Partitioning via Non-Linear Polynomial Functions: More Compact IBEs from Ideal Lattices and Bilinear Maps

Shuichi Katsuamta * Shota Yamada †

September 2, 2016

Abstract

In this paper, we present new adaptively secure identity-based encryption (IBE) schemes. One of the distinguishing properties of the schemes is that it achieves shorter public parameters than previous schemes. Both of our schemes follow the general framework presented in the recent IBE scheme of Yamada (Eurocrypt 2016), employed with novel techniques tailored to meet the underlying algebraic structure to overcome the difficulties arising in our specific setting. Specifically, we obtain the following:

- Our first scheme is proven secure under the ring learning with errors (RLWE) assumption and achieves the best asymptotic space efficiency among existing schemes from the same assumption. The main technical contribution is in our new security proof that exploits the ring structure in a crucial way. Our technique allows us to greatly weaken the underlying hardness assumption (e.g., we assume the hardness of RLWE with a fixed polynomial approximation factor whereas Yamada's scheme requires a super-polynomial approximation factor) while improving the overall efficiency.

- Our second IBE scheme is constructed on bilinear maps and is secure under the 3-computational bilinear Diffie-Hellman exponent assumption. This is the first IBE scheme based on the hardness of a computational/search problem, rather than a decisional problem such as DDH and DLIN on bilinear maps with sub-linear public parameter size.

1 Introduction

Background. Identity-based encryption (IBE) is a generalization of public key encryption (PKE) where the public key of a user can be any arbitrary string such as an e-mail address. The concept of IBE was first proposed by Shamir [Sha85] in 1984, but it took nearly two decades for the first realizations of IBE [SOK00, BF01, Coc01] to appear. Since then, the construction of IBE has been one of the central topics in cryptography. Nowadays, we have constructions of IBEs from assumptions on bilinear maps [BF01, BB04a, BB04b, Wat05, Gen06, Wat09], the quadratic residue assumption [Coc01, BGH07], and from the learning with error (LWE) assumption [GPV08, CHKP10, ABB10] whose hardness is implied by the worst case reductions to certain lattice problems [Reg05].

*The University of Tokyo, National Institute of Advanced Industrial Science and Technology (AIST). E-mail: shuichi_katsumata@it.k.u-tokyo.ac.jp

†National Institute of Advanced Industrial Science and Technology (AIST). E-mail: yamada-shota@aist.go.jp

One of the most standard security definitions for IBE is the adaptive security, or often called full security. While it is not quite hard to obtain the adaptive security for an IBE in the random oracle model [BF01, Coc01, GPV08], the realization in the standard model is much harder. Roughly speaking, currently there are two general techniques in achieving adaptive security in the standard model: the partitioning technique [BB04b, Wat05] and the dual system encryption methodology [Wat09, LW10]. The latter is very attractive, because it allows us to construct very efficient IBE schemes [CW13, JR13] and even more advanced cryptosystems such as attribute-based encryptions [LOS+10] with adaptive security. However, it inherently relies on *decisional assumptions* on bilinear maps (e.g., SXDH and DLIN) and cannot be extended to the proofs based on *computational assumptions* on bilinear maps (e.g., computational bilinear Diffie-Hellman (CBDH) assumption) or assumptions on lattices. On the other hand, the application of the former technique is wider. We can construct adaptively secure IBE from the CBDH assumption (by the straightforward combination of the Goldreich-Levin bit [GL89] and Waters IBE [Wat05]) and from the LWE assumption [CHKP10, ABB10, Boy10]. However, IBE schemes constructed from the former approach typically requires larger parameters due to the use of the Waters' hash [Wat05] or the admissible hash [BB04b, CHKP10].

Very recently, Yamada [Yam16] constructed IBE schemes from lattices based on the partitioning technique with novel ideas that are different from the Waters' hash or the admissible hash. His schemes achieve asymptotically shorter public parameters than previous works. One of the drawbacks of the schemes is that they require super-polynomial size modulus for LWE. As a result, their ciphertexts are longer than those of previous works by a rather large super-constant factor. In addition, they have to assume the hardness of the LWE problem for *all polynomial* (i.e., $O(n^c)$ for *all* $c \in \mathbb{N}$) or the more aggressive *super-polynomial* approximation factor. Though their assumption is plausible, it is much stronger than those used in the previous works where the hardness of the LWE problem for some *fixed polynomial* approximation factor (i.e., $O(n^c)$ for *some* $c \in \mathbb{N}$) is assumed. Furthermore, since he used fully homomorphic computations of trapdoors [BGG+14], a technique unique to the lattice setting, it is a highly non-trivial task to construct analogous schemes in other settings such as bilinear maps.

