# Multi-Bit Cryptosystems Based on Lattice Problems

Akinori Kawachi[1], Keisuke Tanaka[1], and Keita Xagawa[1]

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

**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.

## 1 Introduction

*Lattice-Based Cryptosystems.* The lattice-based cryptosystems have been well-studied since Ajtai's seminal result [1] 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 [2] 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 [3,4,5,6].

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 [7], NTRU [8] and their improvements [9,10,11] 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 [12,13].

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) [2,4,6]. 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 attractive property of the average-case/worst-case connection has been also studied from a theoretical point of view [1,14,15,16].

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 [17]. 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 [6]. 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 [5], 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 [3], the Regev cryptosystems given in [4] and in [6], and the Ajtai cryptosystem [5]. 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.

| | Ajtai-Dwork | | Regev'04 | |
|---|---|---|---|---|
| cryptosystem | $\mathrm{AD}_{\mathrm{GGH}}$ [3] | $\mathrm{mAD}_{\mathrm{GGH}}$ | R04 [4] | 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 [6] | mR05 | A05 [5] | 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. ($\varepsilon$ is any positive constant and $\tilde{O}(f(n))$ means $O(f(n)\,\mathrm{poly}(\log n))$.)

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. We call the cryptosystems proposed in [3,4,6,5] $\mathrm{AD}_{\mathrm{GGH}}$, R04, R05, and A05, respectively. We also call the corresponding multi-bit versions $\mathrm{mAD}_{\mathrm{GGH}}$, mR04, mR05, and mA05.

The problems in the security fields are deeply related to lattice problems. The shortest vector problem within approximation factor $\gamma$ ($\mathrm{SVP}_\gamma$) is generally considered as a hard problem for polynomial factor of $\gamma$, which is defined as follows. Given a lattice $L$, the problem is to find a shortest non-zero vector $\boldsymbol{u} \in L$ within approximation factor $\gamma$.

The unique shortest vector problem (uSVP) is also well known as a hard lattice problem applicable to cryptographic constructions. We say the shortest vector $\boldsymbol{u}$ of a lattice $L$ is $f$-unique if for any non-zero vector $\boldsymbol{v} \in L$ which is not parallel to $\boldsymbol{u}$, $f\,\|\boldsymbol{u}\| \le \|\boldsymbol{v}\|$. Given a lattice $L$ whose shortest vector is $f$-unique, the problem is to find a non-zero vector $\boldsymbol{u} \in L$ such that for any non-zero vector $\boldsymbol{v} \in L$ which is not parallel to $\boldsymbol{u}$, $f\,\|\boldsymbol{u}\| \le \|\boldsymbol{v}\|$.

While the security of $\mathrm{AD}_{\mathrm{GGH}}$, R04, and R05 is based on the above two lattice problems, that of A05 is on a variant of Diophantine approximation problem (DA′). See [5] for the definition of this problem.

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., [18].) 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 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 [19].

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 $\mathrm{mAD}_{\mathrm{GGH}}$ over larger cyclic groups. The property of mR04, mR05, and (a slightly modified version of) mA05 can be shown similarly, whose proof will be given in the full paper. 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 [3]. We therefore have to evaluate the trade-offs in our multi-bit versions between the decryption errors and their security, which depend on their own structures of the cryptosystems.

