Identity-Based Encryption from Lattices with Small Cipher Size

WANG Ziqing[1], TANG Dianhua[1, 2*], and LI Fagen[1]

(1. School of Computer Science and Engineering, University of Electronic Science and Technology of China Chengdu 611731;
2. The 30th Research Institute of China Electronic Technology Group Corporation Chengdu 610041)

Abstract Identity-based encryption (IBE) is very attractive because it does not have certificate management issues. However, the IBEs based on the bilinear Diffie-Hellman problem cannot resist quantum attacks. In order to ensure security under quantum attacks, lattice-based IBE is proposed. However, the existing lattice-based IBEs usually not only have a large ciphertext size but also can only encrypt a few bits of plaintext information in one ciphertext. In this paper, we propose a new lattice-based IBE scheme based on learning with errors (LWE) and its ring version. For the setting $l = n$, our scheme can encrypt the plaintext twice long of other schemes in one ciphertext. Then we prove that our scheme can achieve the indistinguishability of ciphertexts against adaptively chosen identity and chosen plaintext attack (IND-ID-CPA) in the random oracle.

Key words IBE; lattice-based cryptography; LWE; ring LWE

基于格的较小密文身份加密方案

王子卿[1]，汤殿华[1, 2*]，李发根[1]

(1. 电子科技大学计算机科学与工程学院 成都 611731；2. 中国电子科技集团公司第三十研究所 成都 610041)

【摘要】现有的基于格的 IBE 加密方案的密文较大，在一个密文中只能加密较少的明文信息。为此，提出了一种基于带错误学习问题 (LWE) 及其环版本的新的基于格的 IBE 加密方案。当方案的加密参数设置为$l = n$时，相比于其他基于格的 IBE 加密方案，在相同的密文大小下，该方案可以加密两倍长度的明文。在随机预言机模型下，证明了该方案能够在自适应选择身份攻击和选择明文攻击下实现密文不可区分性 (IND-ID-CPA)。

关 键 词 基于身份的加密； 格密码； 带错误学习问题； 环上带错误学习问题
中图分类号 TP309 **文献标志码** A doi:10.12178/1001-0548.2022007

Identity-Based encryption (IBE) was first proposed by Shamir[1] in 1985. Compared with public key infrastructure (PKI), the public keys of users in IBE schemes are derived directly from identities like e-mail addresses or phone numbers instead of being generated by users. The private keys are generated and distributed to users by a trusted authority (TA) who has a master key. As TA will guarantee that the private keys are sent to the users with the corresponding identities, IBE does not have certificate management problems. The IBE schemes based on the bilinear Diffe-Hellman problem may face the threat of quantum attacks. However, since there is no quantum algorithm that can effectively solve the hard problems of lattice, the lattice-based IBE schemes are considered to be able to maintain the security under quantum attacks. In 2008, Ref. [2] proposed the first lattice-based IBE named GPV, which is based on learning

Received date: 2022 − 01 − 04; Revised date: 2022 − 03 − 01
收稿日期：2022 − 01 − 04；修回日期：2022 − 03 − 01
Foundation item: Supported by Sichuan Science and Technology Program (2021YFG0157)
基金项目：四川省科技计划 (2021YFG0157)
Biography: WANG Ziqing was born in 1997，male，PhD student，his research insterest is lattice-based cryptography.
作者简介：王子卿 (1997 −)，男，博士生，主要从事格密码方面的研究.
*Corresponding author: TANG Dianhua，E-mail：tangdianhua86@163.com
*通信作者：汤殿华，E-mail：tangdianhua86@163.com

with errors (LWE) problem[3]. The scheme achieves indistinguishability of ciphertexts against adaptively chosen identity and chosen plaintext attack (IND-ID-CPA) under random oracle. The random oracle model assumes that the adversary can not attack the hash function and the standard oracle doesnot have this assumption, which means that the standard model has better security. In 2010, Ref. [4] proposed the first lattice-based IBE scheme under the standard model. However, the IBE schemes under the standard model have larger size of parameters and ciphertext than the IBE schemes under the random oracle. In the same year, Ref. [5-6] proposed some more efficient lattice-based IBE schemes under the standard model. The above schemes use the trapdoor generation algorithm to generate a uniformly random matrix and a short basis. The trapdoor generation algorithm was proposed by Ref. [7] in 1999 and then improved by Ref. [8] in 2009. In 2012, Ref. [9] gave a new method to generate trapdoors for lattices. The size of the output matrix of their algorithm is smaller than that of previous work. They also gave the trapdoor generation algorithm that can be used for ring. To reduce the size of public parameter and ciphertext, in 2016, Ref. [10] proposed an IBE scheme based on RLWE. In 2018, Ref. [11] proposed another RLWE-based IBE scheme with smaller ciphertext size than Katsumata's[10]. Ref. [12] proposed a RLWE-based IBE with short parameters in 2020. The encryption of the above schemes can be seen as a variant of dual Regev PKE[2], which makes these schemes have large ciphertexts.

In this paper, we propose a new LWE-based IBE scheme with a small size of ciphertext. We also give the ring version of our scheme. The proposed scheme is IND-ID-CPA secure under the random oracle. Compared with GPV[2], if we set the parameter $l=n$, the length of the plaintext that our scheme can encrypt is twice of GPV's[2] without the increase of ciphertext size.

1 Preliminaries

1.1 Notation

The base of logarithmic used in this paper is always 2. We use $\mathbb{Z}_q$ to represent $\mathbb{Z}\cap[0,q)$ when q is a positive integer. $\lfloor r\rceil$ represents the nearest integer of the real number r. When the value of r is the middle value of the two nearest integers, it is rounded up. $\lfloor r\rfloor$ represents round down operation. We use lowercase bold letters to represent column vectors, such as $\mathbf{a}$. Specifically, $\mathbf{0}$ represents the zero vector. We use $|r|$ to represent the absolute value of the number r. The dot product operation of vectors is represented as $\langle\bullet,\bullet\rangle$. $x\leftarrow D$ means that x is sampled from the distribution D. When D is a finite set, it means being uniformly sampled from D. In particular, we use $D_{\mathbb{Z}_q,\sigma}$ to represent a discrete gaussian distribution with the parameter σ on $\mathbb{Z}_q$. We use capital bold letters to denote matrices. $\|\bullet\|_\infty$ represents the infinite norm and $\|\bullet\|$ represents the Euclidean norm. The Gram-Schmidt order orthogonalization of matrix $\boldsymbol{A}$ is denoted by $\tilde{\boldsymbol{A}}$. We use R_q to represent $\mathbb{Z}_q[x]/(x^n+1)$.

