Dual System in Lattice: Fully Secure ABE from LWE Assumption

Geng Wang, Ming Wan, Zhen Liu and Dawu Gu

School of Electronic Information and Electrical Engineering
Shanghai Jiao Tong University, 100072, P.R.China
{wanggxx,wanming,liuzhen,dwgu}@sjtu.edu.cn

Abstract. Dual system encryption is an important method used in pairing-based cryptography for constructing fully secure IBE, ABE and FE schemes. A long time open question is that, whether there is an analogue of dual system method in lattice, which can be used to prove the full security of lattice-based ABE or FE schemes. We solve this problem in this paper.

We do this by constructing a fully secure CP-ABE scheme supporting C-NF as its access policy from lattice with a "dual system". We introduce a new primitive called approximate inner product encryption (aIPE), which is the approximate version of the well known inner product encryption, and prove the security of our scheme under the selective security of aIPE and the LWE assumption. The security proof of the scheme is also similar to dual system in bilinear groups.

We point out that the functionality of aIPE is included in FE for arbitrary circuits, which can be constructed from LWE assumption along with circular security, hence the full security of our scheme can be totally based on the hardness of LWE.

Keywords: Attribute-based encryption, Dual system encryption, LWE, Lattice-based cryptography

1 Introduction

Attribute-based Encryption (ABE for short) was first brought by Sahai and Waters in 2005 [32]. In an ABE scheme, the decryption is correct if and only if the provided attribute set satisfies a certain access policy. By using different types of access policies, ABE can handle flexible access control matters, without using complex key distribution techniques. There are mainly two types of ABE, one is called key-policy ABE (KP-ABE) [21], other is called ciphertext-policy ABE (CP-ABE) [7]. In KP-ABE, the access policy is embedded in the decryption key, while the ciphertext is related to a set of attributes; in CP-ABE, the access policy is embedded in the ciphertext, and attributes are related to the decryption key, held by the users. In [10], ABE is considered as a special case of a more generalized primitive called functional encryption (FE), which given an encrypted data $Enc(x)$, calculate the output $f(x)$ for the encrypted data of a function in a certain function class $f \in \mathcal{F}$.

Most of the early ABE schemes [21, 7, 28, 16, 36, 31] are from a weak security model, which is called selective security. In a selective security model, the adversary must first given the challenge policy (for CP-ABE) or challenge attribute set (for KP-ABE) before it was allowed to get the public key and query for secret keys. It is easy to see that the selective security model greatly restricts the ability of the adversary, and cannot handle many types of real world attacks.

Many researchers focus on removing the restriction to get full security for ABE schemes. Many different approaches have been proposed, but the most successful one among them is the dual-system encryption method, given by Waters in 2009 [35]. Although the original method is for IBE and HIBE, which are only simplified versions of ABE, it was soon used to construct fully secure ABE schemes for various access policies, as in [24, 25, 6, 37, 22, 15, 23].

The schemes above are constructed in bilinear groups, which suffer from quantum attacks. Recently, many researchers have been working on constructing ABE schemes using lattice assumptions, such as learning with error (LWE) problem [3, 2, 11]. Lattice-based ABE schemes are not only quantum secure, but also more powerful than schemes in bilinear groups, as they support much richer classes of access policies, even for arbitrary circuits [19, 9, 13].

However, the existing lattice-based ABE schemes are only selectively secure, except for a recent work by Tsabary [33] that can only support a quite weak class of access policies. Since the original dual-system method is highly related to the properties of pairing in bilinear groups, it was not known whether there exists an analogue for dual-system in lattice, which could be used to prove the full security of lattice-based ABE schemes. This question has been raised in many earlier works, and has been considered as a long time open problem in lattice-based cryptography. In this paper, we solve this problem by extending dual system method into lattice, also giving a CP-ABE scheme supporting CNF policies and proving its full security in the standard model.

1.1 Dual System in Bilinear Groups

We first recall dual system encryption in bilinear groups. In addition to normal secret keys and normal ciphertexts which are used in the real scheme, semi-functional keys and ciphertexts are defined, which are only used in the security proof. Normal secret keys can be used to decrypt normal and semi-functional ciphertexts, while semi-functional keys can only decrypt normal ciphertexts. In the security proof, first the challenge ciphertext, then the queried secret keys are switched into semi-functional. After all ciphertexts and keys become semi-functional, it can perfectly hide the message.

Although we shall not go deep into details, it is helpful that we first simply introduce what a semi-functional ciphertext and a semi-functional secret key in bilinear groups look like. For a typical fully secure ABE scheme such as the scheme in [25], a normal secret key consists of several key elements $K, K_1, ..., K_l$, and a normal ciphertext consists of several ciphertext elements $C, C_1, ..., C_l$. A semi-functional ciphertext is generated by altering at least one of these ciphertext elements, such as C, into $C \cdot V$, V is from another group G_2 different from the group which C is in. Similarly, a semi-functional secret key is generated by altering at least one of the key elements, such as K into $K \cdot W$, W is from G_2. Let e be the pairing operation used in the decryption algorithm, we have that $e(CV, K) = e(C, KW) = e(C, K)$, but $e(CV, KW) = e(C, K)e(V, W)$, where the value of $e(C, K)$ is hidden by $e(V, W)$.

We can see that, for a semi-functional ciphertext, the additional element V multiplied on the normal ciphertext element C can be eliminated by pairing with a normal secret key. So we take a similar approach in lattice-based cryptography: that is, first introducing an additional element into the normal ciphertext to make it semi-functional, and find a way to eliminate it using the decryption algorithm with a certain secret key.

1.2 Our Technique

At the beginning, we point out that instead of using "normal secret keys" and "semi-functional secret keys" as in bilinear groups, we use the term "hyper-functional secret keys" and "normal secret keys". This is because that in bilinear groups, a normally constructed secret key can be used to wipe out the additional term V through pairing, however, in lattice-based schemes, we need special construction for the secret key in order to wipe out the additional term, and it cannot be normal.

In the security proof, our hyper-functional keys act like normal keys in bilinear groups, and our normal keys act like semi-functional keys in bilinear groups. We need an additional step in our security proof, that is, switching the normal keys into hyper-functional keys.

First attempt. We start from the dual-Regev scheme [17], which is widely used to construct lattice-based ABE schemes. Different from bilinear groups, the pairing operation becomes multiplication, and multiplication becomes addition. So we set the semi-functional ciphertext as $\mathbf{s}^T\mathbf{A}+\mathbf{e}^T+\mathbf{v}^T$ for some $\mathbf{v}$, and let the hyper-functional secret key $\mathbf{x}$ satisfies both $\mathbf{A}\mathbf{x} = \mathbf{u}$, and $\mathbf{v}^T\mathbf{x} = 0$. However, this simply cannot work, as given enough hyper-functional secret keys, the adversary can simply reconstruct the lattice $L^{\perp}(\mathbf{v})$, hence get the ability to check whether a key is normal or hyper-functional.

Second attempt. This time, we set $\mathbf{v}^T\mathbf{x} \approx 0$ in a hyper-functional key $\mathbf{x}$. By introducing a small error, the indistinguishability between normal keys and hyper-functional keys can be reduced to the hardness of LWE problem. But again, we find it hard to hide $\mathbf{v}$ in the ciphertext, since $(\mathbf{s}^T\mathbf{A}+\mathbf{e}^T)\mathbf{x}$ and $(\mathbf{s}^T\mathbf{A}+\mathbf{e}^T+\mathbf{v}^T)\mathbf{x}$ are from different distribution. After a few failed attempt from constructing dual-Regev like schemes, we noticed that we need a new primitive which functionality could satisfy our requirement.

The final solution: aIPE. Such primitive that finally comes to us is approximate inner-product encryption (aIPE). Instead of classical IPE schemes [24, 3, 5], where given a ciphertext encrypting $\mathbf{m}$ with a secret key generated from $\mathbf{x}$, calculate the exact inner prodcut $\langle \mathbf{x}, \mathbf{m} \rangle$, aIPE calculates the inner product with a small error: $aIPE.Dec(sk_{\mathbf{x}}, ct_{\mathbf{m}}) = \langle \mathbf{x}, \mathbf{m} \rangle + e$. Hence, we could set the secret key as $sk_{\mathbf{x}}$, and the semi-functional ciphertext as $ct_{\mathbf{s}^T\mathbf{A}+\mathbf{v}^T}$. We require that,

$ct_{\mathbf{m}}$ and $ct_{\mathbf{m}+\mathbf{v}}$ are indistinguishable if for each queried secret key $sk_{\mathbf{x}}$, there is $|\langle \mathbf{x}, \mathbf{v} \rangle| \leq \beta$, β is a small value. Then we can show the indistinguishability between normal and semi-functional ciphertexts.

The existence of aIPE. Now we only need to show the existence of aIPE. Although there are many constructions for (exact) inner product encryption from various assumptions including LWE, extending them into the approximate case turns out to be not an easy task. However, we can see that the functionality of aIPE is obviously weaker than functional encryption (FE) for arbitrary circuits, and the existence of FE for arbitrary circuits has been shown by Goldwasser et al [18] under the hardness of LWE. The security definition of FE is in a very weak model (called selective non-adaptive secure), hence we only have a very weak aIPE security. However, we will show that such security definition is enough for constructing our fully secure ABE scheme.

1.3 Related Works

There are currently a few researches working on lattice-based fully secure identity-based encryption (IBE)[1, 14, 12, 38], which can be considered as ABE which access policy is point function. These security proofs rely on various primitives, such as admissible hash or pseudorandom functions. It is not known how these techniques can be used for other access policies. In [19], the authors claimed that using a result from [8], the selective security of a KP-ABE scheme can be extended to full security assuming the subexponential hardness of LWE. Despite the non-standardness of the hardness assumption, it seems that this method cannot be extended into CP-ABE schemes. In [13, 20], the authors focused on semi-adaptive security of ABE schemes. Although stronger than selective security, it is still weaker than full security.