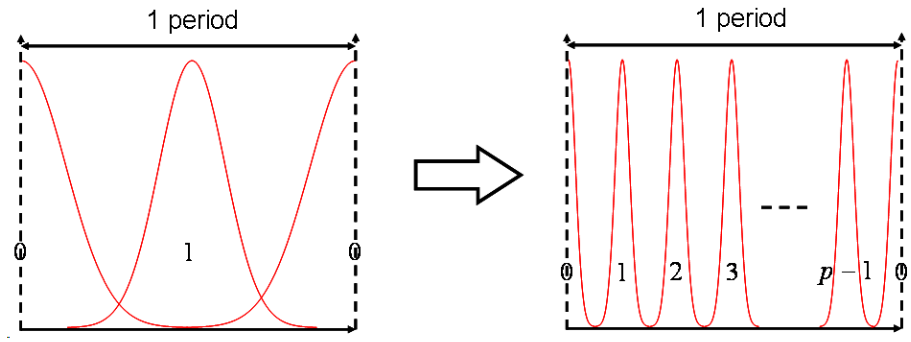


**Fig. 1.** the embedding of periodic Gaussian distributions.

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 $\Phi$ 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 $\Phi$ 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 $\Phi_i$ and $U$ or between $\Phi_j$ and $U$. We now suppose that it can distinguish between $\Phi_i$ and $U$. Note that we can slide $\Phi_i$ to $\Phi_0$ 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 $\Phi_0$ and $U$. $\Phi_0$ is in fact a variance-reduced version of the periodic Gaussian distribution $\Phi$ used in the original cryptosystem. We can guarantee the indistinguishability between such a version $\Phi_0$ 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 omit the description of the other multi-bit versions mR04, mR05, and mA05 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. They will appear in the full paper. We also give concluding remarks in Section 4.

## 2 Basic Notions and Notations

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

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.

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.

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

For more details on lattices, see the textbook by Micciancio and Goldwasser [20].

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

On behalf of four cryptosystems $\mathrm{AD}_{\mathrm{GGH}}$, R04, R05, and A05, we discuss the improved Ajtai-Dwork cryptosystem $\mathrm{AD}_{\mathrm{GGH}}$ given by Goldreich, Goldwasser, and Halevi [3] in detail and apply our technique to construction of its multi-bit version $\mathrm{mAD}_{\mathrm{GGH}}$ 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 $\mathrm{AD}_{\mathrm{GGH}}$. Let $N = n^n = 2^{n \log n}$. We define an $n$-dimensional hypercube $C$ and an $n$-dimensional ball $B_r$ as $C = \{\boldsymbol{x} \in \mathbb{R}^n : 0 \le x_i < N, i = 1, \ldots, n\}$ and $B_r = \{\boldsymbol{x} \in \mathbb{R}^n : \|\boldsymbol{x}\| \le n^{-r}/4\}$ for any constant $r \ge 7$, respectively. For $\boldsymbol{u} \in \mathbb{R}^n$ and an integer $i$ we define a hyperplane $H_i$ as $H_i = \{\boldsymbol{x} \in \mathbb{R}^n : \langle \boldsymbol{x}, \boldsymbol{u} \rangle = i\}$.

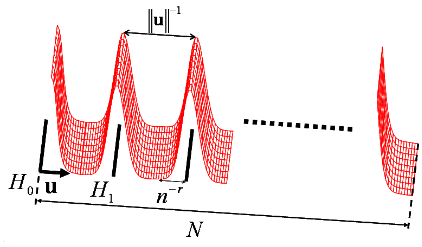


**Fig. 2.** ciphertexts of 0 in $\mathrm{AD}_{\mathrm{GGH}}$

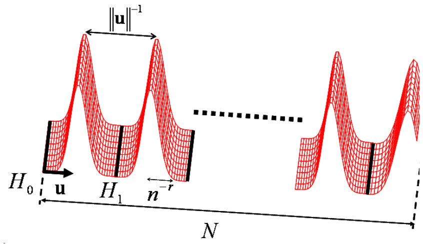


**Fig. 3.** ciphertexts of 1 in $\mathrm{AD}_{\mathrm{GGH}}$

Roughly speaking, $\mathrm{AD}_{\mathrm{GGH}}$ encrypts 0 into a vector distributed closely around hidden $(n-1)$-dimensional parallel hyperplanes $H_0, H_1, H_2, \ldots$ for a normal vector $\boldsymbol{u}$ of $H_0$, and encrypts 1 into a vector distributed closely around their intermediate parallel hyperplanes $H_0 + \boldsymbol{u}/(2\|\boldsymbol{u}\|^2), H_1 + \boldsymbol{u}/(2\|\boldsymbol{u}\|^2), \ldots$. (See Figure 2 and Figure 3.) Then, the private key is the normal vector $\boldsymbol{u}$. 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 $\mathrm{AD}_{\mathrm{GGH}}$ 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 $\boldsymbol{u}$ uniformly at random from the $n$-dimensional unit ball. Let $m = n^3$. Repeating the following procedure $m$ times, we sample $m$ vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m$: (1) We choose $\boldsymbol{a}_i$ from $\{\boldsymbol{x} \in C : \langle \boldsymbol{x}, \boldsymbol{u} \rangle \in \mathbb{Z}\}$ uniformly at random, (2) choose $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ from $B_r$ uniformly at random, (3) and output $\boldsymbol{v}_i = \boldsymbol{a}_i + \sum_{j=1}^{n} \boldsymbol{b}_j$ as a sample. We then take the minimum index $i_0$ satisfying that the width of $\mathcal{P}(\boldsymbol{v}_{i_0+1}, \ldots, \boldsymbol{v}_{i_0+n})$ is at least $n^{-2}N$, where width of a parallelepiped $\mathcal{P}(\boldsymbol{x}_1, \ldots \boldsymbol{x}_n)$ is defined as $\min_{i=1,\ldots,n} \mathrm{Dist}(\boldsymbol{x}_i, \mathrm{span}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{i-1}, \boldsymbol{x}_{i+1}, \ldots, \boldsymbol{x}_n))$ for a distance function $\mathrm{Dist}(\cdot, \cdot)$ between a vector and an $(n-1)$-dimensional hyperplane.