Our Contribution. In this paper, we focus on the constructions of adaptively secure IBE in these settings where dual system encryption methodology is unavailable. In particular, we propose IBE schemes with shorter public parameters from ring/ideal lattices and from a certain computational assumption (rather than a decisional assumption) on bilinear groups, by extending and adding twists to the techniques of [Yam16]. Specifically, we obtain the following results. See Table 1 and 2 for the overview.

- We propose an anonymous and adaptively secure IBE scheme from the ring LWE (RLWE) assumption with *fixed polynomial* approximation factors, which is further reduced to certain worst case problems on ideal lattices. Note that simply instantiating Yamada's scheme using ideal lattices[1] will still require the RLWE assumption for *all polynomial* approximation factors, which is a much stronger assumption than what we use. As for the efficiency, the size of the public parameters, private keys, and ciphertexts in our scheme are $O(n\kappa^{1/d} \log n)$, $O(n \log n)$, and $O(n \log n)$, respectively. Here, n is the dimension of the ring elements, κ is the length of the identities, and d is a flexible constant that can be set arbitrary, but will affect the reduction cost exponentially. We note that all of them achieve the best efficiency among the other adaptively secure IBE from the RLWE assumption in an asymp-

[1] Note that he does not describe nor mention the ring variant of the scheme. However, we can convert his scheme into a ring variant in a straightforward manner as is the case in most previous works [CHKP10, ABB10, Boy10].

totic sense. Compared to the ring version of Yamada's scheme, we managed to reduce the poly-logarithmic factors contained in the public parameters, private keys, and ciphertexts.

- We propose a (non anonymous and) adaptively secure IBE scheme from the 3-computational bilinear Diffie-Hellman exponent (3-CBDHE) assumption. The 3-CBDHE assumption is a weaker variant of the n-decisional bilinear Diffie-Hellman exponent (n-DBDHE) assumption [BBG05, BGW05, BH08]. The former seems to be much a weaker assumption than the latter in two aspects. First, the former is a computational assumption whereas the latter is a decisional assumption. Second, the former is not a parameterized assumption, in the sense that the size of the problem instance only depends on the security parameter. As for the efficiency, the public parameters, private keys, and ciphertexts in our scheme require $O(\sqrt{\kappa})$ group elements. Here, κ is the length of the identities. This is the first adaptively secure IBE scheme from a computational assumption on bilinear groups with public parameters consisting of sub-linear number of group elements in the length of the identities. However, we note that the sizes of the ciphertexts and private keys of our scheme are larger than the previous schemes.

We emphasize that our result for the lattice based construction cannot be obtained through the simple switch to the ring setting in Yamada's scheme. Their proof will still require a super-polynomial-size modulus to work, whereas our new technique allows for a polynomial-size modulus. In addition, the security proof of our scheme requires new ideas that did not appear in [Yam16]. It exploits the commutative properties of the underlying ring elements in an essential way, involves a more generalized partitioning argument, and a careful analysis of the Gaussian error. Refer Sec. 2 for the technical overview. We note that the public parameter of our second scheme could be further reduced to $O(\kappa^{1/d})$ assuming the $d+1$-CBDHE assumption. However, it would come at the cost of even longer ciphertexts and complicated description of the scheme. This is beyond the scope of our work. We finally remark that the reduction costs for both of our schemes are inadmissible as was in the case of [Yam16]. In fact, the reduction loss for the first scheme is worse than [Yam16]. Improving them is left as an open problem.

Related Works. One way to reduce the size of the public parameters in Waters' hash and its analogue is to use Naccache's approach [Nac07, SRB12]. However, with this approach, we are only allowed to reduce the size of public parameters up to logarithmic factor. Ducas et al. [DLP14] constructed efficient IBE over NTRU lattices in the random oracle model. Gentry [Gen06] proposed adaptively secure IBE with compact parameters from a parameterized (or q-type) assumption on bilinear maps. Galindo [Gal10] and Chen et al. [CCZ11] proposed selectively secure CCA-secure IBE schemes from the CBDH assumption.