In [33], the author gave the first fully secure ABE scheme (other than IBE) from standard LWE assumption using a new primitive called conforming cPRF, which is a huge step forward. However, the access policy is only t-CNF for a constant t, which means that each clause exactly contains t literals. This is much weaker than our access policy, which is (unrestricted) CNF. The author claimed that the access policy is only related to the expressibility of the conforming cPRF, however, constructing conforming cPRF supporting various access policies seems to be extremely difficult. The scheme is also quite complex. Despite the complexity in the conforming cPRF itself, the function needs to be evaluated through key-homomorphic encryption [9], which makes the scheme almost impossible for implementation. Although our scheme is also impractical at current time, we note that the complexity of our scheme lies mostly in the construction of aIPE. So the efficiency of our scheme can be easily improved if we found a more simple construction for aIPE.

In [34], a fully secure decentralized ABE is constructed from inner product encryption based on LWE assumption [5]. The idea of using aIPE in our scheme is partly borrowed from their work. The main drawback of their work is that the key queries allowed by the adversary is highly restricted, which makes its security unreliable, while our scheme allows polynomial number of queries (although the number must be predetermined). We also point out that, using similar techniques, it seems that our scheme can also be made decentralized. However, we will not discuss that in this paper.

2 Preliminaries

Notations. $x \leftarrow \chi$ for a distribution χ means that x is sampled from χ or x follows the distribution χ. $x \leftarrow X$ for a set X means that x is uniformly randomly chosen from X. For any odd modulus q, $\mathbb{Z}_q$ and the operation mod q takes value from $[-\frac{q-1}{2}, \frac{q-1}{2}]$. We say that ϵ is negligible in λ, if $\epsilon < 1/\Omega(\lambda^c)$ for any constant $c > 0$. $\lfloor x \rceil$ means the nearest integer to x. For an integer k, $[k]$ means the set $\{1, ..., k\}$.

2.1 Conjunctive Normal Form

Definition 2.1. *Let* $\mathbf{L}$ *be a set of literals (a literal is either* α *or* $\neg\alpha$ *for some variable* α*), and* $T_1, ..., T_k \subseteq \mathbf{L}$ *be a set of* clauses*.*

A conjunctive normal form (CNF) is a boolean function $f = \bigwedge_{i=1}^{k}(\bigvee T_i)$*, which inputs a set of literals* $L \subseteq \mathbf{L}$ *(for each variable* α*,* α *and* $\neg\alpha$ *not both in* L*), and outputs the value* $f(L) = \bigwedge_{i=1}^{k}(\bigvee T_i(L))$*. Here* $\bigvee T_i(L) = 1$ *if and only if* $T_i \cap L \neq \emptyset$*.*

Let $l = |\mathbf{L}|$*, and we label the literals in* $\mathbf{L}$ *by 1 to* l*.*

Note that we do not consider the relationship between α and $\neg\alpha$, and simply let them be two different elements. Such representation does not lower the expressibility of CNF policy. In fact, our definition is stronger than boolean formulas: for an attribute (literal) set L, we allow that neither α nor $\neg\alpha$ is in L, which means that we "do not care" the value of α, as in [16].

2.2 Ciphertext-Policy Attribute-based Encryption

Definition 2.2. *A CP-ABE scheme for CNF formula* f *consists of four algorithms* $(\mathsf{Setup}, \mathsf{Enc}, \mathsf{KeyGen}, \mathsf{Dec})$*:*

- $\mathsf{Setup}(1^\lambda) \rightarrow (\mathsf{mpk}, \mathsf{msk})$*: The setup algorithm gets as input the security parameter* λ*, and outputs the public parameter* mpk*, and the master key* msk*.*
- $\mathsf{Enc}(\mathsf{mpk}, f, m) \rightarrow \mathsf{ct}_f$*: The encryption algorithm gets as input* mpk*, a CNF formula* f*, and a message* $m \in \mathcal{M}$*. It outputs a ciphertext* ct_f*. Note that the policy is known if we know the ciphertext.*
- $\mathsf{KeyGen}(\mathsf{msk}, L) \rightarrow \mathsf{sk}_L$*: The key generation algorithm gets as input* msk *and a set of literals* L*. It outputs a secret key* sk_L*.*
- $\mathsf{Dec}(\mathsf{sk}_L, \mathsf{ct}_f) \rightarrow m$*: The decryption algorithm gets as input a secret key and a ciphertext, and outputs either* $\perp$ *or a message* $m \in \mathcal{M}$*.*

The CP-ABE scheme is correct if and only if the decryption algorithm returns the correct message when $f(L) = 1$ *and returns* $\perp$ *when* $f(L) = 0$*, except for a negligible probability.*

Definition 2.3. *The full security of a CP-ABE scheme is defined through the following game:*

Setup. *The challenger runs* Setup *and gives the adversary* mpk.

Phase 1. *The adversary submits a set of literals L for a* KeyGen *query, and gets* sk_L *from the challenger. These queries can be repeated adaptively.*

Challenge. *The adversary submits two messages m_0 and m_1 of equal length, and a CNF formula f such that $f(L) = 0$ for all previously queried L. The challenger chooses a random bit $b \in \{0, 1\}$, and encrypts m_b under f. The encrypted ciphertext* ct_f *is returned to the adversary.*

Phase 2. *The adversary repeats **Phase 1** to get more secret keys. Each queried L must have $f(L) = 0$.*

Guess. *The adversary outputs a guess b' for b.*

The advantage of the adversary in the full-CP-ABE game is defined by $|Pr[b' = b] - 1/2|$. We say that the CP-ABE scheme is fully secure, if for any adversary, the advantage of the full-CP-ABE game is negligible.

In the discussion below, we also require that the number of key queries (both in Phase 1 and Phase 2) must be bounded by a pre-determined polynomial Q. This is because that we currently only have a bounded non-adaptive aIPE scheme (which will be explained in the next section). If we have an unbounded adaptive aIPE scheme, this restriction can be removed.

2.3 Discrete Gaussian, Lattice Trapdoor and Learning with Errors

Definition 2.4. *[29]*

For any vector $\mathbf{x} \in \mathbb{Z}^m$, let $\rho_s(\mathbf{x}) = \exp(-\pi\|\mathbf{x}\|^2/s^2)$. For $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and $\mathbf{u} \in \mathbb{Z}_q^n$, let $\Lambda_{\mathbf{u}}^{\perp}(\mathbf{A}) = \{\mathbf{x} : \mathbf{A}\mathbf{x} = \mathbf{u}\}$ (which is a lattice coset).

The discrete Gaussian distribution $D_{\Lambda_{\mathbf{u}}^{\perp}(\mathbf{A}),s}$ is defined as:

$$D_{\Lambda_{\mathbf{u}}^{\perp}(\mathbf{A}),s}(\mathbf{x}) = \frac{\rho_s(\mathbf{x})}{\sum_{\mathbf{v}\in\Lambda_{\mathbf{u}}^{\perp}(\mathbf{A})} \rho_s(\mathbf{v})}.$$

We also write $\rho_s(\Lambda_{\mathbf{u}}^{\perp}(\mathbf{A})) = \sum_{\mathbf{v}\in\Lambda_{\mathbf{u}}^{\perp}(\mathbf{A})} \rho_s(\mathbf{v})$.

The following lemma in [17, 26] shows that there exists a trapdoor and a preimage sampling algorithm for discrete Gaussian distribution.

Lemma 2.1. *[17, 26]*

There is an efficient randomized algorithm $\mathsf{TrapSamp}(1^n, 1^m, q)$ *that, given $n \geq 1$, $q \geq 2$, $m = \Omega(n \log q)$, outputs $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and a "trapdoor" $\mathbf{T}$ such that the distribution of $\mathbf{A}$ is close to uniform with negligible distance.*

Moreover, there is an efficient randomized algorithm SamplePre *that for any $\mathbf{u} \in \mathbb{Z}_q^n$, $s = \Omega(\sqrt{n \log q})$,* $\mathsf{SamplePre}(\mathbf{A}, \mathbf{T}, \mathbf{u}, s)$ *outputs a vector $\mathbf{r} \in \mathbb{Z}^m$, which distribution is statistically close to $D_{\Lambda_{\mathbf{u}}^{\perp}(\mathbf{A}),s}$ with negligible distance.*

We sometimes omit the parameter s if there is no confusion.

The following lemma is required for our security proof:

Lemma 2.2. *Let* $(\mathbf{A}, \mathbf{T}) \leftarrow \mathsf{TrapSamp}(1^n, 1^m, q)$*,* $(\mathbf{A}', \mathbf{T}') \leftarrow \mathsf{TrapSamp}(1^{n'}, 1^m, q)$*,* $n' > n$*, and we write* $\mathbf{A}' = \binom{\bar{\mathbf{A}}}{\tilde{\mathbf{A}}}$*,* $\bar{\mathbf{A}} \in \mathbb{Z}_q^{n \times m}$*, and* $\tilde{\mathbf{A}} \in \mathbb{Z}_q^{(n'-n) \times m}$*. Then there exists* $s > 0$ *such that the following two distribution are statistically indistinguishable:*

- $\mathbf{A}, \mathbf{x} \leftarrow \mathsf{SamplePre}(\mathbf{A}, \mathbf{T}, \mathbf{u}, s)$*;*
- $\bar{\mathbf{A}}, \bar{\mathbf{x}} \leftarrow \mathsf{SamplePre}(\mathbf{A}', \mathbf{T}', \binom{\mathbf{u}}{\mathbf{b}}, s)$*, where* $\mathbf{b} \leftarrow \mathbb{Z}_q^{n'-n}$*.*

Proof. See Appendix A. □

Now we give our hardness assumption: the (decisional) learning with errors (LWE) problem, first introduced in [30]. It has the nice property called worst-case to average-case reduction: solving LWE on the average is as hard as (quantumly) solving GapSVP and SIVP problems in the worst case.