Now let $\boldsymbol{w}_j = \boldsymbol{v}_{i_0+j}$ for every $j \in \{1, \ldots, n\}$, $V = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m)$, and $W = (\boldsymbol{w}_1, \ldots, \boldsymbol{w}_n)$. We also choose an index $i_1$ uniformly at random from $\{i : \langle \boldsymbol{a}_i, \boldsymbol{u} \rangle$ is odd$\}$, where $\boldsymbol{a}_i$ is the vector appeared in the sampling procedure for $\boldsymbol{v}_i$. 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, $\|\boldsymbol{u}\|$ 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 $\boldsymbol{u}$ again. Then, the private key is $\boldsymbol{u}$ 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 $\boldsymbol{x} = \frac{\sigma}{2}\boldsymbol{v}_{i_1} + \sum_{i \in S} \boldsymbol{v}_i \bmod \mathcal{P}(W)$.

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

Carefully reading the results in [2,3], we obtain the following theorem on the cryptosystem $\mathrm{AD_{GGH}}$.

**Theorem 1 ([3]).** *The cryptosystem* $\mathrm{AD_{GGH}}$ *encrypts a* 1*-bit plaintext into an* $n\lceil n(\log n + 1)\rceil$*-bit ciphertext with no decryption errors. 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 [21], we can actually improve the security of $\mathrm{AD_{GGH}}$ by a result in [21]. We will give the proof in the full paper.

**Theorem 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 $\{i : \langle \boldsymbol{a}_i, \boldsymbol{u} \rangle \not\equiv 0 \bmod p\}$ instead of $i_1$ in the original key generation procedure. We set decryption information $k \equiv \langle \boldsymbol{a}_{i'_1}, \boldsymbol{u} \rangle \bmod p$. Note that there is such a $k$ with probability $1 - (1/p)^m = 1 - o(1)$. Then, the private key is $(\boldsymbol{u}, k)$ 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 $\boldsymbol{x} = \frac{\sigma}{p}\boldsymbol{v}_{i'_1} + \sum_{i \in S} \boldsymbol{v}_i \bmod \mathcal{P}(W)$.

**Decryption:** We decrypt a received ciphertext $\boldsymbol{x} \in \mathcal{P}(W)$ to $\lfloor p \langle \boldsymbol{x}, \boldsymbol{u} \rangle \rceil k^{-1} \bmod p$, 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 (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 4 (pseudohomomorphism).** *Let* $r \geq 7$ *be any constant. Also, let* $p$ *be a prime and let* $\kappa$ *be an integer such that* $\kappa p \leq n^{r-7}$*. Let* $E_{\mathrm{m}}$ *be the encryption function of* $\mathrm{mAD_{GGH}}$*. For any* $\kappa$ *plaintexts* $\sigma_1, \ldots, \sigma_\kappa$ $(0 \leq \sigma_i \leq p-1)$*, we can decrypt the sum of* $\kappa$ *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 [2,3]. Since we generalize this theorem for analysis of the pseudohomomorphism in $\mathrm{mAD_{GGH}}$ (Theorem 7), we here give a precise proof.

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

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

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

Now let $\boldsymbol{\rho}$ be a ciphertext of $\sigma$. 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 $\langle \boldsymbol{v}_{i'_1}/p, \boldsymbol{u} \rangle \in \mathbb{Z} + k/p \pm n^{1-r}/4p$ and

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

Therefore, we obtain $\langle \boldsymbol{\rho}, \boldsymbol{u} \rangle \in \mathbb{Z} + k\sigma/p \pm 1/(2p)$ and decrypt $\boldsymbol{\rho}$ into $\sigma$ 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 $\sigma$ 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 $E_{\mathrm{m}}$ 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 2 and the results in [2] and [3] with our generalization.