Note on Recent Works. Here, we mention two important recent related works.

Apon et al. [AFL16] proposed an adaptively secure IBE scheme from lattices whose parameters are very compact, using collision resistant hash function with output-length $\kappa = \omega(\log \lambda)$. Here, λ is the security parameter. While their scheme is more efficient than our scheme, we clarify that they implicitly assume exponential security on the collision resistant hash function, which is a stronger assumption than what we use. To demonstrate this, let us set $\kappa = \log^2 \lambda$. If there is no better attack than the birthday attack against the hash function, no PPT adversary can find a collision with more than negligible probability. On the other hand, the existence of even a sub-exponential time attack would compromise the security of the IBE. For example, assume that there exists an attack that finds a collision in time $2^{\sqrt{\kappa}}$. Then, the collision for the hash can be found in linear time in λ, since $2^{\sqrt{\kappa}} = 2^{\log \lambda} = \lambda$.

In their very recent work, Zhang et al. [ZCZ16] constructed an IBE scheme with poly-logarithmic public parameters. While their scheme achieves better asymptotic space efficiency than our scheme, their scheme is Q-bounded, in the sense that the security of the scheme is not guaranteed any more if the adversary obtains more than Q private keys. This restriction cannot be removed by just making Q super-polynomial, because the running time of the encryption algorithm in their scheme is at least linear in Q. We note that our scheme is secure against an unbounded collusion.

2 Overview of Our Techniques

2.1 Construction from Ring and Ideal Lattices

The Yamada IBE. We briefly review the Yamada IBE [Yam16], for our proposed IBE scheme follows the framework of theirs and overcomes some of the major problems posed by their construction. Their construction follows the general framework of constructing lattice-based IBE schemes that associates to each identity ID the matrix $[\mathbf{A}|\mathsf{H}(\mathsf{ID})] \in \mathbb{Z}_q^{n\times 2m}$. In previous IBE constructions [ABB10, CHKP10], the function $\mathsf{H}(\mathsf{ID})$ was computed by using the rather long κ public matrices $\{\mathbf{B}_i\}_{i\in[\kappa]}$, where $\kappa = O(n)$ is the length of the identities. The main technical contribution of the Yamada IBE was in reducing the size of the public matrices to $\kappa^{1/d}$ for any constant d and hence reducing the size of the public parameters by incorporating a primitive called fully homomorphic trapdoor functions. Hereafter, we consider the case $d = 2$ for simplicity. In detail, they used an injective map $S : \{0,1\}^\kappa \to 2^{[\ell]\times[\ell]}$ that maps an identity to a subset of the set $[\ell] \times [\ell]$ where $\ell = \lceil \kappa^{1/2} \rceil$, and computed the function $\mathsf{H}(\mathsf{ID})$ as

$$\mathsf{H}(\mathsf{ID}) = \mathbf{B}_0 + \sum_{(i,j)\in S(\mathsf{ID})} \mathbf{B}_{1,i} \cdot \mathbf{G}^{-1}(\mathbf{B}_{2,j}) \tag{1}$$

where the number of public matrices $\mathbf{B}_0, \{\mathbf{B}_{i,j}\}_{(i,j)\in[2]\times[\ell]}$ are now reduced to $O(\kappa^{1/2})$. Here, $\mathbf{G}$ is a special gadget matrix whose trapdoor is publicly known [MP12] and $\mathbf{G}^{-1}$ is viewed as a deterministic function rather than a matrix, that maps a matrix $\mathbf{V} \in \mathbb{Z}_q^{n\times m}$ to a matrix $\mathbf{U} \in \{0,1\}^{m\times m}$ such that $\mathbf{GU} = \mathbf{V} \mod q$.