1.2 Lattices and Hard Problems

Given positive integers $q,m,n\in\mathbb{Z}$, and an arbitrary matrix $\boldsymbol{A}\in\mathbb{Z}_q^{n\times m}$, we can define a lattice described as follows:

$$\Lambda_q^{\perp}(\boldsymbol{A})=\{\boldsymbol{x}\in\mathbb{Z}^m:\boldsymbol{A}\boldsymbol{x}=\mathbf{0}\bmod q\}$$

For an arbitrary vector $\boldsymbol{u}\in\mathbb{Z}^n$, a coset of lattice $\Lambda_q^{\perp}(\boldsymbol{A})$ is defined as:

$$\Lambda_q^{\boldsymbol{u}}(\boldsymbol{A})=\{\boldsymbol{x}\in\mathbb{Z}^m:\boldsymbol{A}\boldsymbol{x}=\boldsymbol{u}\bmod q\}$$

In the rest sections, we will use $D_{\Lambda_q^{\perp}(\boldsymbol{A}),s}$ and $D_{\Lambda_q^{\boldsymbol{u}}(\boldsymbol{A}),s}$ to represent the discrete gaussian distribution with the parameter s over $\Lambda_q^{\perp}(\boldsymbol{A})$ and $\Lambda_q^{\boldsymbol{u}}(\boldsymbol{A})$ respectively.

Definition 1[13] (short integer solution (SIS))For positive integers m,n,q, a uniformly random matrix $\boldsymbol{A}\in\mathbb{Z}_q^{n\times m}$and a positive real number β, the goal of the $\mathrm{SIS}_{n,q,\beta,m}$ is to find a nonzero vector $\boldsymbol{x}\in\Lambda_q^{\perp}(\boldsymbol{A})$ which satisfies $\|\boldsymbol{x}\|<\beta$.

It is obviously that when β is small enough, the vector $\boldsymbol{x}$ will be a short vector of $\Lambda_q^{\perp}(\boldsymbol{A})$. To guarantee that the solution vector exists, the parameters m and β should satisfy $\beta\geqslant\sqrt{n\log q}$ and $m\geqslant n\log q$. The inhomogeneous SIS problem is to find a nonzero vector $\boldsymbol{x}\in\Lambda_q^{\boldsymbol{u}}(\boldsymbol{A})$ that satisfies $\|\boldsymbol{x}\|<\beta$, where $\boldsymbol{u}$ is uniformly random and independent with of $\boldsymbol{A}$. It has been proved that the hardness of inhomogeneous SIS

is the same as SIS.

Definition 2[14] (ring short integer solution (RSIS)) For positive integers m,n,q, a uniformly random vector $\boldsymbol{a}\in R_q^m$ and a positive real number β, the goal of the $R-\mathrm{SIS}_{n,q,\beta,m}$ is to find a nonzero vector $\boldsymbol{x}\in R_q^m$ which satisfies $\langle \boldsymbol{a},\boldsymbol{x}\rangle = 0\in \boldsymbol{R}_q$ and has norm $\|\boldsymbol{x}\|<\beta$.

In the RSIS problem, to guarantee that the solution vector exists, m should be no less than $\log q$.

Definition 3[3] (LWE) For integers $n\geqslant 1, q\geqslant 2$, a real number $\alpha\in(0,1)$ and a uniformly random vector $\boldsymbol{s}\in\mathbb{Z}_q^n$, we can define a distribution $\mathrm{LWE}_{n,q,\alpha}$ over $\mathbb{Z}_q^n\times\mathbb{Z}_q$ as this:

$$(\boldsymbol{a},\langle \boldsymbol{a},\boldsymbol{s}\rangle + e)\leftarrow \mathrm{LWE}_{n,q,\alpha}$$

where $\boldsymbol{a}\leftarrow\mathbb{Z}_q^n$ and $e\leftarrow D_{\mathbb{Z}_q,\alpha q}$.

The decision LWE problem can be described as a challenge game between an adversary A and a challenger oracle O. First, given $\mathrm{LWE}_{n,q,\alpha}$, define two types of challenge oracle O_1 and O_2 that output a sample in $\mathbb{Z}_q^n\times\mathbb{Z}_q$:

O_1: The output sample is sampled from $\mathrm{LWE}_{n,q,\alpha}$.

O_2: The output sample is sampled from uniformly random distribution.

The challenger oracle O in the game is either O_1 or O_2. The adversary A can repeat queries of the oracle O to get fresh samples. At the end of the game, A outputs a bit b to guess the type of O. The advantage of A is:

$$\mathrm{Adv}(A)=|\Pr[b=1|O_1]-\Pr[b=1|O_2]|$$

It has been proved that the decision LWE problem is as hard as some worst-case hard problems on n-dimensional lattices under a quantum reduction.

Ref. [15] proved that when $\boldsymbol{s}\leftarrow D_{\mathbb{Z}_q^n,\alpha q}$, the LWE problem is still hard. This form of LWE is called normal form LWE.

Definition 4[16] (ring learning with errors (RLWE)) For integers $n\geqslant 1, q\geqslant 2$, a real number $\alpha\in(0,1)$, a uniformly random element $s\in R_q$, we can define a distribution $\mathrm{RLWE}_{n,q,\alpha}$ over $R_q\times R_q$ as this:

$$(a,as+e)\leftarrow \mathrm{LWE}_{n,q,\alpha}$$

where $a\leftarrow R_q$ and $e\leftarrow D_{R_q,\alpha q}$.

The RLWE problem and its normal form have also been proved as hard as some worst-case hard problems on lattices under a quantum reduction.

1.3 Algorithms

The paper introduces some algorithms for lattices which are important for building a encryption scheme.

Lemma 1[17] For positive integers q,n,m, where $q>2$ and $m\approx 2n\log q$, the PPT algorithm $\mathrm{TrapGen}(n,m,q)$ can output a matrix $\boldsymbol{A}\in\mathbb{Z}_q^{n\times m}$ and a short basis $\boldsymbol{T}_{\boldsymbol{A}}\in\mathbb{Z}_q^{m\times m}$ for lattice $\Lambda_q^{\perp}(\boldsymbol{A})$. The statistical distance between the distribution of the output A and uniform distribution over $\mathbb{Z}_q^{n\times m}$ is negligible. The norm of the Gram-Schmidt ordered orthogonalization of the output basis $\boldsymbol{T}_{\boldsymbol{A}}$ satisfies $\|\widetilde{\boldsymbol{T}_{\boldsymbol{A}}}\|\leqslant 3.8\sqrt{n\log q}$.