Definition 2.5 (LWE **problem).** *[30] For a vector* $\mathbf{s} \in \mathbb{Z}_q^n$ *called the secret, the LWE distribution* $A_{\mathbf{s},\chi}$ *over* $\mathbb{Z}_q^n \times \mathbb{Z}_q$ *is sampled by choosing* $\mathbf{a} \leftarrow \mathbb{Z}_q^n$ *uniformly at random, choosing* $e \leftarrow \chi$*, and outputting* $(\mathbf{a}, b = \mathbf{s}^T\mathbf{a} + e \ mod \ q)$*.*

The decisional learning with errors (LWE) problem $\mathsf{LWE}_{n,q,\chi,m}$ *is that given* m *independent samples* $(\mathbf{a}_i, b_i) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ *where the samples are distributed according to either* $A_{\mathbf{s},\chi}$ *for a uniformly random* $\mathbf{s}$ *or the uniform distribution, distinguish which is the case with non-negligible advantage.*

For parameters, it is often required that $m = \text{poly}(n)$, $q = O(2^{n^\epsilon})$ for some $\epsilon > 0$, and χ is the discrete Gaussian. We say that the distribution χ is β-bounded, if $|\chi| \leq \beta$ with overwhelming probability. We can choose appropriate parameters for χ to be β-bounded given $\beta = \text{poly}(n)$ such that $\mathsf{LWE}_{n,q,\chi,m}$ is hard.

We give a lemma which will be used in our proof:

Lemma 2.3. *For* $\mathbf{s} \leftarrow \mathbb{Z}_q^n$*, let* $\{(\mathbf{a}_i, b_i)\}_{i \in [m]}$ *be sampled from* $A_{\mathbf{s},\chi}$*. Let* $M \subseteq [m]$*, and* $\{(\mathbf{a}'_i, b'_i)\}_{i \in [m]}$ *be defined as: for* $i \in M$*,* $(\mathbf{a}'_i, b'_i) \leftarrow A_{\mathbf{s},\chi}$*, otherwise* $(\mathbf{a}'_i, b'_i)$ *is uniformly random. Then* $\{(\mathbf{a}_i, b_i)\}_{i \in [m]}$ *and* $\{(\mathbf{a}'_i, b'_i)\}_{i \in [m]}$ *are indistinguishable assuming the hardness of* $\mathsf{LWE}_{n,q,\chi,m}$*.*

Proof. Let $\{(\mathbf{a}^*_i, b^*_i)\}_{i \in [m]}$ be a set of m uniformly random samples, then it is indistinguishable with $\{(\mathbf{a}_i, b_i)\}_{i \in [m]}$ from the hardness of $\mathsf{LWE}_{n,q,\chi,m}$. For those $i \in M$, we replace $(\mathbf{a}^*_i, b^*_i)$ by LWE samples from $A_{\mathbf{s},\chi}$ to get $\{(\mathbf{a}'_i, b'_i)\}_{i \in [m]}$, and the two are also indistinguishable from the hardness of $\mathsf{LWE}_{n,q,\chi,m}$. □

3 Approximate Inner-product Encryption

Definition 3.1. *An approximate IPE scheme consists of the following algorithms:*

- *Setup*(1^λ)*: output a pair* (PK, MSK)*.*
- *KeyGen*$(MSK, \mathbf{x})$*: for* $\mathbf{x} \in \mathbb{Z}_q^m$*, output a secret key* $sk_{\mathbf{x}}$*.*

- *$Enc(PK, \mathbf{m}; r)$: for $\mathbf{m} \in \mathbb{Z}_q^m$, and a random seed $r \leftarrow \mathcal{R}$, output a ciphertext $ct_{\mathbf{m};r}$.*
- *$Dec(ct_{\mathbf{m};r}, sk_{\mathbf{x}})$: Output an approximate inner product for $\mathbf{m}, \mathbf{x}$.*

An aIPE scheme is γ-correct if for any $ct_{\mathbf{m};r}, sk_{\mathbf{x}}$, $Dec(ct_{\mathbf{m};r}, sk_{\mathbf{x}}) - \langle \mathbf{m}, \mathbf{x} \rangle \ mod\ q \leq \gamma$. We say that aIPE has simulatable error, if there exists a function h such that $Dec(ct_{\mathbf{m};r}, sk_{\mathbf{x}}) - \langle \mathbf{m}, \mathbf{x} \rangle \ mod\ q = h(r, \mathbf{x}, \langle \mathbf{m}, \mathbf{x} \rangle)$. Furthermore, we say that the error follows distribution $\bar{\chi}$, if $h(r, \mathbf{x}, \langle \mathbf{m}, \mathbf{x} \rangle) \leftarrow \bar{\chi}$ for a uniformly random chosen seed r and any possible $\mathbf{m}, \mathbf{x}$.

For the simplicity of our further discussion, we explicitly write down the random seed used in *Enc*, so that *Enc* becomes a deterministic algorithm. *Setup* and *KeyGen* are still probabilistic algorithms. Sometimes we can omit r, write the encryption algorithm as $Enc(PK, \mathbf{m})$ and the ciphertext as $ct_{\mathbf{m}}$. The error-simulatable property is required by the simulation-based security below.

As we mentioned above, our aIPE scheme is a direct instance of functional encryption for arbitrary circuits [18]. So we also adopt the security definition from [18] (using a slightly different description).

Definition 3.2. *An aIPE scheme (which has simulatable error) is Q-selective non-adaptive simulation-based secure, if there exists a simulator algorithm S such that for any adversary, the advantage of winning the following game is negligible:*

***Init**. The adversary chooses a challenge message $\mathbf{m}$ and gives it to the challenger.*

***Setup**. The challenger runs the **Setup** algorithm and gives the adversary PK.*

***Key Query**. The adversary submits a vector $\mathbf{x}$ for a **KeyGen** query. The challenger answers with a secret key $\mathsf{sk}_{\mathbf{x}}$ for $\mathbf{x}$. These queries can be repeated adaptively for at most Q times.*

***Challenge**. The challenger chooses a random bit $b \in \{0, 1\}$, and a random seed $r \leftarrow \mathcal{R}$. If $b = 0$, it returns $\mathsf{ct}_{\mathbf{m}} = Enc(PK, \mathbf{m}; r)$ to the adversary. If $b = 1$, it returns $ct_{\mathbf{m}} = S(PK, \{\mathsf{sk}_{\mathbf{x}_i}, \mathbf{x}_i, \langle \mathbf{x}_i, \mathbf{m} \rangle + e_i\}_{i \in [Q]})$, where $e_i = h(r, \mathbf{x}_i, \langle \mathbf{x}_i, \mathbf{m} \rangle)$, $\mathbf{x}_i$ is the queried vector in the i-th key query.*

***Guess**. The adversary outputs a guess b' for b, and the winning advantage is defined as $|Pr[b' = b] - 1/2|$.*

In the definition above, we require that the number of key queries must be *bounded* by a pre-determined Q.

The term "non-adaptive" has multiple meanings in cryptography. In the definition above, "non-adaptive" means that the adversary cannot ask for key queries after seeing the challenge ciphertext, which is defined in [4] for functional encryption. They also prove that adaptive FE scheme for arbitrary circuits simply does not exist, and since we rely on the existence of FE to construct our aIPE scheme, we have to use the non-adaptive definition. We give the following lemma, and put the aIPE construction in the appendix.

Lemma 3.1 *Let $\gamma = \Omega(2^{\lambda^\epsilon})$ for some $\epsilon > 0$. Then there exists a γ-correct aIPE scheme for some γ with simulatable error which distribution is indistinguishable from uniform in $[-\gamma, \gamma]$, with Q-selective non-adaptive simulation-based security for any polynomial Q, if there exists a functional encryption for arbitrary circuits with Q-selective non-adaptive simulation-based security.*

Proof. See Appendix B. □

Since FE scheme for arbitrary circuits with bounded key queries has already been constructed in [18] under LWE assumption and circular security (used in fully-homomorphic encryption schemes), we have the following result:

Corollary 3.2 *Let $\gamma = \Omega(2^{\lambda^\epsilon})$ for some $\epsilon > 0$. Then there exists a γ-correct aIPE scheme for some γ with simulatable error which distribution is indistinguishable from uniform in $[-\gamma, \gamma]$, with Q-selective non-adaptive simulation-based security for any polynomial Q, under the hardness of LWE problem along with circular security.*

Above we define the simulation-based security. However, what we exactly need in our ABE construction is the indistinguishable-based security. As it was shown in [4], simulation-based security is stronger than indistinguishable-based. We give the definition, and the security result we require:

Definition 3.3. *An aIPE scheme (which has simulatable error) is Q-selective non-adaptive β-indistinguishable-based secure, if for any adversary, the advantage of winning the following game is negligible:*

Init. *The adversary chooses two challenge messages $\mathbf{m}_0, \mathbf{m}_1$ and gives it to the challenger.*

Setup. *The challenger runs the **Setup** algorithm and gives the adversary PK.*

Key Query. *The adversary submits a vector $\mathbf{x}$ for a **KeyGen** query. The challenger answers with a secret key $\mathsf{sk}_\mathbf{x}$ for $\mathbf{x}$. These queries can be repeated adaptively for at most Q times.*

Challenge. *The challenger first checks whether for all queried $\mathbf{x}$, there is $|\langle \mathbf{x}, \mathbf{m}_0 - \mathbf{m}_1 \rangle| \le \beta$. If this does not hold, then the challenger aborts. Otherwise, it chooses a random bit $b \in \{0,1\}$, a random seed $r \leftarrow \mathcal{R}$, and returns $\mathsf{ct}_\mathbf{m} = Enc(PK, \mathbf{m}_b; r)$ to the adversary.*

Guess. *The adversary outputs a guess b' for b, and the winning advantage is defined as $|Pr[b' = b] - 1/2|$.*

Lemma 3.3 *Let $\beta/\gamma = O(2^{-\lambda^\epsilon})$ for some $\epsilon > 0$. Then any γ-correct, Q-selective non-adaptive simulation-based secure aIPE for polynomial Q, which error follows uniform distribution in $[-\gamma, \gamma]$, is Q-selective non-adaptive β-indistinguishable-based secure.*

Proof. We prove this by a hybrid of games. Let S be the simulator in the simulation-based security definition.

- Game 0 is the original indistinguishable game.
- In Game 1, if $b = 0$, the challenger answers the challenge ciphertext by the simulator: $S(PK, \{\mathsf{sk}_{\mathbf{x}_i}, \mathbf{x}_i, \langle \mathbf{x}_i, \mathbf{m}_0\rangle + e_i\}_{i\in[Q]})$.
- In Game 2, the challenger answers the challenge ciphertext by the simulator: $S(PK, \{\mathsf{sk}_{\mathbf{x}_i}, \mathbf{x}_i, \langle \mathbf{x}_i, \mathbf{m}_b\rangle + e_i\}_{i\in[Q]})$.

