi_{\sqrt{\lambda}s}$. (Recall that its density function Ψ_σ is of the form $\Psi_\sigma(l) = \sum_{k\in\mathbb{Z}}(1/\sigma)\exp\left(-\pi((l-k)/\sigma)^2\right)$.) By Lemma A.10, we have

$$\Pr_{\mathbf{z}\sim\nu_s^{(n)}}\left[\mathrm{frc}\left(\langle\mathbf{z},\mathbf{u}\rangle\right) > \sqrt{\lambda}s\eta(n)\right] \leq \frac{1}{\pi\eta(n)}\exp\left(-\pi\eta^2(n)\right).$$

This completes the proof. □

C.3 Security of mA05

The security of our cryptosystem mA05 can be also proven by a similar technique to $\mathrm{mAD}_{\mathrm{GGH}}$.

Theorem C.5. *If there exist plaintexts $\sigma_1, \sigma_2 \in \{0,\ldots,p-1\}$ and a polynomial-time algorithm that distinguishes between the ciphertext of σ_1 and σ_2 in* mA05 *with its public key, there exists a polynomial-time algorithm that distinguishes between the ciphertexts of* 0 *and* 1 *in* A05 *with its public key.*

C.4 Pseudohomomorphism of mA05′

Decryption Errors for Sum of Ciphertexts.

C.4.1 Evaluation for mA05′

Recall that we adopt the precision of $2^{-n\log n}$ for the ciphertexts in mA05′. We denote by E_{m}^s the encryption function of mA05′ such that we use the Gaussian distribution with standard deviation s in the encryption procedure.

Theorem C.6 (mA05′)**.** *Let $\eta(n) = \omega(\sqrt{\log n})$. Also let p be a prime and κ be an integer such that $\kappa p < n^{r/6}/(4\eta(n))$ for any constant $r > 0$. We can decrypt the sum of κ ciphertexts $\sum_{i=1}^{\kappa} E_{\mathrm{m}}^1(\sigma_i) \bmod \mathcal{P}(F)$ into $\sum_{i=1}^{\kappa}\sigma_i \bmod p$ with decryption error probability at most $2^{-\Omega(\eta^2(n))}$.*

Proof. Since the precision is $2^{-n\log n}$, we can consider $\sum_{i=1}^{\kappa} E_{\mathrm{m}}^{1}(\sigma_i) \bmod \mathcal{P}(F)$ as $E_{\mathrm{m}}^{\sqrt{\kappa}}(\sum_{i=1}^{\kappa} \sigma_i \bmod p)$. Replacing s and p by $\sqrt{\kappa}$ and κp respectively, we can evaluate the decryption errors with the same argument as the proof of Theorem C.4 by the fact that $|\langle \bar{\mathbf{y}} - \mathbf{y}, \mathbf{u}\rangle| \le n\lambda 2^{-n\log n} = 2^{-\Omega(n)}$. □

Security for Sum of Ciphertexts. Combining Lemma 3.11 with the security proof of A05 in [4], we guarantee the security of the sum of ciphertexts in mA05′. Note that we can regard $\sum_{i=1}^{\kappa} E_{\mathrm{m}}^{1}(\sigma_i) \bmod \mathcal{P}(W)$ as $E_{\mathrm{m}}^{\sqrt{\kappa}}(\sum_{i=1}^{\kappa} \sigma_i \bmod p)$ in mA05′ by replacing the precision $1/n$ of the ciphertexts to $2^{-n\log n}$.

Theorem C.7. *If there exist two sequences of plaintexts $(\sigma_1, \ldots, \sigma_\kappa)$ and $(\sigma'_1, \ldots, \sigma'_\kappa)$ and a polynomial-time algorithm $\mathcal{D}_1$ that distinguishes between $(\sum_{i=1}^{\kappa} E_{\mathrm{m}}^{1}(\sigma_i), pk)$ and $(\sum_{i=1}^{\kappa} E_{\mathrm{m}}^{1}(\sigma'_i), pk)$, then there exists a probabilistic polynomial-time algorithm $\mathcal{A}$ that solves* DA′.