Lemma 2[2] There exists a PPT algorithm $\mathrm{SamplePre}(\boldsymbol{A},\boldsymbol{T}_{\boldsymbol{A}},\boldsymbol{u},\sigma)$ that can output a vector $\boldsymbol{x}\in\Lambda_q^{\boldsymbol{u}}(\boldsymbol{A})$ which is sampled from a distribution statistically close to $D_{\Lambda_q^{\boldsymbol{u}}(\boldsymbol{A}),\sigma}$. The inputs of SamplePre should satisfy that $\boldsymbol{A}\in\mathbb{Z}_q^{n\times m}, m>n, q>2$ and $\sigma\geqslant\|\widetilde{\boldsymbol{T}_{\boldsymbol{A}}}\|\cdot\omega\left(\sqrt{\log m}\right)$. $\boldsymbol{T}_{\boldsymbol{A}}$ is a short basis of $\Lambda_q^{\perp}(\boldsymbol{A})$.

Lemma 3[11] For positive integers q,n,m,k and a real number σ, where $k=\log q, m>k$, n is a power of two and q is a prime satisfying $q\equiv 1 \bmod 2n$. There exists a PPT algorithm $\mathrm{TrapGen}(n,m,q,\sigma)$ that outputs a uniformly random vector $\boldsymbol{a}\in R_q^m$ and a trapdoor $\boldsymbol{T}_{\boldsymbol{a}}\in R^{(m-k)\times k}$.

Lemma 4[18] There exists a PPT algorithm $\mathrm{SamplePre}(\boldsymbol{a},\boldsymbol{T}_{\boldsymbol{a}},u,\sigma)$ that can output a vector $\boldsymbol{x}\in R_q^m$ which satisfies $\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x}=u$ and follows a discrete gaussian distribution of parameter σ. $\boldsymbol{T}_{\boldsymbol{A}}$ is the trapdoor of $\boldsymbol{a}$.

2 The IBE Scheme

The IBE scheme uses the TrapGen and SamplePre algorithms to generate the master public key, master secret key and users' secret keys, which is similar with other LWE-based IBE schemes. The main difference between our scheme and others is in the encryption. Compared with other LWE-based schemes, the first part of ciphertexts of our scheme can also encrypt messages. While other schemes only encrypt the message in the second part of ciphertexts. This makes our scheme encrypt more bits than other schemes. To encrypt messages, our scheme uses LWE instances to mask the plaintext. As the LWE instances

$(\boldsymbol{A}, \boldsymbol{As}+\boldsymbol{e})$ are indistinguishable from uniformly random, $\boldsymbol{As}+\boldsymbol{e}+\boldsymbol{\Delta m}$ and $\boldsymbol{As}+\boldsymbol{e}+\boldsymbol{A\Delta m}$ will also be indistinguishable from uniformly random vector, where $\boldsymbol{\Delta}$ is $q/2$.

2.1 Construction Based on LWE

Our IBE scheme is described as follows:

Setup(1^{λ}): The PKG chooses an odd prime number $q>2$, positive integers n, m, l, k, a real number σ and a hash functions $H: \{0,1\}^{k} \to \mathbb{Z}_q^{n\times n} \times \mathbb{Z}_q^{n\times l}$, where k is the length of an identity and l is the length of the second part of message. The first part of the output of H is an invertible matrix. Then it runs TrapGen(n, m, q) to generate a uniformly random matrix $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$and a short basis $\boldsymbol{T_A} \in \mathbb{Z}_q^{m\times m}$ of the lattice $\Lambda_q^{\perp}(\boldsymbol{A})$. The PKG sets the master public key mp$k = \boldsymbol{A}$ and master secret key $\text{msk} = \boldsymbol{T_A}$. Finally, the PKG publishes the parameters $\text{param} = (n, m, l, q, \sigma, \boldsymbol{A}, H)$.

Extract$(\text{param}, \text{id}, \text{msk})$: For a users' identity id, the PKG first calculates $(\mathbf{U}_{1\text{id}}, \mathbf{U}_{2\text{id}}) = H(\text{id}) \in \mathbb{Z}_q^{n\times n} \times \mathbb{Z}_q^{n\times l}$. Then it runs SamplePre$(\boldsymbol{A}, \boldsymbol{T_A}, \boldsymbol{u}_{1i}, \sigma)$ for $i = 1, 2, \cdots, n$ to generate a matrix $\boldsymbol{S}_{1\text{id}}$ satisfying $\boldsymbol{A}S_{1\text{id}} = \boldsymbol{U}_{1\text{id}}$, where $\boldsymbol{u}_{1i}$ is the i-th column of $\boldsymbol{U}_{1\text{id}}$. Next, it computes $\boldsymbol{U}_{\text{id}} = \boldsymbol{U}_{1\text{id}}\boldsymbol{U}_{2\text{id}}$. Then PKG runs SamplePre$(\boldsymbol{A}, \boldsymbol{T_A}, \boldsymbol{U}_{\text{id}}, \sigma)$ to sample a matrix $\boldsymbol{S}_{2\text{id}}$ satisfying $\boldsymbol{U}_{1\text{id}}^{-1}\boldsymbol{A}\boldsymbol{S}_{2\text{id}} = \boldsymbol{U}_{2\text{id}}$. Finally, PKG sends $\text{sk}_{\text{id}} = (\boldsymbol{S}_{1\text{id}}, \boldsymbol{S}_{2\text{id}})$ to the user securely.