Game 0 and Game 1, Game 1 and Game 2 are indistinguishable by the simulation-based security. We now show that the advantage for any adversary to win Game 2 is negligible.

By our assumption, we have that $\langle \mathbf{x}_i, \mathbf{m}_0\rangle + e_i \leftarrow [\langle \mathbf{x}_i, \mathbf{m}_0\rangle - \gamma, \langle \mathbf{x}_i, \mathbf{m}_0\rangle + \gamma]$, and $\langle \mathbf{x}_i, \mathbf{m}_1\rangle + e_i \leftarrow [\langle \mathbf{x}_i, \mathbf{m}_0\rangle - \langle \mathbf{x}_i, \mathbf{m}_0 - \mathbf{m}_1\rangle - \gamma, \langle \mathbf{x}_i, \mathbf{m}_0\rangle - \langle \mathbf{x}_i, \mathbf{m}_0 - \mathbf{m}_1\rangle + \gamma]$. Then, we can see that the statistical distance between $\langle \mathbf{x}_i, \mathbf{m}_0\rangle + e_i$ and $\langle \mathbf{x}_i, \mathbf{m}_1\rangle + e_i$ is $\frac{2\langle \mathbf{x}_i, \mathbf{m}_0 - \mathbf{m}_1\rangle}{2\gamma+1} \leq \frac{2\beta}{2\gamma+1} < \beta/\gamma$. Let Q be the number of total KeyGen queries, so the statistical distance between $S(PK, \{\mathsf{sk}_{\mathbf{x}_i}, \mathbf{x}_i, \langle \mathbf{x}_i, \mathbf{m}_0\rangle + e_i\}_{i\in[Q]})$ and $S(PK, \{\mathsf{sk}_{\mathbf{x}_i}, \mathbf{x}_i, \langle \mathbf{x}_i, \mathbf{m}_1\rangle + e_i\}_{i\in[Q]})$ is at most $Q\beta/\gamma$. Since Q is polynomial in λ, $Q\beta/\gamma$ is negligible. So the advantage for any adversary to win Game 2 is negligible. □

4 Fully Secure CP-ABE Scheme for CNF policies

Now we give our main theorem:

Theorem 4.1. *There exists a fully secure CP-ABE scheme for CNF policies, assuming the existence of a selectively non-adaptively simulation-based secure aIPE scheme and the hardness of LWE problem.*

We combine Theorem 4.1 and Corollary 3.2, and immediately get the following result:

Corollary 4.2 *There exists a fully secure CP-ABE scheme for CNF policies, assuming the the hardness of LWE problem along with circular security.*

4.1 Construction

Let aIPE be a γ-secure aIPE scheme defined in Section 3, we choose a β-bounded error distribution χ, where $\beta/\gamma = O(2^{-\lambda^\epsilon})$ for some $\epsilon > 0$. The CP-ABE scheme is constructed as follows:

- Setup: Let l be the number of literals. Run TrapSamp $l+1$ times to generate: $(\mathbf{A}_1, \mathbf{T}_1), ..., (\mathbf{A}_l, \mathbf{T}_l), (\mathbf{A}, \mathbf{T}) \leftarrow \mathsf{TrapSamp}(1^n, 1^m, q)$, and let $\mathbf{u} \in \mathbb{Z}_q^n$. Run $\mathsf{aIPE}.\mathit{Setup}$ $l+1$ times to generate $(PK_1, MSK_1), ..., (PK_l, MSK_l), (PK, MSK)$. Output $mpk = (\mathbf{A}_1, ..., \mathbf{A}_l, \mathbf{A}, \mathbf{u}, PK_1, ..., PK_l, PK)$, and $msk = (\mathbf{T}_1, ..., \mathbf{T}_l, \mathbf{T}, MSK_1, ..., MSK_l, MSK)$.
- KeyGen(msk, L): Let $\mathbf{a} \leftarrow \mathbb{Z}_q^n$. Sample $\mathbf{x} \leftarrow \mathsf{SamplePre}(\mathbf{A}, \mathbf{T}, \mathbf{a} + \mathbf{u})$, and let $K \leftarrow \mathsf{aIPE}.\mathit{KeyGen}(MSK, \mathbf{x})$. For each $i \in L$, sample $\mathbf{x}_i \leftarrow \mathsf{SamplePre}(\mathbf{A}_i, \mathbf{T}_i, \mathbf{a})$, and let $K_i \leftarrow \mathsf{aIPE}.\mathit{KeyGen}(MSK_i, \mathbf{x}_i)$. Return the secret key $K, \{K_i\}_{i\in L}$.

- Enc(mpk, f, μ), $\mu \in \{0,1\}$: Let $T_1, ..., T_k$ be clauses in f. Generate uniform $\mathbf{s}_1, ..., \mathbf{s}_k \leftarrow \mathbb{Z}_q^n$. For each $j \in T_i$, let $C_{i,j} = \mathsf{aIPE}.Enc(PK_j, \mathbf{s}_i^T \mathbf{A}_j)$. Let $C = \mathsf{aIPE}.Enc(PK, (\sum_{i=1}^k \mathbf{s}_i)^T \mathbf{A})$, and $C' = (\sum_{i=1}^k \mathbf{s}_i)^T \mathbf{u} + \mu \lfloor q/2 \rfloor + \bar{e}$, $\bar{e} \leftarrow \chi$. Return the ciphertext $(\{C_{i,j}\}_{i \in [k], j \in T_i}, C, C')$.
- Dec(ct_f, sk_L): First check if L satisfies the policy f. If $f(L) = 0$, return $\perp$. If $f(L) = 1$, then for each $i \in [k]$, there is at least one literal $l_i \in L \cap T_i$, let $d_i = aIPE.Dec(K_i, C_{i,l_i})$. Let $d = \mathsf{aIPE}.Dec(K, C)$. Calculate $(\sum_{i=1}^k d_i) - d + C'$, if the value is close to 0, return 0; if the value is close to $q/2$, return 1.

Theorem 4.3. *Let $q > 4(l+1)\gamma + 4\beta$, and aIPE is γ-correct. Then the CP-ABE scheme above is correct.*

Proof. First, by the correctness of aIPE, for $j \in L \cap T_i$, $d_i = \mathbf{s}_i^T \mathbf{A}_j \mathbf{x}_j + e_i = \mathbf{s}_i^T \mathbf{a} + e_j$, $|e_j| \leq \gamma$. Also, $d = (\sum_{i=1}^k \mathbf{s}_i)^T \mathbf{A}\mathbf{x} + e = (\sum_{i=1}^k \mathbf{s}_i)^T (\mathbf{a} + \mathbf{u}) + e$, $|e| \leq \gamma$.

So $(\sum_{i=1}^k d_i) - d + C' = \mu \lfloor q/2 \rfloor + \sum_{i=1}^k e_i - e + \bar{e}$, which is $(l+1)\gamma + \beta$-close to 0 or $\lfloor q/2 \rfloor$. Since $(l+1)\gamma + \beta < q/4$, we can get the correct message. □

4.2 Hyper-functional Keys and Semi-functional Ciphertexts

Now we are ready to prove Theorem 4.1. But before we start the security proof, we first define hyper-functional secret keys and semi-functional ciphertexts.

Hyper-functional key. For a hyper-functional key, we not only change the key generation algorithm, but also the setup algorithm. In *Setup*, instead of generating $\mathbf{A}$ along with its trapdoor, we generate $\mathbf{A}' \in \mathbb{Z}_q^{(n+1) \times m}$ along with its trapdoor $\mathbf{T}'$. We write the first n rows of $\mathbf{A}'$ as $\mathbf{A}$, and the last row as $\tilde{\mathbf{a}}^T$, which means that $\mathbf{A}' = \binom{\mathbf{A}}{\tilde{\mathbf{a}}^T}$. $\mathbf{A}$ is included in the public key as normal. We also generate $\mathbf{t} \leftarrow \mathbb{Z}_q^n$.

For *KeyGen* queries, we first sample $e', e \leftarrow \chi$. Let $\mathbf{x} \leftarrow \mathsf{SamplePre}(\mathbf{A}', \mathbf{T}', \binom{\mathbf{a}+\mathbf{u}}{\mathbf{t}^T(\mathbf{a}+\mathbf{u}) + e' + \bar{e} + e})$). Then we have $(\tilde{\mathbf{a}}^T - \mathbf{t}^T \mathbf{A})\mathbf{x} = e' + \bar{e} + e \approx 0$. Let $K \leftarrow \mathsf{aIPE}.KeyGen(MSK, \mathbf{x})$ and other key elements generated the same as normal. We say that the secret key is hyper-functional related to $\tilde{\mathbf{a}}^T - \mathbf{t}^T \mathbf{A}$.

Note that we also call a secret key "normal", if $\mathbf{x} \leftarrow \mathsf{SamplePre}(\mathbf{A}', \mathbf{T}', \binom{\mathbf{a}+\mathbf{u}}{b})$) for $b \leftarrow \mathbb{Z}_q$.

For the indistinguishability between hyper-functional and normal keys, we have the following lemma:

Lemma 4.4 *Let $(\mathbf{A}_0, \mathbf{T}_0) \leftarrow \mathsf{TrapGen}(1^n, 1^m, q)$. For $i \in [Q]$ and $\mathbf{a}^i \leftarrow \mathbb{Z}_q^n$, $\mathbf{x}_0^i =$ $\mathsf{SamplePre}(\mathbf{A}_0, \mathbf{T}_0, \mathbf{a}^i)$. Let $(\mathbf{A}', \mathbf{T}') \leftarrow \mathsf{TrapGen}(1^{n+1}, 1^m, q)$, $\mathbf{x}_1^i = \mathsf{SamplePre}(\mathbf{A}', \mathbf{T}', \binom{\mathbf{a}^i}{a'^i + e^i})$), where $\mathbf{A}' = \binom{\mathbf{A}_1}{\tilde{\mathbf{a}}}$, $e^i \leftarrow \chi$, $a'^i \in \mathbb{Z}_q$. Then $(\mathbf{A}_0, \{\mathbf{x}_0^i\}_{i \in [Q]})$ is computationally indistinguishable from $(\mathbf{A}_1, \{\mathbf{x}_1^i\}_{i \in [Q]})$ assuming the hardness of LWE.*

Proof. We prove the lemma by showing the following distributions are pairwise indistinguishable (either statistical or computational).