**Lemma 1 ([2,3]).** *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 2.** *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 $\boldsymbol{x}$ 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 $\mathcal{D}_1(\boldsymbol{x}, (V, W, j'))$ if we find $j'$. Otherwise, we output a uniformly random bit.

Note that we have $|p_0 - x_0(j')/T| \le \varepsilon/4$ and $|p_U - x_U(j')/T| \le \varepsilon/4$ with probability exponentially close to 1 by the Hoeffding bound [22]. 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.** *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 $E_{\mathrm{m}}(\sigma_1, (V, W, i_1')) = E_{\mathrm{m}}(0, (V, W, i_1')) + \frac{\sigma_1}{p}\boldsymbol{v}_{i_1'} \bmod \mathcal{P}(W)$ by the definition of $E_{\mathrm{m}}$. Then, we can transform a given $\boldsymbol{x}$ from $E_{\mathrm{m}}(0, (V, W, i_1'))$ to another sample $\boldsymbol{y}$ 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 $\mathrm{mAD_{GGH}}$.

**Theorem 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 $\sigma_1$ and $\sigma_2$ of* $\mathrm{mAD_{GGH}}$ *with its public key, there exists a polynomial-time algorithm for the worst-case of $O(n^{r+4})$*-uSVP *for $r \ge 7$.*

### 3.4 Pseudohomomorphism of $\mathrm{mAD_{GGH}}$

As stated in Theorem 4, $\mathrm{mAD_{GGH}}$ 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 $\mathrm{mAD_{GGH}}$. Recall that $\mathbb{Z} \pm a := \{x \in \mathbb{R} : \mathrm{frc}\,(x) \le a\}$ for $a \ge 0$ and $\mathbb{Z} + a \pm b := \{x \in \mathbb{R} : \mathrm{frc}\,(x - a) \le b\}$ for $a, b \ge 0$.

**Theorem 7.** *Let $r \ge 7$ be any constant. Also let $p$ be a prime and $\kappa$ be an integer such that $\kappa p \le n^{r-7}$. For any $\kappa$ plaintexts $\sigma_1, \ldots, \sigma_\kappa$ $(0 \le \sigma_i \le p-1)$, we can decrypt the sum of $\kappa$ 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 $\boldsymbol{\rho}_1, \dots, \boldsymbol{\rho}_\kappa$ as ciphertexts of $\sigma_1, \dots, \sigma_\kappa$, respectively. We will show that we can decrypt $\boldsymbol{\rho} := \sum_{i=1}^{\kappa} \boldsymbol{\rho}_i \bmod \mathcal{P}(W)$ into $\sum_{i=1}^{\kappa} \sigma_i \bmod p$. From the proof of Theorem 5, we have

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

Hence, we obtain

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

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

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

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

*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 $E_{\mathrm{m}}$, also regarded as a random variable $E_{\mathrm{m}}(\sigma, pk)$ for a plaintext $\sigma$ and a public key $pk$. If the public key is obvious, we abbreviate $E_{\mathrm{m}}(\sigma, pk)$ to $E_{\mathrm{m}}(\sigma)$. Let $\mathcal{C}$ be the ciphertext space and $U_{\mathcal{C}}$ be the uniform distribution on $\mathcal{C}$.

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 $\kappa$ samples from $E_{\mathrm{m}}(0)$ and those from $U_{\mathcal{C}}$.