D Proof of Theorem 3.2

For the proof of Theorem 3.2, we first describe the transference theorems.

D.1 Transference theorems

Let $B(r)$ be an n-dimensional ball in $\mathbb{R}^n$ centered at $\mathbf{0}$ with radius r, i.e., $B(r) = \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\| \le r\}$.

Definition D.1 (Minkowski's successive minima)**.** *For an n-dimensional lattice L in $\mathbb{R}^n$ the i-*th successive minima *$\lambda_i(L)$ is defined as follows:*

$$\lambda_i(L) = \min_{\mathbf{v}_1,\ldots,\mathbf{v}_i \in L} \max_{1\le j\le i} \left\|\mathbf{v}_j\right\|,$$

where the sequence of vectors $\mathbf{v}_1, \ldots, \mathbf{v}_i \in L$ ranges over all i linearly independent lattice vectors.

It is not hard to show that

$$\lambda_i(L) = \min\{r : \max_{\mathbf{v}_1,\ldots,\mathbf{v}_i \in L\cap B(r)} \dim(\mathrm{span}(\mathbf{v}_1, \ldots, \mathbf{v}_i)) = i\}.$$

Banaszczyk showed the following transference theorem in [7].

Theorem D.2 ([7])**.** *For every n-dimensional lattice L and every constant $c > 3/2\pi$,*

$$\lambda_i(L) \cdot \lambda_{n-i+1}(L^*) \le cn,$$

for all sufficiently large n.

We say a sublattice $L' \subseteq L$ is a *saturated sublattice* if $L' = L \cap \mathrm{span}(L')$, where $\mathrm{span}(L')$ is the linear subspace of $\mathbb{R}^n$ spanned by the basis of L'. For $1 \le i \le n$, we define $g_i(L)$ to be the minimum r such that the sublattice generated by $L \cap B(r)$ contains an i-dimensional saturated sublattice L'. Clearly, $\lambda_i(L) \le g_i(L)$ for $1 \le i \le n$.

Cai improved Theorem D.2 as the following theorem.

Theorem D.3 ([9])**.** *For every an n-dimensional lattice L and for every constant $c > 3/2\pi$,*

$$\lambda_i(L) \cdot g_{n-i+1}(L^*) \le cn,$$

for all sufficiently large n.

D.2 Proof of Theorem 3.2

Now, we give the proof of Theorem 3.2.

Proof of Theorem 3.2. The proof is similar to the argument of [5, 6]. Let $H_{\mathbf{u}}$ be the distribution of $\mathbf{v}_i$ in the key generation procedure of $\mathrm{AD}_{\mathrm{GGH}}$. Ajtai and Dwork gave the following two lemmas.

Lemma D.4 (Lemma 8.1, [6])**.** *If there exists a probabilistic polynomial-time algorithm $\mathcal{D}_1$ such that distinguishes between $E(0)$ and $U_{\mathcal{P}(W)}$ with (V, W), there exists a probabilistic polynomial-time algorithm $\mathcal{D}_2$ such that distinguishes between $H_{\mathbf{u}}$ and U_C, where U_C is an uniform distribution on C.*

Lemma D.5 (Lemma 8.2, [6])**.** *If there exists a probabilistic polynomial-time algorithm $\mathcal{D}_2$ such that distinguishes between $H_{\mathbf{u}}$ and U_C, there exists a probabilistic polynomial-time algorithm $\mathcal{A}$ such that solve the worst case of $f(n)$-*uSVP*.*

We now evaluate the value of $f(n)$. Given an instance of $f(n)$-uSVP, we obtain a lattice L by certain linear transformations shown in [6] such that we can efficiently compute its shortest vector $\mathbf{u}$ if there exists an efficient attacking algorithm for $\mathrm{AD}_{\mathrm{GGH}}$. Then, the dual lattice $J = L^*$ of L has a saturated sublattice J' on a hyperplane H_0 orthogonal to $\mathbf{u}$. Let l be the length of the smallest basis of J', where the length of the basis $\mathbf{B} = (\mathbf{v}_1, \ldots, \mathbf{v}_n)$ is defined as $\max_{i=1,\ldots,n} \|\mathbf{v}_i\|$.