- (1) Let $(\mathbf{A}'', \mathbf{T}'') \leftarrow \mathsf{TrapGen}(1^{2n+1}, 1^m, q)$, $\mathbf{A}''^T = (\mathbf{A}_2^T | \bar{\mathbf{A}}^T | \bar{\mathbf{a}})$. Let $\mathbf{x}_2^i =$ $\mathsf{SamplePre}(\mathbf{A}'', \mathbf{T}'', ({\mathbf{a}^i}^T | \bar{\mathbf{b}^i}^T | \bar{b}^i)^T)$, where $\bar{\mathbf{b}}^i \leftarrow \mathbb{Z}_q^n$ and $\bar{b}^i \leftarrow \mathbb{Z}_q$. By Lemma 2.2, we have $(\mathbf{A}_0, \{\mathbf{x}_0^i\}_{i \in [Q]})$ is statistically indistinguishable from $(\mathbf{A}_2, \{\mathbf{x}_2^i\}_{i \in [Q]})$.

- (2) We first choose $\tilde{b}^i \leftarrow \mathbb{Z}_q^n$ and write $\bar{b}^i = \tilde{b}^i + {a'}^i$. This does not change the distribution.
- (3) We first choose $\mathbf{s} \leftarrow \mathbb{Z}_q^n$, let $\bar{b'}^i = \mathbf{s}^T\bar{\mathbf{b}}^i + e^i + {a'}^i$, and let ${\mathbf{x}'_2}^i = \mathsf{SamplePre}(\mathbf{A}'', \mathbf{T}'', ({\mathbf{a}^i}^T | \bar{\mathbf{b}^i}^T | \bar{b'}^i)^T)$. By the hardness of LWE problem, any adversary cannot distinguish between $\bar{\mathbf{b}}^i, \tilde{b}^i$ and $\bar{\mathbf{b}}^i, \mathbf{s}^T\bar{\mathbf{b}}^i + e^i$, hence cannot distinguish between $\mathbf{x}_2^i$ and ${\mathbf{x}'_2}^i$.
- (4) Let $\tilde{\mathbf{a}} = \bar{\mathbf{a}} - \bar{\mathbf{A}}^T\mathbf{s}$, and we have $\tilde{\mathbf{a}}^T{\mathbf{x}'_2}^i = {a'}^i + e^i$.
- (5) This time we write ${\mathbf{A}''}^T = (\mathbf{A}_2^T | \bar{\mathbf{A}}^T | \tilde{\mathbf{a}})$, and set ${\mathbf{x}''_2}^i = \mathsf{SamplePre}(\mathbf{A}''$, $\mathbf{T}'', ({\mathbf{a}^i}^T | \bar{\mathbf{b}^i}^T | {a'}^i + e^i)^T)$. Then ${\mathbf{x}'_2}^i$ and ${\mathbf{x}''_2}^i$ are from the same distribution.
- (6) By Lemma 2.2, $(\mathbf{A}_2, \{{\mathbf{x}''_2}^i\}_{i\in[Q]})$ is statistically indistinguishable from $(\mathbf{A}_1, \{\mathbf{x}_1^i\}_{i\in[Q]})$.

□

Semi-functional ciphertext.

A ciphertext is semi-functional, if the ciphertext element C is $\mathsf{aIPE}.Enc(PK, (\sum_{i=1}^k \mathbf{s}_i - \mathbf{t})^T\mathbf{A} + \tilde{\mathbf{a}})$ instead of $\mathsf{aIPE}.Enc(PK, (\sum_{i=1}^k \mathbf{s}_i)^T\mathbf{A})$.

It follows directly from the indistinguishable security of aIPE that a semi-functional ciphertext element is indistinguishable from a normal one if all secret keys are hyper-functional.

Along with hyper-functional keys and semi-functional ciphertexts, we also define temporary hyper-functional keys and i-temporary semi-functional ciphertexts, which will be used in our security proof. We note that in our definition, "hyper-functional" and "temporary hyper-functional" form two independent dimensions: a temporary hyper-functional key can be either normal or hyper-functional.

Temporary hyper-functional key. Let l be the number of literals. Like the definition of hyper-functional keys, we not only change the key generation algorithm, but also the setup algorithm. In *Setup*, instead of generating $\mathbf{A}_j$, $j \in [l]$ along with its trapdoor, we generate $\mathbf{A}'_j \in \mathbb{Z}_q^{(n+1)\times m}$ along with its trapdoor $\mathbf{T}'_j$. We write the first n rows of $\mathbf{A}'_j$ as $\mathbf{A}_j$, and the last row as ${\tilde{\mathbf{a}_j}}^T$, which means that $\mathbf{A}'_j = \binom{\mathbf{A}_j}{{\tilde{\mathbf{a}_j}}^T}$. $\mathbf{A}_j$ is included in the public key as normal.

For *KeyGen* queries, let L be the queried literal set. For $j \in L$, let $\mathbf{x}_j \leftarrow \mathsf{SamplePre}(\mathbf{A}'_j, \mathbf{T}'_j, \binom{\mathbf{a}}{\mathbf{t}^T\mathbf{a} + e' + e_j})$), where $e_j \leftarrow \chi$, and if the key is normal, we sample $e' \leftarrow \chi$, if the key is hyper-functional, we use the same e' as in the generation of $\mathbf{x}$. Then we have $({\tilde{\mathbf{a}_j}}^T - \mathbf{t}^T\mathbf{A}_j)\mathbf{x}_j = e' + e_j \approx 0$. Let $K_j \leftarrow \mathsf{aIPE}.KeyGen(MSK_j, \mathbf{x}_j)$. We say that the secret key is temporary hyper-functional related to $\{\tilde{\mathbf{a}}_j^T - \mathbf{t}^T\mathbf{A}_j\}_{j\in L}$.

We can also use Lemma 4.4 to prove the indistinguishability between temporary hyper-functional keys and non-temporary normal/hyper-functional keys.

i-Temporary semi-functional ciphertext. A ciphertext is i-temporary semi-functional, if for each $j \in T_i$, the ciphertext element $C_{i,j}$ is $\mathsf{aIPE}.Enc(PK_j, (\mathbf{s}_i - \mathbf{t})^T\mathbf{A}_j + \tilde{\mathbf{a}_j})$ instead of $\mathsf{aIPE}.Enc(PK_j, \mathbf{s}_i^T\mathbf{A}_j)$.

It follows directly from the indistinguishable security of aIPE that a temporary semi-functional ciphertext element is indistinguishable from a semi-functional one if all secret keys but those $sk_L, L \cap T_i = \emptyset$ are temporary hyper-functional.

4.3 Security Proof

We first give the outline of our proof. Our security proof is similar to [25], which is constructed using dual system encryption from bilinear groups.

- Switch all queried secret keys into hyper-functional keys.
- Switch the challenge ciphertext into semi-functional ciphertext.
- For the p-th query in Phase 1 which challenge literal set is L:
  - Switch all secret keys into temporary hyper-functional secret keys.
  - Find an i such that $L \cap T_i = \emptyset$, and switch the ciphertext into i-temporary semi-functional ciphertext.
  - Switch the p-th secret key into a normal one using LWE assumption.
  - Switch the ciphertext into a non-temporary semi-functional ciphertext.
  - Switch all secret keys into non-temporary normal or hyper-functional secret keys.
- For queries in Phase 2, and $i \in [k]$, k is the maximal number of clauses:
  - Switch all secret keys into temporary hyper-functional secret keys.
  - Switch the ciphertext into i-temporary semi-functional ciphertext.
  - Switch all Phase 2 secret keys such that $L \cap T_i = \emptyset$ into a normal one using LWE assumption.
  - Switch the ciphertext into a non-temporary semi-functional ciphertext.
  - Switch all secret keys into non-temporary normal or hyper-functional secret keys.
- Now C is uniformly random, independent with any queried secret keys. We further switch C' into a uniformly random element, and thus have our result.

Now we define the game sequence.

Game 0 is the original game.

Game 1 is same as **Game** 0 except that each queried secret key is a hyper-functional key. **Game** 0 and **Game** 1 are indistinguishable by Lemma 4.4.

Game 2 is same as **Game** 1 except that the challenge ciphertext is semi-functional. We note that the security definition of aIPE is "non-adaptive", which means that we cannot query secret keys after seeing the challenge aIPE ciphertext. So in order to use Lemma 3.3, we first define **Game** 1a and **Game** 2a as follows:

- The Setup phase and Phase 1 are the same as Game 1 or Game 2.
- In the Challenge phase, let Q_1 be the number of Phase 1 queries, so $Q_2 = Q - Q_1$ is the maximal number of Phase 2 queries. Before the challenge ciphertext is given, for each $r \in [Q_2]$, we generate $\mathbf{a}^r \leftarrow \mathbb{Z}_q^n$, ${e'}^r, e^r \leftarrow \chi$, and $\mathbf{x}^r \leftarrow \mathsf{SamplePre}(\mathbf{A}', \mathbf{T}', \left(\begin{smallmatrix}\mathbf{a}^r+\mathbf{u}\\ \mathbf{t}^T(\mathbf{a}^r+\mathbf{u})+{e'}^r+\bar{e}+e^r\end{smallmatrix}\right)$. Let $K^r = \mathsf{aIPE}.KeyGen(MSK, \mathbf{x}^r)$.
- In Game 1a, the challenger returns a normal ciphertext, and in Game 2a, it returns a semi-functional one.

- In the r-th Phase 2 query, we let $e' = e'^r$, $e = e^r$, $\mathbf{a} = \mathbf{a}^r$, and the key element $K = K^r$. Other key elements are generated as before.

It is easy to see that **Game** 1 and **Game** 1a; **Game** 2 and **Game** 2a are the same from the adversary's point of view. We now show that **Game** 1a and **Game** 2a are indistinguishable.