During the security proof, the reduction algorithm first prepares random integers $y_0, \{y_{i,j}\}_{(i,j)\in[2]\times[\ell]} \in \mathbb{Z}_q$ from certain domains whose size grows linear in the number of key extraction query Q of the adversary. Then after sampling $\mathbf{R}_0, \{\mathbf{R}_{i,j}\}_{i\in[2],j\in[\ell]} \in \mathbb{Z}^{m\times m}$ with small spectral norm, the reduction algorithm prepares the public parameters as

$$\mathbf{B}_0 = \mathbf{AR}_0 + y_0\mathbf{G}, \quad \mathbf{B}_{i,j} = \mathbf{AR}_{i,j} + y_{i,j}\mathbf{G} \tag{2}$$

for $(i,j) \in [2] \times [\ell]$. Then during the security reduction the hash value for identity ID Eq.(1) is computed as

$$\begin{aligned}
\mathsf{H}(\mathsf{ID}) &= (\mathbf{AR}_0 + y_0\mathbf{G}) + \sum_{(i,j)\in S(\mathsf{ID})} (\mathbf{AR}_{1,i} + y_{1,i}\mathbf{G}) \cdot \mathbf{G}^{-1}(\mathbf{B}_{2,j}) \\
&= (\mathbf{AR}_0 + y_0\mathbf{G}) + \sum_{(i,j)\in S(\mathsf{ID})} (\mathbf{AR}_{1,i}\mathbf{G}^{-1}(\mathbf{B}_{2,j}) + y_{1,i}\mathbf{B}_{2,j}) \\
&= (\mathbf{AR}_0 + y_0\mathbf{G}) + \sum_{(i,j)\in S(\mathsf{ID})} \big(\mathbf{AR}_{1,i}\mathbf{G}^{-1}(\mathbf{B}_{2,j}) + y_{1,i}(\mathbf{AR}_{2,j} + y_{2,j}\mathbf{G})\big)
\end{aligned}$$

$$
\begin{aligned}
&= (\mathbf{A}\mathbf{R}_0 + y_0\mathbf{G}) + \sum_{(i,j)\in S(\mathsf{ID})} \left(\mathbf{A}\mathbf{R}_{1,i}\mathbf{G}^{-1}(\mathbf{B}_{2,j}) + \mathbf{A}(y_{1,i}\mathbf{R}_{2,j}) + y_{1,i}y_{2,j}\mathbf{G}\right) \\
&= \mathbf{A}\underbrace{\left(\mathbf{R}_0 + \sum_{(i,j)\in S(\mathsf{ID})} \left(\mathbf{R}_{1,i}\mathbf{G}^{-1}(\mathbf{B}_{2,j}) + y_{1,i}\mathbf{R}_{2,j}\right)\right)}_{:=\mathbf{R}_{\mathsf{ID}},\text{ which is ``small''}} + \underbrace{\left(y_0 + \sum_{(i,j)\in S(\mathsf{ID})} y_{1,i}y_{2,j}\right)}_{:=\mathsf{F}_{\mathbf{y}}(\mathsf{ID})}\cdot\mathbf{G} \\
&= \mathbf{A}\mathbf{R}_{\mathsf{ID}} + \mathsf{F}_{\mathbf{y}}(\mathsf{ID})\mathbf{G}. \qquad (3)
\end{aligned}
$$

Observe that we implicitly relied on the fact that $\mathbf{A}$ and $y_{1,i}$ commutes. Therefore, the reduction algorithm is able to sample a secret key for ID using the trapdoor of $\mathbf{G}$ if and only if $\mathsf{F}_{\mathbf{y}}(\mathsf{ID}) \neq 0 \mod q$. Hence, the simulation succeeds when the adversary queries on secret keys for ID satisfying $\mathsf{F}_{\mathbf{y}}(\mathsf{ID}) \neq 0 \mod q$, and queries for a challenge ciphertext for $\mathsf{ID}^\star$ satisfying $\mathsf{F}_{\mathbf{y}}(\mathsf{ID}^\star) = 0 \mod q$ in which case the reduction algorithm can embed its LWE challenge.