Enc$(\text{param}, \text{id}, \boldsymbol{m})$: To encrypt a message $\boldsymbol{m} \in \{0,1\}^{n+l}$, the sender first encodes the message as $\boldsymbol{m}' = \lfloor q/2 \cdot \boldsymbol{m} \rceil \in \mathbb{Z}_q^{n+l}$ and partitions it into two parts $\boldsymbol{m}'_1 \in \mathbb{Z}_q^{n}, \boldsymbol{m}'_2 \in \mathbb{Z}_q^{l}$ satisfying $\boldsymbol{m}' = [\boldsymbol{m}'^{\mathrm{T}}_1 | \boldsymbol{m}'^{\mathrm{T}}_2]^{\mathrm{T}}$. The public key of receiver can be calculated as $\text{pk}_{\text{id}} = (\boldsymbol{A}_{\text{id}}, \boldsymbol{U}_{2\text{id}}) = (U_{1\text{id}}^{-1}\boldsymbol{A}, \boldsymbol{U}_{2\text{id}})$. Then he samples are $\boldsymbol{e}_1 \leftarrow D_{\mathbb{Z}_q^n, \sigma}, \boldsymbol{e}_2 \leftarrow D_{\mathbb{Z}_q^m, \sigma}, \boldsymbol{e}_3 \leftarrow D_{\mathbb{Z}_q^l, \sigma}$ and we can calculate the ciphertext as $\mathbf{c} = (\mathbf{c}_0, \mathbf{c}_1) = (\boldsymbol{A}_{\text{id}}^{\mathrm{T}}\boldsymbol{m}'_1 + \boldsymbol{A}_{\text{id}}^{\mathrm{T}}\boldsymbol{e}_1 + \boldsymbol{e}_2, \boldsymbol{U}_{2\text{id}}^{\mathrm{T}}\boldsymbol{e}_1 + \boldsymbol{e}_3 + \boldsymbol{m}'_2) \in \mathbb{Z}_q^m \times \mathbb{Z}_q^l$.

Dec$(\text{param}, \text{sk}_{\text{id}}, \boldsymbol{c})$: After receiving a ciphertext $\boldsymbol{c}$, the receiver calculates $\boldsymbol{m}_1 = \lfloor 2/q \cdot \boldsymbol{S}_{1\text{id}}^{\mathrm{T}}\boldsymbol{c}_0 \rceil \in \{0,1\}^n$, $\boldsymbol{m}_2 = \lfloor 2/q \cdot (\boldsymbol{c}_1 - \boldsymbol{S}_{2\text{id}}^{\mathrm{T}}(\boldsymbol{c}_0 - \boldsymbol{A}_{\text{id}}^{\mathrm{T}}\lfloor q/2 \cdot \boldsymbol{m}_1 \rceil)) \rceil$. The message is $\boldsymbol{m} = [\boldsymbol{m}_1^{\mathrm{T}} | \boldsymbol{m}_2^{\mathrm{T}}]^{\mathrm{T}}$.

2.2 Correctness

Theorem 1 The decrypt algorithm can recover the message from ciphertext correctly if the following inequations are hold.

$$|\boldsymbol{e}_{1i} + \langle \boldsymbol{s}_{1\text{id}i}, \boldsymbol{e}_2 \rangle| < \frac{q}{4}, |\boldsymbol{e}_{3i} - \langle \boldsymbol{s}_{2\text{id}i}, \boldsymbol{e}_2 \rangle| < \frac{q}{4}$$

where $\boldsymbol{e}_{1i}$ is the i-th component of $\boldsymbol{e}_1$, $\boldsymbol{e}_{3i}$ is the i-th component of $\boldsymbol{e}_3$, $\mathbf{s}_{1\text{id}i}$ is the i-th column of $\boldsymbol{S}_{1\text{id}}$ and $\boldsymbol{s}_{2\text{id}i}$ is the i-th column of $\boldsymbol{S}_{2\text{id}}$.

Proof We first consider the error term of the first part of the decryption result. According to the scheme, we have $\boldsymbol{S}_{1\text{id}}^{\mathrm{T}}\boldsymbol{c}_0 = \boldsymbol{S}_{1\text{id}}^{\mathrm{T}}\boldsymbol{A}_{id}^{\mathrm{T}}\boldsymbol{m}'_1 + \boldsymbol{S}_{1\text{id}}^{\mathrm{T}}\boldsymbol{A}_{\text{id}}^{\mathrm{T}}\boldsymbol{e}_1 + \boldsymbol{S}_{1\text{id}}^{\mathrm{T}}\boldsymbol{e}_2 = \boldsymbol{S}_{1\text{id}}^{\mathrm{T}}\boldsymbol{A}_{\text{id}}^{\mathrm{T}}(\boldsymbol{m}'_1 + \boldsymbol{e}_1) + \boldsymbol{S}_{1\text{id}}^{\mathrm{T}}\boldsymbol{e}_2 = \boldsymbol{S}_{1\text{id}}^{\mathrm{T}}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{U}_{1\text{id}}^{-1\,\mathrm{T}}(\boldsymbol{m}'_1 + \boldsymbol{e}_1) + \boldsymbol{S}_{1\text{id}}^{\mathrm{T}}\boldsymbol{e}_2 = \boldsymbol{U}_{1\text{id}}^{\mathrm{T}}\boldsymbol{U}_{1\text{id}}^{-1\,\mathrm{T}}(\boldsymbol{m}'_1 + \boldsymbol{e}_1) + \boldsymbol{S}_{1\text{id}}^{\mathrm{T}}\boldsymbol{e}_2 = \boldsymbol{m}'_1 + \boldsymbol{e}_1 + \boldsymbol{S}_{1\text{id}}^{\mathrm{T}}\boldsymbol{e}_2 = \lfloor q/2 \cdot \boldsymbol{m}_1 \rceil + \boldsymbol{e}_1 + \boldsymbol{m}_1 \rceil + \boldsymbol{e}_1 + \boldsymbol{S}_{1\text{id}}^{\mathrm{T}}\boldsymbol{e}_2$. The i-th component of the first part of the decryption result and the message will be equal to each other if $|\boldsymbol{e}_{1i} + \langle \boldsymbol{s}_{1\text{id}i}, \boldsymbol{e}_2 \rangle| < q/4$. If the first part of the decryption result is correct, then for the second part of the decryption result, we have $\boldsymbol{c}_1 - \boldsymbol{S}_{2\text{id}}^{\mathrm{T}}(\boldsymbol{c}_0 - \boldsymbol{A}_{\text{id}}^{\mathrm{T}}\lfloor q/2 \cdot \boldsymbol{m}_1 \rceil) = \boldsymbol{c}_1 - \boldsymbol{S}_{2\text{id}}^{\mathrm{T}}(\boldsymbol{A}_{\text{id}}^{\mathrm{T}}\boldsymbol{e}_1 + \boldsymbol{e}_2) = \boldsymbol{U}_{2\text{id}}^{\mathrm{T}}\boldsymbol{e}_1 + \boldsymbol{e}_3 + \boldsymbol{m}'_2 - (\boldsymbol{U}_{2\text{id}}^{\mathrm{T}}\boldsymbol{e}_1 + \boldsymbol{S}_{2\text{id}}^{\mathrm{T}}\boldsymbol{e}_2) = \lfloor q/2 \cdot \boldsymbol{m}_2 \rceil + \boldsymbol{e}_3 - \boldsymbol{S}_{2\text{id}}^{\mathrm{T}}\boldsymbol{e}_2$. So the i-th component of the second part of the decryption result is correct if $|\boldsymbol{e}_{3i} - \langle \boldsymbol{s}_{2\text{id}i}, \boldsymbol{e}_2 \rangle| < \frac{q}{4}$.