For the challenger, instead of generating all Ks and C itself, it now runs an indistinguishable game for aIPE, gets K by the KeyGen query of aIPE, and gets C as the challenge ciphertext of aIPE. We can see that all KeyGen queries are made before generating the challenge ciphertext of aIPE, so the aIPE game can proceed correctly. Because $|(\tilde{\mathbf{a}}^T - \mathbf{t}^T\mathbf{A})\mathbf{x}| \leq 3\beta$ and $\beta/\gamma = O(2^{-\lambda^\epsilon})$, we have the indistinguishable result by Lemma 3.3.

Game $2(p)$, $p \in [Q_1 + 1]$, Q_1 is the number of phase 1 queries: **Game** $2(p)$ is same as **Game** 2, except that the first $p - 1$ Phase 1 keys are normal. Then **Game** 2(1) is **Game** 2, and in **Game** $2(Q_1 + 1)$, all Phase 1 keys are normal. We prove the following result:

Lemma 4.5 ***Game*** *2(p) and* ***Game*** *2(p + 1) are indistinguishable assuming the security of aIPE and the hardness of LWE.*

Proof. We prove this by the following game sequence:

Game 2-1(p): **Game** 2-1(p) is same as **Game** 2(p), except that we change all keys into temporary hyper-functional keys. **Game** 2-1(p) is indistinguishable from **Game** 2(p) according to Lemma 4.4.

Let L be the challenge literal set in the p-th query of Phase 1. So there must be clause T_i such that $L \cap T_i = \emptyset$. This i will be used in the following games.

Game 2-2(p, j): **Game** 2-2(p, j) is same as **Game** 2-1(p), except that for any $C_{i,j'}$ such that $j' \leq j$ and $j' \in T_i$, $C_{i,j'}$ is generated as $\mathsf{aIPE}.Enc(PK_{j'}, (\mathbf{s}_i - \mathbf{t})^T\mathbf{A}_{j'} + \tilde{\mathbf{a}}_{j'}^T)$. So **Game** 2-2($p, 0$) is **Game** 2-1($p$), and in **Game** 2-2($p, l$), the ciphertext is i-temporary semi-functional. We now show that **Game** 2-2($p, j-1$) is indistinguishable from Game 2-2(p, j).

Similar to the discussion above, we must generate the required secret key element at the challenge phase, in order to use Lemma 3.3. Let $Q_2 = Q - Q_1$ be the maximal number of Phase 2 queries. We define **Game** 2-2a(p, j) and **Game** 2-2b(p, j) as follows:

Game 2-2a(p, j): The game is same as **Game** 2-2(p, j), except that:

- In the Challenge phase, before the challenge ciphertext is given, we first check whether $j + 1 \in T_i$. If $j + 1 \notin T_i$, the game proceeds as **Game** 2-2(p, j). If $j+1 \in T_i$, for each $r \in [Q_2]$, we generate $\mathbf{a}^r \leftarrow \mathbb{Z}_q^n$, $e'^r, e_{j+1}^r \leftarrow \chi$, and $\mathbf{x}_{j+1}^r \leftarrow \mathsf{SamplePre}(\mathbf{A}'_{j+1}, \mathbf{T}'_{j+1}, \binom{\mathbf{a}^r}{\mathbf{t}^T\mathbf{a}^r + e'^r + e_{j+1}^r}))$. Let $K_{j+1}^r = aIPE.KeyGen(MSK_{j+1}, \mathbf{x}_{j+1}^r)$.
- In the r-th Phase 2 query, if $j + 1 \in T_i$, we let $e' = e'^r$, $e_{j+1} = e_{j+1}^r$, $\mathbf{a} = \mathbf{a}^r$, and the key element $K_{j+1} = K_{j+1}^r$. Then, generate other key elements as in **Game** 2-2(p, j).

Game 2-2b(p, j): The game is same as **Game** 2-2(p, j), except that:

- In the Challenge phase, before the challenge ciphertext is given, we first check whether $j \in T_i$. If $j \notin T_i$, the game proceeds as **Game** 2-2(p, j). If $j \in T_i$, for each $r \in [Q_2]$, we generate $\mathbf{a}^r \leftarrow \mathbb{Z}_q^n$, $e'^r, e_j^r \leftarrow \chi$, and $\mathbf{x}_j^r \leftarrow \mathsf{SamplePre}(\mathbf{A}'_j, \mathbf{T}'_j, \left(\begin{smallmatrix} \mathbf{a}^r \\ \mathbf{t}^T\mathbf{a}^r + e'^r + e_j^r \end{smallmatrix}\right))$. Let $K_j^r = aIPE.KeyGen(MSK_j, \mathbf{x}_j^r)$.
- In the r-th Phase 2 query, if $j \in T_i$, we let $e' = e'^r$, $e_j = e_j^r$, $\mathbf{a} = \mathbf{a}^r$, and the key element $K_j = K_j^r$. Then, generate other key elements as in **Game** 2-2(p, j).

It is easy to see that **Game** 2-2(p, j), **Game** 2-2a(p, j) and **Game** 2-2b(p, j) are the same from the adversary's point of view. We now show that **Game** 2-2a$(p, j-1)$ and **Game** 2-2b(p, j) are indistinguishable.

For the challenger, instead of generating all K_js and $C_{i,j}$ itself, it now runs a indistinguishable game for aIPE, get K_j by the KeyGen query of aIPE, and get $C_{i,j}$ as the challenge ciphertext of aIPE. Since $|(\tilde{\mathbf{a}_j}^T - \mathbf{t}^T\mathbf{A}_j)\mathbf{x}_j| \leq 2\beta$ and $\beta/\gamma = O(2^{-\lambda^\epsilon})$ by assumption, we only need to show that the aIPE game can proceed correctly. If $j \in T_i$, all KeyGen queries are made before the challenge ciphertext, which is legal in the aIPE game. If $j \notin T_i$, the aIPE challenge ciphertext is never required, so all KeyGen queries can be made correctly. Thus we have the indistinguishable result by Lemma 3.3.

Now we have that **Game** 2-1(p) is indistinguishable from **Game** 2-2(p, l).

Game 2-3(p): The game is same as **Game** 2-2(p, l), except that:

- In the challenge phase, we generate $\bar{\mathbf{s}} \leftarrow \mathbb{Z}_q^n$, $\{\mathbf{s}_{i'}\}_{i' \neq i} \leftarrow \mathbb{Z}_q^n$, and generate $C_{i,j}$ for any $j \in T_i$ as $\mathsf{aIPE}.Enc(PK_j, (\bar{\mathbf{s}} - \sum_{i' \neq i} \mathbf{s}_{i'})^T\mathbf{A}_j + \tilde{\mathbf{a}}_j)$.
- We also generate $C = \mathsf{aIPE}.Enc(PK, \bar{\mathbf{s}}^T\mathbf{A} + \tilde{\mathbf{a}}^T)$, and $C' = (\bar{\mathbf{s}} + \mathbf{t})^T\mathbf{u} + \mu\lfloor q/2 \rfloor + \bar{e}$.

Note that in **Game** 2-3(p), we implicitly set $\mathbf{s}_i = \bar{\mathbf{s}} + \mathbf{t} - \sum_{i' \neq i} \mathbf{s}_{i'}$, so that for the adversary, **Game** 2-3(p) is the same as **Game** 2-2(p, l). Now we see that $\mathbf{t}$ only occurs in C' and in KeyGen queries. All these occurrences of $\mathbf{t}$ take the form of LWE samples: $\mathbf{t}^T\mathbf{a} + e'$, and $\mathbf{t}^T\mathbf{u} + \bar{e}$.

Game 2-4(p): The game is same as **Game** 2-3(p), except that in the p-th query:

- We choose random $\tilde{b} \leftarrow \mathbb{Z}_q$, and let $\mathbf{x} \leftarrow \mathsf{SamplePre}(\mathbf{A}', \mathbf{T}', \left(\begin{smallmatrix} \mathbf{a} \\ \tilde{b} + \mathbf{t}^T\mathbf{u} + \bar{e} + e \end{smallmatrix}\right))$.
- For $i \in [l]$, $\mathbf{x}_i \leftarrow \mathsf{SamplePre}(\mathbf{A}'_i, \mathbf{T}'_i, \left(\begin{smallmatrix} \mathbf{a} \\ \tilde{b} + e_i \end{smallmatrix}\right))$.

Game 2-3(p) and **Game** 2-4(p) are indistinguishable using Lemma 2.3, by the hardness of LWE problem. We also define **Game** 2-4a(p), which removes $\bar{\mathbf{s}}$ from **Game** 2-4(p), and $\mathbf{s}_i$ is uniformly sampled in the challenge phase. **Game** 2-4a(p) is the same as **Game** 2-4(p) from the adversary's point of view.

Game 2-5(p, j): **Game** 2-5(p, j) is same as **Game** 2-4a(p), except that for any $C_{i,j'}$ such that $j' \leq j$ and $j' \in T_i$, $C_{i,j'}$ is generated as $aIPE.Enc(PK_{j'}, \mathbf{s}_i^T\mathbf{A}_{j'})$. So **Game** 2-5$(p, 0)$ is **Game** 2-4a(p), and in **Game** 2-5(p, l), the ciphertext is (non-temporary) semi-functional.

The indistinguishability between **Game** 2-5$(p, j-1)$ and **Game** 2-5(p, j) is nearly the same as **Game** 2-2$(p, j-1)$ and **Game** 2-2(p, j), except that this time, for the p-th query with literal set L, $|(\tilde{\mathbf{a}_j}^T - \mathbf{t}^T\mathbf{A}_j)\mathbf{x}_j|$ may not be small. However, since $L \cap T_i = \emptyset$, for each $j \in T_i$ where it is required to generate the ciphertext element $C_{i,j}$, the corresponding key element $\mathbf{x}_j$ does not exist. So the ciphertext can be generated correctly in the reduction. Now we have that **Game** 2-4a(p) is indistinguishable from **Game** 2-5(p, l).