It is also commented in [6] that the length l of the smallest basis of J' is approximately $O(n^2/f(n))$. It also holds that this upper bound is $O(n^{-r-3})$ by combining the argument in [6] with our generalization. Thus, we obtain $f(n) = O(n^{r+5})$.

On the other hand, we obtain $\lambda_2(L) \cdot g_{n-1}(L^*) \leq cn$ by Theorem D.3 with $i = 2$, i.e., $\lambda_2(L) \cdot l \leq cn$ for some constant $c > 3/2\pi$. We can also see that $\lambda_2(L) \geq f(n) \|\mathbf{u}\|$ from the definition. Thus, we can obtain an upper bound $O(n/f(n))$ of l.

By the above argument, we obtain $f(n) = O(n^{r+4})$, which completes the proof of Theorem 3.2. □

E Lattice Problems and Their Complexity

We list up well-known hard problems used for lattice-based cryptosystems. The length of vectors is defined by the l_2 norm in this paper.

The shortest vector problem (SVP) and its approximation version (SVP_γ) have been deeply studied in the computer science.

Definition E.1 (SVP)**.** *Given a basis $\mathbf{B}$ of a lattice L, find a non-zero vector $\mathbf{v} \in L$ such that for any non-zero vector $\mathbf{x} \in L$, $\|\mathbf{v}\| \leq \|\mathbf{x}\|$.*

Definition E.2 (SVP_γ)**.** *Given a basis $\mathbf{B}$ of a lattice L, find a non-zero vector $\mathbf{v} \in L$ such that for any non-zero vector $\mathbf{x} \in L$, $\|\mathbf{v}\| \leq \gamma \|\mathbf{x}\|$.*

The NP-hardness of SVP was shown by Ajtai [3] under a randomized reduction in 1998. Recently, Khot [20] proved that SVP_c is NP-hard under the assumption NP $\not\subseteq$ RP for any constant c. He also proved that $\mathrm{SVP}_{2^{O((\log n)^{1/2-\varepsilon})}}$ is NP-hard within under the assumption NP $\not\subseteq$ RTIME$(2^{\mathrm{poly}(\log n)})$.

Even within a polynomial approximation factor, it is not known whether there exists a polynomial-time algorithm for the approximation version of SVP. The most well-known solution to this approximation problem is the so-called LLL algorithm proposed in [23]. This algorithm can solve $\mathrm{SVP}_{2^{n/2}}$ in polynomial time.

On the other hand, there are several non-NP-hardness results on the approximation version of SVP with a polynomial approximation factor. Goldreich and Goldwasser [12] showed $\mathrm{SVP}_{\Omega(\sqrt{n/\log n})}$ is in NP $\cap$ coAM. Aharonov and Regev [1] showed that $\mathrm{SVP}_{\Omega(\sqrt{n})}$ is in NP $\cap$ coNP.

The unique shortest vector problem (uSVP) is also well known as a hard lattice problem applicable to cryptographic constructions. We say the shortest vector $\mathbf{v}$ of a lattice L is f-unique if for any non-zero vector $\mathbf{x} \in L$ which is not parallel to $\mathbf{v}$, $f\,\|\mathbf{v}\| \leq \|\mathbf{x}\|$. The definition of uSVP is given as follows.

Definition E.3 (f-uSVP). *Given a basis* $\mathbf{B}$ *of a lattice* L *whose shortest vector is* f*-unique, find a non-zero vector* $\mathbf{v} \in L$ *such that for any non-zero vector* $\mathbf{x} \in L$ *which is not parallel to* $\mathbf{v}$, $f\,\|\mathbf{v}\| \leq \|\mathbf{x}\|$.