2.3 Security

We first give the definition of IND-ID-CPA. Then we prove that our scheme can achieve IND-ID-CPA security.

Definition 5 For a IBE scheme P(Setup, Extract, Enc, Dec), a challenger C and an adversary A, the IND-ID-CPA game is described as follows.

Setup C runs the Setup algorithm to generate the public parameters param, master public key mpk and master secret key msk, then sends param and mpk to A.

Phase 1 A can make secret key extraction queries for any identity id. The challenger C runs Extract$(\text{param}, \text{id}, \text{msk})$ to generate the secret key sk and sends it to A.

Challenge A selects an identity id^* which has not been queried in phase 1 and two different messages m_0, m_1 and sends them to C. C runs $c \leftarrow \text{Enc}(\text{param}, \text{id}^*, m_b)$ and responses c to A, where b is a uniformly random bit.

Phase 2 A can repeat Phase 1. The only restriction is that he can't query the secret key of id^*.

Guess A outputs a bit b'.

If there does not exist any polynomial adversary A that achieves non-negligible $\mathrm{Adv}(A)=|\Pr[b'=b]-\Pr[b'\neq b]|$, we say that P is IND-ID-CPA secure. If the target identity is submitted before the setup, the game is named indistinguishability of ciphertexts against selectively chosen identity and chosen plaintext attack (IND-sID-CPA).

Theorem 2 If there exists a polynomial adversary A that can break the IND-ID-CPA secure of the scheme described in section 3.1 in the random oracle model, then there exists a polynomial algorithm that can solve the normal form decision-LWE problem.

Proof We show that how to use the adversary A to construct an algorithm C that can answer the normal form LWE oracle O. We set C to be the challenger of the IND-ID-CPA game. First, C queries the oracle O and gets $m+l$ instances $(\boldsymbol{a}_i,b_i)\in\mathbb{Z}_q^n\times\mathbb{Z}_q$. Then C uses these instances to combine two matrices $\boldsymbol{A_1}\in\mathbb{Z}_q^{m\times n},\boldsymbol{A_2}\in\mathbb{Z}_q^{l\times n}$ and two vectors $\boldsymbol{b_1}\in\mathbb{Z}_q^m,\boldsymbol{b_2}\in\mathbb{Z}_q^l$, where every row of matrices is $\boldsymbol{a}_i$ and every component of vectors is b_i. The rank of the matrix $\boldsymbol{A_1}$ must be n. If not, C can repeat the query until he gets a rank n matrix. After this, C performs the game with A as follows.

Setup　C generates the public params of the IBE scheme as follows.

1) Let Q_H be the number of hash queries made by A. C selects a random number $1\leqslant I\leqslant Q_H$.

2) C selects a uniformly random invertible matrix $\boldsymbol{U}_1^*\in\mathbb{Z}_q^{n\times n}$, then it calculates $\boldsymbol{A}^*=\boldsymbol{U}_1^*\boldsymbol{A}_1^{\mathrm{T}}$ and returns the $\mathrm{param}=(n,m,l,q,\sigma,\boldsymbol{A}^*,H)$ to A, where n,q,σ are the parameters of the normal LWE oracle.

Random-oracle hash queries　For the Q-th query submitted by A, C answers the query as follows. (We assume that each identity will only be queried once.)

1) If $Q=I$, C returns $(\boldsymbol{U}_1^*,\boldsymbol{A}_2^{\mathrm{T}})$ and saves the tuple $(Q,\mathrm{id},\boldsymbol{U}_1^*,\boldsymbol{A}_2^{\mathrm{T}},\perp,\perp)$.

2) If $Q\neq I$, C samples $n+l$ vectors $\mathbf{e}_i\leftarrow D_{\mathbb{Z}_q^m,\sigma}$ and uses these vector to combine two matrices $\boldsymbol{S}_{1\mathrm{id}}\in\mathbb{Z}_q^{m\times n}$, $\boldsymbol{S}_{2\mathrm{id}}\in\mathbb{Z}_q^{m\times l}$, which satisfies that $\boldsymbol{A}^*\boldsymbol{S}_{1\mathrm{id}}$ is an invertible matrix. Then C calculates $\boldsymbol{U}_{1\mathrm{id}}=\boldsymbol{A}^*\boldsymbol{S}_{1\mathrm{id}}$ and $\boldsymbol{U}_{2\mathrm{id}}=\boldsymbol{U}_{1\mathrm{id}}^{-1}\boldsymbol{A}^*\boldsymbol{S}_{2\mathrm{id}}$. C saves the tuple $(Q,\mathrm{id},\boldsymbol{U}_{1\mathrm{id}},\boldsymbol{U}_{2\mathrm{id}},\boldsymbol{S}_{1\mathrm{id}},\boldsymbol{S}_{2\mathrm{id}})$ and returns $(\boldsymbol{U}_{1\mathrm{id}},\boldsymbol{U}_{2\mathrm{id}})$ to A.

Secret key queries　When A queries the secret key for an id, C searches the saved tuple $(Q,\mathrm{id},\boldsymbol{U}_{1\mathrm{id}},\boldsymbol{U}_{2\mathrm{id}},\boldsymbol{S}_{1\mathrm{id}},\boldsymbol{S}_{2\mathrm{id}})$ from the history of the hash oracle query. (We assume that every id has been queried in hash query before secret key query.) If $\boldsymbol{S}_{1\mathrm{id}}=\perp$ and $\boldsymbol{S}_{2\mathrm{id}}=\perp$, C aborts and outputs a random bit. Else, C returns $(\boldsymbol{S}_{1\mathrm{id}},\boldsymbol{S}_{2\mathrm{id}})$ to A.