Game 2-6(p): **Game** 2-6(p) is same as **Game** 2-5(p, l) except that in the p-th KeyGen query, instead of generating random $\tilde{b}$, we sample $b \leftarrow \mathbb{Z}_q$, and set $\tilde{b} = b - \mathbf{t}^T\mathbf{u} - \bar{e} - e$. **Game** 2-6($p$) is same as **Game** 2-5(p, l) from the adversary's point of view. We can see that **Game** 2-6(p) is indistinguishable from **Game** 2($p + 1$) from Lemma 4.4. □

Game 3(i), $i \in [k + 1]$, k is the number of clauses in the challenge access policy: The Phase 1 keys are normal, and for each Phase 2 key which challenge literal set is L, the key is normal iff there exists $i' < i$ such that $L \cap T_{i'} = \emptyset$. **Game** 3(1) is the same as **Game** 2($Q_1 + 1$). Since L must not satisfy the access policy, it is easy to see that in **Game** 3($k + 1$), all keys are normal.

Lemma 4.6 *__Game__ 3(i) and __Game__ 3($i+1$) are indistinguishable assuming the security of aIPE and the hardness of LWE.*

Proof. The proof is essentially the same as Lemma 4.5. We omit the details here. □

Game 4: **Game** 4 is same as **Game** 3($k + 1$), except that all secret keys are temporary hyper-functional keys. **Game** 4 is indistinguishable from **Game** 3($k + 1$) by Lemma 4.4.

Game 5: **Game** 5 is same as **Game** 4, except that the challenge ciphertext is 1-temporary semi-functional. Using similar discussion from **Game** 2-2(p, j) in Lemma 4.5, we have that **Game** 4 and **Game** 5 are indistinguishable by Lemma 3.3.

Game 6: The game is same as **Game** 5, except that:

– In the challenge phase, we generate $\bar{\mathbf{s}} \leftarrow \mathbb{Z}_q^n$, $\{\mathbf{s}_{i'}\}_{i' \neq 1} \leftarrow \mathbb{Z}_q^n$, and write $C_{1,j}$ for any $j \in T_1$ as $\mathsf{aIPE}.Enc(PK_j, (\bar{\mathbf{s}} - \sum_{i' \neq 1} \mathbf{s}_{i'})^T\mathbf{A}_j + \tilde{\mathbf{a}}_j)$.
– We also write $C = \mathsf{aIPE}.Enc(PK, \bar{\mathbf{s}}^T\mathbf{A} + \tilde{\mathbf{a}}^T)$, and $C' = (\bar{\mathbf{s}} + \mathbf{t})^T\mathbf{u} + \mu\lfloor q/2\rfloor + \bar{e}$.

Game 6 is same as **Game** 5 from the adversary's point of view. Note that this time, $\mathbf{t}^T\mathbf{u}$ only occurs in C'.

Game 7: The game is same as **Game** 6, except that in the challenge phase, C' is generated by $\bar{\mathbf{s}}^T\mathbf{u} + v + \mu\lfloor q/2\rfloor$, $v \leftarrow \mathbb{Z}_q$. **Game** 7 is indistinguishable from **Game** 6 by Lemma 2.3 from LWE assumption.

Game 8: The game is same as **Game** 7, except that in the challenge phase, we let $v' \leftarrow \mathbb{Z}_q$, and $v = v' - \bar{\mathbf{s}}^T\mathbf{u} - \mu\lfloor q/2\rfloor$, so $C' = v'$. **Game** 7 and **Game** 8 are the same from the adversary's point of view. Then in **Game** 8, the ciphertext contains no information on μ, so the advantage for any adversary is 1/2. Thus we finish our proof.

5 Conclusion and Future Works

In this paper, we give a lattice version of the widely used dual-system method from pairing-based cryptography, and use it to prove the full security of a CP-ABE scheme supporting CNF access policies from lattice assumptions. The expressibility of our access policies is stronger than the existing result, which can only support t-CNF policies for a constant t. We also point out that this is the first time of using dual-system method to prove the full security of a lattice-based ABE scheme, which solves a long time open question.

Our scheme is only the first and simple fully secure ABE construction using dual system encryption in lattice, while the potential of our method seems to be more than what we have shown in this paper. If our method has the same ability as what has been proven for dual system encryption in bilinear groups, we hope that we can use it to construct more fully secure ABE schemes, including ABE with stronger expressibility (maybe for arbitrary circuits), constant decryption cost, large universe, or even functional encryption schemes.

We note that dual-system based schemes in bilinear groups have comparable efficiency to earlier schemes, but our scheme is currently impractical, mainly because we have not found an efficient construction for aIPE scheme. Also, since the aIPE scheme in this paper is bounded and non-adaptive, the number of key queries for ABE must be bounded, which makes its security slightly lower than the standard definition. Although we have high confidentiality for the existence of a simple aIPE scheme, the construction seems to be harder than we first imagine. We shall continue to work on the construction of aIPE in order to make lattice-based ABE from dual system truly practical.

References

1. Shweta Agrawal, Dan Boneh, and Xavier Boyen. Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical IBE. In *Advances in Cryptology - CRYPTO 2010*, pages 98–115, 2010.
2. Shweta Agrawal, Xavier Boyen, Vinod Vaikuntanathan, Panagiotis Voulgaris, and Hoeteck Wee. Functional encryption for threshold functions (or fuzzy ibe) from lattices. *International Workshop on Public Key Cryptography.*
3. Shweta Agrawal, David Mandell Freeman, and Vinod Vaikuntanathan. Functional encryption for inner product predicates from learning with errors. *International Conference on the Theory and Application of Cryptology and Information Security*, 2011:21–40, 2011.
4. Shweta Agrawal, Sergey Gorbunov, Vinod Vaikuntanathan, and Hoeteck Wee. Functional encryption: New perspectives and lower bounds. In *Advances in Cryptology - CRYPTO 2013*, pages 500–518, 2013.
5. Shweta Agrawal, Benoît Libert, and Damien Stehlé. Fully secure functional encryption for inner products, from standard assumptions. In *Annual International Cryptology Conference*, pages 333–362. Springer, 2016.
6. Nuttapong Attrapadung. Dual system encryption via doubly selective security: framework, fully secure functional encryption for regular languages, and more. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 557–577. Springer, 2014.

7. John Bethencourt, Amit Sahai, and Brent Waters. Ciphertext-policy attribute-based encryption. *IEEE Symposium on Security and Privacy*, pages 321–334, 2007.
8. Dan Boneh and Xavier Boyen. Efficient selective-id secure identity-based encryption without random oracles. *Annual International Conference on the Theory and Application of Cryptographic Techniques*, 2004:223–238, 2004.
9. Dan Boneh, Craig Gentry, Sergey Gorbunov, Shai Halevi, Valeria Nikolaenko, Gil Segev, Vinod Vaikuntanathan, and Dhinakaran Vinayagamurthy. Fully key-homomorphic encryption, arithmetic circuit abe, and compact garbled circuits. *Annual International Conference on the Theory and Application of Cryptographic Techniques*, 2014:533–556, 2014.
10. Dan Boneh, Amit Sahai, and Brent Waters. Functional encryption: Definitions and challenges. In *Theory of Cryptography Conference*, pages 253–273. Springer, 2011.
11. Xavier Boyen. Attribute-based functional encryption on lattices. *Theory of Cryptography Conference*, 2012:122–142, 2013.
12. Xavier Boyen and Qinyi Li. Towards tightly secure lattice short signature and id-based encryption. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 404–434. Springer, 2016.
13. Zvika Brakerski and Vinod Vaikuntanathan. Circuit-abe from lwe: unbounded attributes and semi-adaptive security. In *Annual International Cryptology Conference*, pages 363–384. Springer, 2016.
14. David Cash, Dennis Hofheinz, Eike Kiltz, and Chris Peikert. Bonsai trees, or how to delegate a lattice basis. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 523–552. Springer, 2010.
15. Jie Chen, Junqing Gong, Lucas Kowalczyk, and Hoeteck Wee. Unbounded abe via bilinear entropy expansion, revisited. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 503–534. Springer, 2018.
16. Ling Cheung and Calvin Newport. Provably secure ciphertext policy abe. In *Proceedings of the 14th ACM conference on Computer and communications security*, pages 456–465, 2007.
17. Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. *Symposium on the Theory of Computing*, 2007:197–206, 2008.
18. Shafi Goldwasser, Yael Tauman Kalai, Raluca A. Popa, Vinod Vaikuntanathan, and Nickolai Zeldovich. Reusable garbled circuits and succinct functional encryption. In *Symposium on Theory of Computing Conference, STOC'13*, pages 555–564, 2013.
19. Sergey Gorbunov, Vinod Vaikuntanathan, and Hoeteck Wee. Attribute-based encryption for circuits. *Symposium on the Theory of Computing*, 2013:545–554, 2013.
20. Rishab Goyal, Venkata Koppula, and Brent Waters. Semi-adaptive security and bundling functionalities made generic and easy. In *Theory of Cryptography Conference*, pages 361–388. Springer, 2016.
21. Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for fine-grained access control of encrypted data. *Proceedings of the 13th ACM conference on Computer and Communications Security*, 2006:89–98, 2006.
22. Lucas Kowalczyk and Allison Bishop Lewko. Bilinear entropy expansion from the decisional linear assumption. In *Annual Cryptology Conference*, pages 524–541. Springer, 2015.

23. Lucas Kowalczyk and Hoeteck Wee. Compact adaptively secure abe for NC^1 from k-lin. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 3–33. Springer, 2019.
24. Allison Lewko, Tatsuaki Okamoto, Amit Sahai, Katsuyuki Takashima, and Brent Waters. Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 62–91. Springer, 2010.
25. Allison B. Lewko and Brent Waters. New proof methods for attribute-based encryption: Achieving full security through selective techniques. In *Advances in Cryptology - CRYPTO 2012*, pages 180–198.
26. Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. *Annual International Conference on the Theory and Application of Cryptographic Techniques*, 2011:700–718, 2012.
27. Daniele Micciancio and Oded Regev. Worst-case to average-case reductions based on gaussian measures. *SIAM Journal on Computing*, 37(1):267–302, 2007.
28. Rafail Ostrovsky, Amit Sahai, and Brent Waters. Attribute-based encryption with non-monotonic access structures. In *Proceedings of the 14th ACM conference on Computer and communications security*, pages 195–203, 2007.
29. Oded Regev. New lattice-based cryptographic constructions. *Journal of the ACM (JACM)*, 51(6):899–942, 2004.
30. Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM*, 56(6):34, 2009.
31. Yannis Rouselakis and Brent Waters. Practical constructions and new proof methods for large universe attribute-based encryption. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pages 463–474, 2013.
32. Amit Sahai and Brent Waters. Fuzzy identity-based encryption. *Annual International Conference on the Theory and Application of Cryptographic Techniques*, 2004:457–473, 2005.
33. Rotem Tsabary. Fully secure attribute-based encryption for t-cnf from LWE. In *Advances in Cryptology - CRYPTO 2019*, pages 62–85, 2019.
34. Zhedong Wang, Xiong Fan, and Feng-Hao Liu. FE for inner products and its application to decentralized ABE. In *Public-Key Cryptography - PKC 2019*, pages 97–127, 2019.
35. Brent Waters. Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In *Advances in Cryptology - CRYPTO 2009*, pages 619–636, 2009.
36. Brent Waters. Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. In *International Workshop on Public Key Cryptography*, pages 53–70. Springer, 2011.
37. Hoeteck Wee. Dual system encryption via predicate encodings. In *Theory of Cryptography Conference*, pages 616–637. Springer, 2014.
38. Shota Yamada. Adaptively secure identity-based encryption from lattices with asymptotically shorter public parameters. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 32–62. Springer, 2016.