**Lemma 4.** *If there exist two sequences of plaintexts $(\sigma_1, \dots, \sigma_\kappa)$ and $(\sigma'_1, \dots, \sigma'_\kappa)$ 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 $\kappa$ ciphertexts and its public key $(E_{\mathrm{m}}(0, pk), \dots, E_{\mathrm{m}}(0, pk), pk)$ and uniformly random $\kappa$ ciphertexts and the public key $(U_{\mathcal{C}}, \dots, U_{\mathcal{C}}, pk)$.*

*Proof.* By the hybrid argument, the distinguisher $\mathcal{D}_1$ can distinguish between $\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma_i)$ and $U_{\mathcal{C}}$ or between $\sum_{i=1}^{\kappa} E_{\mathrm{m}}(\sigma'_i)$ and $U_{\mathcal{C}}$ 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 $(U_{\mathcal{C}}, pk)$. 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 $\mathcal{D}_2$. $\square$

As already stated in Section 1 (and Lemma 2 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 $U_{\mathcal{C}}$ 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 $(U_{\mathcal{C}}, \ldots, U_{\mathcal{C}}, pk)$. Thus, we obtain from $\mathcal{D}_2$ in Lemma 4 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 4, we guarantee the security of the sum of ciphertexts in $\mathrm{mAD}_{\mathrm{GGH}}$.

**Theorem 8.** *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 [19].

## References


1. Ajtai, M.: Generating hard instances of lattice problems. Electronic Colloquium on Computational Complexity (ECCC) **3**(007) (1996).
2. Ajtai, M., Dwork, C.: A public-key cryptosystem with worst-case/average-case equivalence. In STOC '97 (1997) 284–293. See also ECCC TR96-065.

3. Goldreich, O., Goldwasser, S., Halevi, S.: Eliminating decryption errors in the Ajtai-Dwork cryptosystem. In CRYPTO '97 (1997) 105–111. See also ECCC TR097-018.
4. Regev, O.: New lattice based cryptographic constructions. In STOC 2003 (2003) 407–416.
5. Ajtai, M.: Representing hard lattices with $O(n \log n)$ bits. In STOC 2005 (2005) 94–103.
6. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In STOC 2005 (2005) 84–93.
7. Goldreich, O., Goldwasser, S., Halevi, S.: Public-key cryptosystems from lattice reduction problems. In CRYPTO '97 (1997) 112–131.
8. Hoffstein, J., Pipher, J., Silverman, J.H.: NTRU: A ring-based public key cryptosystem. In ANTS-III (1998) 267–288.
9. Micciancio, D.: Improving lattice based cryptosystems using the Hermite normal form. In CaLC 2001 (2001) 126–145.
10. Nguyen, P.Q.: Analysis and improvements of NTRU encryption paddings. In CRYPTO 2002 (2002) 210–225.
11. Howgrave-Graham, N., Nguyen, P.Q., Pointcheval, D., Proos, J., Silverman, J.H., Singer, A., Whyte, W.: The impact of decryption failures on the security of NTRU encryption. In CRYPTO 2003 (2003) 226–246.
12. Nguyen, P.Q.: Cryptanalysis of the Goldreich-Goldwasser-Halevi cryptosystem from Crypto '97. In CRYPTO '99 (1999) 288–304.
13. Gentry, C.: Key recovery and message attacks on NTRU-composite. In EUROCRYPT 2001 (2001) 182–194.
14. Micciancio, D., Regev, O.: Worst-case to average-case reductions based on Gaussian measures. In FOCS 2004 (2004) 372–181.
15. Micciancio, D.: Generalized compact knapsacks, cyclic lattices, and efficient one-way functions from worst-case complexity assumptions. Electronic Colloquium on Computational Complexity (ECCC) **11**(095) (2004).
16. Peikert, C., Rosen, A.: Efficient collision-resistant hashing from worst-case assumptions on cyclic lattices. In TCC 2006 (2006) 145–166.
17. Nguyen, P.Q., Stern, J.: Cryptanalysis of the Ajtai-Dwork cryptosystem. In CRYPTO '98 (1998) 223–242.
18. Rappe, D.: Homomorphic Cryptosystems and Their Applications. PhD thesis, University of Dortmund (2004). Also available at http://eprint.iacr.org/2006/001.
19. Goldwasser, S., Kharchenko, D.: Proof of plaintext knowledge for the Ajtai-Dwork cryptosystem. In TCC 2005 (2005) 529–555.
20. Micciancio, D., Goldwasser, S.: Complexity of Lattice Problems: a cryptographic perspective. Kluwer Academic Publishers, Boston, Massachusetts (2002).
21. Cai, J.Y.: A new transference theorem in the geometry of numbers and new bounds for Ajtai's connection factor. Discrete Applied Mathematics **126**(1) (2003) 9–31.
22. Hoeffding, W.: Probability inequalities for sums of bounded random variables. Journal of the American Statistical Association **58**(301) (1963) 13–30.