Challenge　A submits the target id^* and two random messages $\mathbf{m}_0,\mathbf{m}_1$ to C. If id^* is not the identity that is queried in the I-th hash query, C aborts and outputs a random bit. Otherwise, C chooses a random bit b, and computes the challenge ciphertext $\boldsymbol{c}=(\boldsymbol{c}_0,\boldsymbol{c}_1)=(\boldsymbol{A_1}\boldsymbol{m}'_{b_1}+\boldsymbol{b_1},\boldsymbol{b_2}+\boldsymbol{m}'_{b_2})$, where $\boldsymbol{m}'_b=\lfloor q/2\cdot\boldsymbol{m}_b\rceil=[\boldsymbol{m}'^{\mathrm{T}}_{b_1}|\boldsymbol{m}'^{\mathrm{T}}_{b_2}]^{\mathrm{T}}$.

If the instances returned by O are uniformly random vectors, the challenge ciphertext is uniformly random and completely hides the information of b. If the instances returned by O are normal LWE instances, then the challenge ciphertext will be a valid encryption of $\mathbf{m}_b$ for identity id^*.

A can make more secret key queries after the challenge phase with the restriction that it can't make the secret key query for identity id^*. After that, A outputs a bit b' to C. If $b'=b$, C outputs 1, else C outputs 0.

If C dose not abort, the interaction between oracle O, algorithm C and adversary A will be as shown in Figure 1.

By a standard argument, C will not abort during the game with the probability $1/Q_H$. If C does not abort, it means that the target identity submitted by A is the identity queried by the I-th hash query. In this case, if the challenge ciphertext is uniformly random, the probability $b'=b$ is $1/2$. (Which means that the instances returned by O are uniformly random.) If the challenge ciphertext is a valid encryption, the probability $b'=b$ is $1/2+\mathrm{Adv}(A)$. (Which means that the instances returned by O are LWE instances.) So the advantage that C can distinguish LWE instances from uniformly random vectors is $\mathrm{Adv}(A)/Q_H$. Thus, if A has an non-negligible advantage to break the IND-ID-CPA security, then C has a non-negligible advantage

to solve decision normal LWE problem.

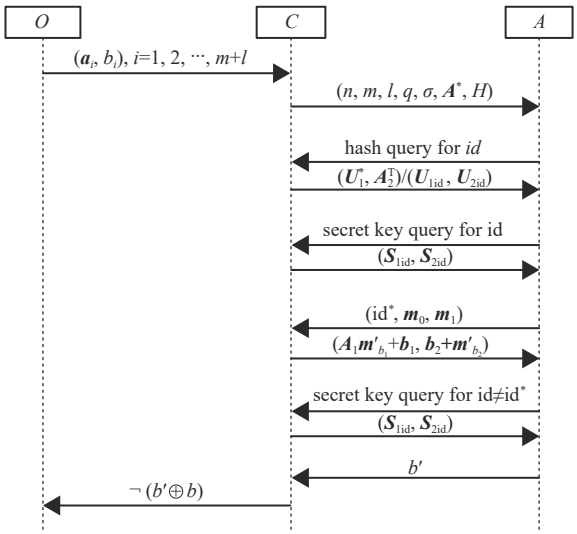

Fig. 1 The interaction between O,C and A.

2.4 Construction Based on RLWE

Our ring version IBE scheme is described as follows:

Setup(1^λ): The PKG chooses positive integers n, m, q, k, l, two real number σ, θ and a hash functions $H: \{0,1\}^l \to R_q \times R_q$, where l is the length of an identity. The first part of the output of H is an invertible element of R_q. Then PKG runs TrapGen(n, m, q, θ) to generate a random vector $\boldsymbol{a} \in R_q^m$ and a short basis $\boldsymbol{T}_{\boldsymbol{a}} \in R_q^{(m-k)\times k}$ of the lattice $\Lambda_q^{\perp}(\boldsymbol{a})$. The PKG sets the master public key $\mathrm{mpk} = \boldsymbol{a}$ and master secret key $\mathrm{msk} = \boldsymbol{T}_{\boldsymbol{a}}$. Finally, the PKG publishes the parameters $param = (n, m, l, q, k, \sigma, \theta, \boldsymbol{a}, H)$.

Extract(param, id, msk): For a users' identity id, the PKG first calculates $(u_{1\mathrm{id}}, u_{2\mathrm{id}}) = H(\mathrm{id}) \in R_q \times R_q$. Then it runs SamplePre$(\boldsymbol{a}, \boldsymbol{T}_{\boldsymbol{a}}, u_{1\mathrm{id}}, \sigma)$ to generate a vector $\mathbf{x}_{1\mathrm{id}}$ satisfying $\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x}_{1\mathrm{id}} = u_{1\mathrm{id}}$. Next, it computes $u_{\mathrm{id}} = u_{1\mathrm{id}}u_{2\mathrm{id}}$. Then PKG runs SamplePre$(\boldsymbol{a}, \boldsymbol{T}_{\boldsymbol{a}}, u_{\mathrm{id}}, \sigma)$ to sample a vector $\boldsymbol{x}_{2\mathrm{id}}$ satisfying $u_{1\mathrm{id}}^{-1}\boldsymbol{a}^{\mathrm{T}}\boldsymbol{x}_{2\mathrm{id}} = u_{2\mathrm{id}}$. Finally, PKG sends $\mathrm{sk}_{\mathrm{id}} = (\boldsymbol{x}_{1\mathrm{id}}, \boldsymbol{x}_{2\mathrm{id}})$ to the user securely.

Enc(param, id, $\boldsymbol{m}$): To encrypt a message $\boldsymbol{m} \in R_2 \times R_2$, the sender first encodes the message as $\boldsymbol{m}' = \lfloor q/2 \cdot \boldsymbol{m} \rceil \in R_q \times R_q$ and partitions it into two parts $m'_1 \in R_q, m'_2 \in R_q$. The public key of receiver can be calculated as $\mathrm{pk}_{\mathrm{id}} = (\boldsymbol{a}_{\mathrm{id}}, u_{2\mathrm{id}}) = (u_{1\mathrm{id}}^{-1}\boldsymbol{a}, u_{2\mathrm{id}})$. Then it samples $e_1, e_3 \leftarrow D_{R_q,\sigma}, \boldsymbol{e}_2 \leftarrow D_{R_q^m,\sigma}$ and calculates the ciphertext $\boldsymbol{c} = (\mathbf{c}_0, c_1) = (m'_1\boldsymbol{a}_{\mathrm{id}} + e_1\boldsymbol{a}_{\mathrm{id}} + \boldsymbol{e}_2, u_{2\mathrm{id}}e_1 + e_3 + m'_2) \in R_q^m \times R_q$.