A Proof of Lemma 2.2

We first give the following lemma which is proven in [27, 17].

Lemma A.1. *[17]*

For any $\epsilon \in (0,1)$, there exists $\eta > 0$, such that for $s \geq \eta$, $\rho_s(\Lambda^\perp_{\mathbf{u}}(\mathbf{A})) \in [\frac{1-\epsilon}{1+\epsilon}, 1] \cdot \rho_s(\Lambda^\perp_{\mathbf{0}}(\mathbf{A}))$.

By Lemma 2.1, we have that the distribution of $\mathbf{x}$ is statistically close to $D_{\Lambda^\perp_{\mathbf{u}}(\mathbf{A}),s}$. So we only need to show that the distribution of $\mathbf{x}'$ is statistically close to $D_{\Lambda^\perp_{\mathbf{u}}(\mathbf{A}'),s}$.

It is easy to see that $\{\Lambda^\perp_{(\mathbf{u}^T|\mathbf{b}^T)^T}(\mathbf{A}')\}_{\mathbf{b}\in\mathbb{Z}_q^{n'-n}}$ forms a partition of the lattice co-set $\Lambda^\perp_{\mathbf{u}}(\bar{\mathbf{A}})$. So by the definition of discrete Gaussian, we have that, for any $\mathbf{c} \in \Lambda^\perp_{\mathbf{u}}(\bar{\mathbf{A}})$, let $\mathbf{b} = \tilde{\mathbf{A}}\mathbf{c}$, we have $Pr(\mathbf{x} = \mathbf{c}) = q^{-(n'-n)}\rho_s(\mathbf{c})/\rho_s(\Lambda^\perp_{(\mathbf{u}^T|\mathbf{b}^T)^T}(\mathbf{A}'))$. For a negligible ϵ, we choose s satisfies Lemma A.1. Then we have that for any $\mathbf{b}'$, $\rho_s(\Lambda^\perp_{(\mathbf{u}^T|\mathbf{b}^T)^T}(\mathbf{A}'))/\rho_s(\Lambda^\perp_{(\mathbf{u}^T|\mathbf{b}'^T)^T}(\mathbf{A}')) \in [\frac{1-\epsilon}{1+\epsilon}, \frac{1+\epsilon}{1-\epsilon}]$.

By definition, we have:

$$D_{\Lambda^\perp_{\mathbf{u}}(\mathbf{A}'),s}(\mathbf{c}) = \frac{\rho_s(\mathbf{c})}{\rho_s(\Lambda^\perp_{\mathbf{u}}(\bar{\mathbf{A}}))} = \frac{\rho_s(\mathbf{c})}{\sum_{\mathbf{b}'\in\mathbb{Z}_q^{n'-n}} \rho_s(\Lambda^\perp_{(\mathbf{u}^T|\mathbf{b}'^T)^T}(\mathbf{A}'))}.$$

So:

$$\frac{1-\epsilon}{1+\epsilon} \cdot \frac{\rho_s(\mathbf{c})}{q^{n'-n}\rho_s(\Lambda^\perp_{(\mathbf{u}^T|\mathbf{b}^T)^T}(\mathbf{A}'))} \leq D_{\Lambda^\perp_{\mathbf{u}}(\mathbf{A}'),s}(\mathbf{c}) \leq \frac{1+\epsilon}{1-\epsilon} \cdot \frac{\rho_s(\mathbf{c})}{q^{n'-n}\rho_s(\Lambda^\perp_{(\mathbf{u}^T|\mathbf{b}^T)^T}(\mathbf{A}'))}.$$

Now we have that the statistical distance between the two distributions is no more than 2ϵ, thus we have our result.

B Proof of Lemma 3.1

First, we introduce the concept of functional encryption and its security definition before we proceed.

Definition B.1. *A functional encryption for a function class $\mathcal{F} : \mathcal{X} \to \mathcal{Y}$ scheme consists of the following algorithms:*

- *$FE.Setup(1^\lambda)$: output a pair (PK, MSK).*
- *$FE.KeyGen(MSK, f)$: for $f \in \mathcal{F}$, output a secret key sk_f.*
- *$FE.Enc(PK, x)$: for $x \in \mathcal{X}$, output a ciphertext ct_x.*
- *$FE.Dec(ct_x, sk_f)$: Output $f(x)$.*

A functional encryption scheme is correct if the probability for $FE.Dec(ct_x, sk_f) \neq f(x)$ is negligible.

Definition B.2. *A functional encryption scheme is Q-selective non-adaptive simulation-based secure, if there exists a simulator algorithm S such that for any adversary, the advantage of winning the following game is negligible:*

Init. *The adversary chooses a challenge message $x \in \mathcal{X}$ and gives it to the challenger.*

Setup. *The challenger runs the* ***Setup*** *algorithm and gives the adversary* PK.

Key Query. *The adversary submits a function* $f \in \mathcal{F}$ *for a* ***KeyGen*** *query. The challenger answers with a secret key* sk_f *for* f. *These queries can be repeated adaptively for at most* Q *times.*

Challenge. *The challenger chooses a random bit* $b \in \{0,1\}$. *If* $b = 0$, *it returns* $\mathsf{ct}_x = Enc(PK, x)$ *to the adversary. If* $b = 1$, *it returns* $ct_x = S(PK, \{\mathsf{sk}_f, f, f(x)\}_{i\in[Q]})$.

Guess. *The adversary outputs a guess* b' *for* b, *and the winning advantage is defined as* $|Pr[b' = b] - 1/2|$.

Since we already have FE for arbitrary circuits in [18] (Lemma 3.5), we only need to define the function class we need in order to implement aIPE. We choose the modular q and $p << q$, set $\gamma = \lceil (\frac{q}{p} - 1)/2 \rceil$. Let $\bar{h}$ be a psueudorandom function maps $\mathcal{R} \times \mathbb{Z}_q^m$ to $[-\gamma, \gamma]$. The function class $\mathcal{F}$ is defined as:

$$f_{\mathbf{x}}(\mathbf{m}; r) = \lfloor \frac{q}{p} \cdot \lfloor (\langle \mathbf{m}, \mathbf{x} \rangle + \bar{h}(r, \mathbf{x})) \cdot \frac{p}{q} \rceil - \bar{h}(r, \mathbf{x}) \rceil.$$

Let $FE_{\mathcal{F}}$ be the functional encryption scheme supporting $\mathcal{F}$. We only need to show that $FE_{\mathcal{F}}$ satisfies the definition of aIPE. For the simplicity of our discussion, we choose $q = p^k$, p is an odd prime, so that q/p is an integer. (Choosing other q and p only adds a negligible distance onto the error distribution, as long as $p/q = O(2^{-\lambda^{\epsilon}})$ for some $\epsilon > 0$).

We have that $\lfloor (\langle \mathbf{m}, \mathbf{x} \rangle + \bar{h}(r, \mathbf{x})) \cdot \frac{p}{q} \rceil - (\langle \mathbf{m}, \mathbf{x} \rangle + \bar{h}(r, \mathbf{x})) \cdot \frac{p}{q} \in [-1/2, 1/2)$. So $f_{\mathbf{x}}(\mathbf{m}; r) - (\frac{q}{p} \cdot ((\langle \mathbf{m}, \mathbf{x} \rangle + \bar{h}(r, \mathbf{x})) \cdot \frac{p}{q}) - \bar{h}(r, \mathbf{x})) = f_{\mathbf{x}}(\mathbf{m}; r) - \langle \mathbf{m}, \mathbf{x} \rangle \in [-\frac{q}{2p}, \frac{q}{2p})$. Since the error $f_{\mathbf{x}}(\mathbf{m}; r) - \langle \mathbf{m}, \mathbf{x} \rangle$ is an integer, the value is in $[-(\frac{q}{p} - 1)/2, (\frac{q}{p} - 1)/2] = [-\gamma, \gamma]$. It is easy to see that the error can be determined by $\bar{h}(r, \mathbf{x})$ and $\langle \mathbf{m}, \mathbf{x} \rangle$, hence is simulatable.

Also, by the pseudorandomness of $\bar{h}$, we see that $\bar{h}(r, \mathbf{x})$, $r \leftarrow \mathcal{R}$ is indistinguishable from uniform in $[-\gamma, \gamma]$. It is also straight to show that for different $\bar{h}(r, \mathbf{x})$, $\lfloor (\langle \mathbf{m}, \mathbf{x} \rangle + \bar{h}(r, \mathbf{x})) \cdot \frac{p}{q} \rceil - (\langle \mathbf{m}, \mathbf{x} \rangle + \bar{h}(r, \mathbf{x})) \cdot \frac{p}{q}$ takes different values, hence the error $f_{\mathbf{x}}(\mathbf{m}; r) - \langle \mathbf{m}, \mathbf{x} \rangle$ takes different values. If the output of $\bar{h}$ is uniform, the error takes $2\gamma + 1$ different values each with probability $1/(2\gamma + 1)$, and is uniformly random. Since $\bar{h}$ is indistinguishable from uniform, the error distribution is also indistinguishable form uniform. Thus we finish the proof.