Similarly to the case of SVP, the exact version of uSVP is shown to be in NP-hard by Kumar and Sivakumar [21]. Cai [8] showed that $\Omega(n^{1/4})$-uSVP is in NP $\cap$ coAM. See Figure 9 for the hardness of SVP and uSVP.

In the computational complexity theory on lattice problems, the shortest linearly independent vectors problem (SIVP) and its approximation version SIVP_γ are also considered as a hard lattice problem.

Definition E.4 (SIVP). *Given a basis* $\mathbf{B}$ *of a lattice* L*, find a sequence of* n *linearly independent vectors* $\mathbf{v}_1, \ldots, \mathbf{v}_n \in L$ *such that for any sequence of* n *linearly independent vectors* $\mathbf{x}_1, \ldots, \mathbf{x}_n \in L$, $\max_{i=1,\ldots,n} \|\mathbf{v}_i\| \leq \max_{i=1,\ldots,n} \|\mathbf{x}_i\|$.

Definition E.5 (SIVP_γ). *Given a basis* $\mathbf{B}$ *of a lattice* L*, find a sequence of* n *linearly independent vectors* $\mathbf{v}_1, \ldots, \mathbf{v}_n \in L$ *such that for any sequence of* n *linearly independent vectors* $\mathbf{x}_1, \ldots, \mathbf{x}_n \in L$, $\max_{i=1,\ldots,n} \|\mathbf{v}_i\| \leq \gamma \max_{i=1,\ldots,n} \|\mathbf{x}_i\|$.

Although the Diophantine Approximation (DA) was originally a number-theoretic problem, DA is deeply related to the lattice theory. (See, e.g., [16].) The problem DA is defined as follows.

Definition E.6 (DA). *Given* n *real numbers* $r_1, \ldots, r_n$ *and an integer* M*, find an integer* $m \in [1, M^n]$ *such that* $\max_{i=1}^{n} \operatorname{frc}(mr_i) \leq 1/M$.

From a complexity-theoretical point of view, Lagarias [22] showed that decisional version of DA is NP-complete. Trolin [36] also showed a reduction from the decisional version of DA to a certain lattice problem. In the context of cryptography, Ajtai defined a variant of DA and constructed an efficient lattice-based cryptosystem based on the hardness of this variant [4]. We refer to this variant as DA′, defined as follows.

Definition E.7 (DA′, [4]). *Let* $c_1, c_2 > 0$ *be constants. Assume that* $r_1, \ldots, r_n$ *are samples from the uniform distribution on* $(0, 1)$ *with the condition that there exists an integer* m *such that*

$$1 \leq m \leq n^{c_1 n} \text{ and } \operatorname{frc}(mr_i) \leq n^{-(c_1+c_2)} \text{ for } i = 1, \ldots, n.$$

Given n, $r_1, \ldots, r_n$, c_1, c_2*, find such an integer* m*.*

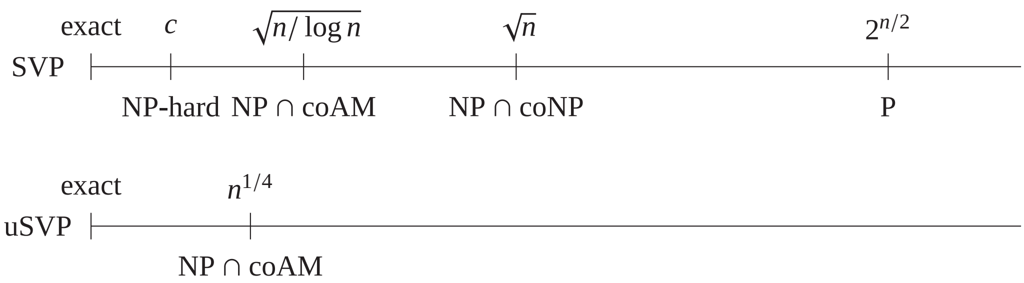

Figure 9: the hardness of SVP and uSVP