Dec(param, $\mathrm{sk}_{\mathrm{id}}, \boldsymbol{c}$): After receiving a ciphertext $\boldsymbol{c}$, the receiver calculates $m_1 = \lfloor 2/q \cdot \boldsymbol{c}_0^{\mathrm{T}}\boldsymbol{x}_{1\mathrm{id}} \rceil \in R_q$, $m_2 = \lfloor 2/q \cdot \left(c_1 - (\mathbf{c}_0 - \lfloor q/2 \cdot \boldsymbol{m}_1 \rceil \boldsymbol{a}_{\mathrm{id}})^{\mathrm{T}}\boldsymbol{x}_{2\mathrm{id}}\right) \rceil$. The message is $\boldsymbol{m} = (m_1, m_2)$.

The decrypt algorithm can recover the message from ciphertext correctly if the following inequations hold:

$$\| e_1 + \langle \boldsymbol{x}_{1\mathrm{id}}, \boldsymbol{e}_2 \rangle \|_\infty < \frac{q}{4}, \| e_3 - \langle \boldsymbol{x}_{2\mathrm{id}}, \boldsymbol{e}_2 \rangle \|_\infty < \frac{q}{4}$$

The proof is similar as that in section 3.2, so we omit it.

Assume that the decision-RLWE problem is hard, the ring version IBE is IND-ID-CPA secure. The proof of the security of the ring version IBE is similar as that in section 2.3, so we omit it.

3 Comparison

We compare the correctness conditions for our scheme with the conditions for GPV[2] and ABB[5]. The security model and the error term in the decryption results of our scheme, GPV[2] and ABB[5] are listed in Table 1. (The schemes used for comparison are multi bits encryption version.)

Table 1 Correctness condition of three schemes

Schemes	Security model	Hardness assumption	Correctness Condition
GPV[2]	Random oracle	LWE	$\|\boldsymbol{e}_2 - \boldsymbol{S}_{\mathrm{id}}^{\mathrm{T}}\boldsymbol{e}_1\|_\infty < q/4$ $\boldsymbol{e}_2 \leftarrow D_{\mathbb{Z}_q^l,\sigma}, \boldsymbol{e}_1 \leftarrow D_{\mathbb{Z}_q^m,\sigma}$ $\boldsymbol{S}_{\mathrm{id}} \leftarrow$ SamplePre
ABB[5]	Standard model	LWE	$\|\boldsymbol{e}_2 - \boldsymbol{S}_{\mathrm{id}}^{\mathrm{T}}[\boldsymbol{e}_1^{\mathrm{T}} \| \boldsymbol{e}_1^{\mathrm{T}}\boldsymbol{R}]^{\mathrm{T}}\|_\infty < q/4$ $\boldsymbol{e}_2 \leftarrow D_{\mathbb{Z}_q^l,\sigma}, \boldsymbol{e}_1 \leftarrow D_{\mathbb{Z}_q^m,\sigma}$ $\boldsymbol{R} \leftarrow \{-1, 1\}^{m\times m}$ $\boldsymbol{S}_{\mathrm{id}} \leftarrow$ SampleLeft
Ours	Random oracle	LWE	$\|\boldsymbol{e}_3 - \boldsymbol{S}_{2\mathrm{id}}^{\mathrm{T}}\boldsymbol{e}_2\|_\infty < q/4$ $\|\boldsymbol{e}_1 + \boldsymbol{S}_{1\mathrm{id}}^{\mathrm{T}}\boldsymbol{e}_2\|_\infty < q/4$ $\boldsymbol{e}_1 \leftarrow D_{\mathbb{Z}_q^m,\sigma}, \boldsymbol{e}_2 \leftarrow D_{\mathbb{Z}_q^m,\sigma}, e_3 \leftarrow D_{\mathbb{Z}_q^l,\sigma}$ $\boldsymbol{S}_{1\mathrm{id}}, \boldsymbol{S}_{2\mathrm{id}} \leftarrow$ SamplePre

The correctness condition of our scheme is similar with GPV[2]. The difference between GPV[2] and our scheme is that our scheme needs to guarantee the second inequation. As we can set the distribution parameters of secret random vector $\boldsymbol{e}_1$ to be the same as the distribution parameters of $\boldsymbol{e}_2$, the second

inequation will always hold as long as the first inequation holds. Thus, the parameter setting of GPV[2] can be used for our scheme without losing correctness and security. ABB[5] has more decryption noise because of the standard model. The length of the error term used in ABB's[5] encryption is $2m$. The distribution of the outputs of SampleLeft is the same as the outputs of SamplePre except the length of outputs. Therefore, the parameter setting that guarantee the correctness of ABB[5] can also guarantee the correctness of our scheme.

Compared with other lattice-based IBE schemes, our scheme can encrypt more bits in a ciphertext, which means that our scheme can achieve smaller ciphertext size than other lattice-based IBE schemes when the length of the plaintext is fixed. To encrypt more bits, the size of secret key of our IBE scheme is bigger than that of other schemes. However, the ratio of the size of secret key to the size of plaintext is not greater than that of other schemes. The comparison of the ciphertext size, secret key size and plaintext size is shown by Table 2. To encrypt kl bits, one can just divide it into k pieces and encrypt each piece to get k ciphertexts. However, the ratio of the ciphertext size to the plaintext size of each scheme will not be changed, as every ciphertext only corresponds to a piece of plaintext with l bits.

When the parameters n,m,l,q of schemes are chosen, our scheme can encrypt $l+n$ bits in a ciphertext whose size is $(m+l)\log q$, while other LWE-based schemes can only encrypt l bits in a ciphertext with the same ciphertext size. If $l=n$, the length of message that our scheme can encrypt is twice of other schemes, and the size of the ciphertext is not bigger than that of other schemes.

Table 2　Comparison of ciphertext size, secret key size and plaintext size

Schemes	Size of k ciphertexts	Secret key size	Size of k plaintexts	Security model	Security	Assumption
GPV[2]	$k(m+l)\log q$	$ml\log q$	kl	Random oracle	IND-ID-CPA	LWE
ABB[5]	$k(2m+l)\log q$	$2ml\log q$	kl	Standard model	IND-ID-CPA	LWE
ZCZ[19]	$k(m+m'+l)\log q$	$(m+m')l\log q$	kl	Standard model	IND-ID-CPA	LWE
Y[20]	$k(2m+l)\log q$	$2ml\log q$	kl	Standard model	IND-ID-CPA	LWE
BFR[11]	$k(m+1)n\log q$	$mn\log q$	kn	Standard model	IND-sID-CPA	RLWE
ZLGZW[12]	$k(2m+1)n\log q$	$2mn\log q$	kn	Standard model	IND-ID-CPA	RLWE
Ours	$k(m+l)\log q$	$m(l+n)\log q$	$k(l+n)$	Random oracle	IND-ID-CPA	LWE
Ours (ring)	$k(m+1)n\log q$	$2mn\log q$	$k(2n)$	Random oracle	IND-ID-CPA	RLWE

The disadvantage of our scheme is the computation cost. Our scheme needs to perform one more discrete gaussian sampling and vector addition in encryption than other schemes. In decryption, our scheme needs to perform two more matrix multiplications, and one more vector subtraction. However, as our scheme is able to transmit n more bits, we think the performance penalty of adding these computations is acceptable.

4　Conclusion

In this paper, we propose an IBE scheme and its ring version based on LWE/RLWE problem and prove that our schemes is IND-ID-CPA secure under the random oracle model. Compared with previous lattice-based IBE schemes, the computation cost of encryption and decryption of the proposed schemes is more than others, but our schemes can encrypt more message without increasing the size of ciphertext.

References

[1] SHAMIR A. Identity-Based cryptosystems and signature schemes[C]//Workshop on the Theory and Application of Cryptographic Techniques. Berlin, Heidelberg: Springer, 1984: 47-53.

[2] GENTRY C, PEIKERT C, VAIKUNTANATHAN V. Trapdoors for hard lattices and new cryptographic constructions[C]//Proceedings of the 40th Annual ACM Symposium on Theory of Computing. [s.l.]: ACM, 2008: 197-206.

[3] REGEV O. On lattices, learning with errors, random linear codes, and cryptography[C]//Proceedings of the 37th

Annual ACM Symposium on Theory of Computing. [S.l.]: ACM, 2005: 84-93.
[4] CASH D, HOFHEINZ D, KILTZ E, et al. Bonsai trees, or how to delegate a lattice basis[C]//Annual International Conference on the Theory and Applications of Cryptographic Techniques. Berlin, Heidelberg: Springer, 2010: 523-552.
[5] AGRAWAL S, BONEH D, BOYEN X. Efficient lattice (H) IBE in the standard model[C]//Annual International Conference on the Theory and Applications of Cryptographic Techniques. Berlin, Heidelberg: Springer, 2010: 553-572.
[6] AGRAWAL S, BONEH D, BOYEN X. Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical IBE[C]//Annual Cryptology Conference. Berlin, Heidelberg: Springer, 2010: 98-115.
[7] AJTAI M. Generating hard instances of the short basis problem[C]// International Colloquium on Automata, Languages, and Programming. Berlin, Heidelberg: Springer, 1999: 1-9.
[8] ALWEN J, PEIKERT C. Generating shorter bases for hard random lattices[C]//The 26th International Symposium on Theoretical Aspects of Computer Science STACS 2009. [S.l.]: IBFI Schloss Dagstuhl, 2009: 75-86.
[9] MICCIANCIO D, PEIKERT C. Trapdoors for lattices: Simpler, tighter, faster, smaller[C]//Annual International Conference on the Theory and Applications of Cryptographic Techniques. Berlin, Heidelberg: Springer, 2012: 700-718.
[10] KATSUMATA S, YAMADA S. Partitioning via non-linear polynomial functions: More compact IBEs from ideal lattices and bilinear maps[C]//International Conference on the Theory and Application of Cryptology and Information Security. Berlin, Heidelberg: Springer, 2016: 682-712.
[11] BERT P, FOUQUE P A, ROUX-LANGLOIS A, et al. Practical implementation of ring-SIS/LWE based signature and IBE[C]//International Conference on Post-Quantum Cryptography. Cham: Springer, 2018: 271-291.
[12] ZHANG Y, LIU Y, GUO Y, et al. Adaptively secure efficient (H) IBE over ideal lattice with short parameters[J]. Entropy, 2020, 22(11): 1247.
[13] AJTAI M. Generating hard instances of lattice problems[C]//Proceedings of the 28th Annual ACM Symposium on Theory of Computing. [S.l.]: ACM, 1996: 99-108.
[14] MICCIANCIO D. Generalized compact knapsacks, cyclic lattices, and efficient one-way functions from worst-case complexity assumptions[C]//The 43rd Annual IEEE Symposium on Foundations of Computer Science Proceedings.[S.l.]: IEEE, 2002: 356-365.
[15] APPLEBAUM B, CASH D, PEIKERT C, et al. Fast cryptographic primitives and circular-secure encryption based on hard learning problems[C]//Annual International Cryptology Conference. Berlin, Heidelberg: Springer, 2009: 595-618.
[16] LYUBASHEVSKY V, PEIKERT C, REGEV O. On ideal lattices and learning with errors over rings[C]//Annual International Conference on the Theory and Applications of Cryptographic Techniques. Berlin, Heidelberg: Springer, 2010: 1-23.
[17] LUO F, WANG K, LIN C. Leveled hierarchical identity-based fully homomorphic encryption from learning with rounding[C]//International Conference on Information Security Practice and Experience. Cham: Springer, 2018: 101-115.
[18] GENISE N, MICCIANCIO D. Faster gaussian sampling for trapdoor lattices with arbitrary modulus[C]//Annual International Conference on the Theory and Applications of Cryptographic Techniques. Cham: Springer, 2018: 174-203.
[19] ZHANG J, CHEN Y, ZHANG Z. Programmable hash functions from lattices: Short signatures and IBEs with small key sizes[C]//Annual International Cryptology Conference. Berlin, Heidelberg: Springer, 2016: 303-332.
[20] YAMADA S. Adaptively secure identity-based encryption from lattices with asymptotically shorter public parameters[C]//Annual International Conference on the Theory and Applications of Cryptographic Techniques. Berlin, Heidelberg: Springer, 2016: 32-62.

编 辑 税 红