Overview of the Construction and Security Proof. The major drawback of the Yamada IBE is that they require the modulus size q to be super-polynomial. This stems from the fact that the size of $y_0, y_{i,j} \in \mathbb{Z}_q$ must grow linearly in the number of adversarial key extraction query Q for the security proof to be meaningful, i.e., $\Pr_{\mathbf{y}}[\mathsf{F}_{\mathbf{y}}(\mathsf{ID}^\star) = 0 \wedge \mathsf{F}_{\mathbf{y}}(\mathsf{ID}_1) \neq 0 \wedge \cdots \wedge \mathsf{F}_{\mathbf{y}}(\mathsf{ID}_Q) \neq 0]$ is noticeable in n. However, since the size of the $\mathbf{G}$-trapdoor $\mathbf{R}_{\mathsf{ID}}$ used during simulation grows proportionally to the size of $y_{1,i}$ (check above Eq.(3) to see how $\mathbf{R}_{\mathsf{ID}}$ was created), thereby growing proportional to $Q = \mathsf{poly}(n)$, we need to set the modulus size q to be at least super-polynomial in n for the trapdoor to operate properly. Therefore, if we try to restrict ourselves to a polynomial sized modulus q, it seems the best we can achieve is a scheme where we have to set a bound on the number of adversarial key extraction queries before instantiation, i.e., a Q-bounded scheme.

In our work, we combine several ideas in a novel way to circumvent the above seemingly inevitable problem. The first idea is to extend the elements $y_0, y_{i,j} \in \mathbb{Z}_q$ to matrices $\mathbf{Y}_0, \mathbf{Y}_{i,j} \in \mathbb{Z}_q^{n\times n}$ so that instead of increasing the size of the element $y \in \mathbb{Z}_q$, we can "pack" small elements in the entries of the matrix $\mathbf{Y} \in \mathbb{Z}_q^{n\times n}$. Namely, since the matrix has n^2 entries, if the number of key extraction query is $Q = n^c$ for some constant c, we can always set up the matrix so that c of the entries are packed by elements of size $O(n)$. Since there are n^2 entries in total, this allows us to pack the matrix with small entries (e.g., $O(n)$) for arbitrary $Q = \mathsf{poly}(n)$ without the need of increasing the modulus size q. However, this simple idea alone does not work, since during the security proof to obtain Eq.(3), we crucially relied on the fact that $\mathbf{A}$ and $y_{1,i}$ commutes. For our idea to work we need the two matrices $\mathbf{A}$ and $\mathbf{Y}_{1,i}$ to commute, however, in general this does not hold.

To overcome this problem, we introduce our second idea of using the ring structure of ideal lattices. Concretely, we use the special polynomial ring $R = \mathbb{Z}[X]/(X^n + 1)$ to construct our scheme for n a power of 2. The construction itself is exactly the same as the ring analogue of the Yamada IBE, however, our new security proof relies crucially on the underlying ring structure. In detail, the reduction algorithm prepares the public parameters as

$$
\boldsymbol{b}_0 = \boldsymbol{a}\boldsymbol{R}_0 + y_0\boldsymbol{g}, \quad \boldsymbol{b}_{i,j} = \boldsymbol{a}\boldsymbol{R}_{i,j} + y_{i,j}\boldsymbol{g} \qquad (4)
$$

for $(i,j) \in [2]\times[\ell]$, where $\boldsymbol{a}, \boldsymbol{b}_0, \boldsymbol{b}_{i,j} \in R_q^k$, $\boldsymbol{R} \in R_q^{k\times k}$, $y_0, y_{i,j} \in R_q$ and $\boldsymbol{g} \in R_q^k$ is the ring analogue of the $\mathbf{G}$-trapdoor. Observe that $y_0, y_{i,j}$ are now elements in R_q instead of $\mathbb{Z}_q$. Although this y is not quite a matrix, this is actually more than enough for us to use the packing technique described above. This can be seen by first noticing the natural isomorphism between $R_q \cong \mathbb{Z}_q^n$ induced by the coefficient embedding and viewing $y \in R_q$ as a vector in $\mathbb{Z}_q^n$. Since y has n entries when viewed

as vectors, it can support up to n^n queries by packing each entry with small elements of size $O(n)$. Furthermore, the second part of the problem addressed above is naturally resolved, since now that we are working in a ring we get the commutativity of $\boldsymbol{a}$ and $y_{1,j}$ for free. This key role in the commutativity for rings is somewhat reminiscent to the signature scheme of [DM14]. We note that the technique used by [Alp15] (which has also been used in [Xag13]) to extend the results of [DM14] to matrices seems to be inapplicable in our setting. This is because in our setting we need to commute the LWE challenge matrix $\mathbf{A}$ instead of the gadget matrix $\mathbf{G}$ whose associating trapdoor is known. To summarize, by incorporating our second idea, we obtain the ring variant of Eq.(3) and the trapdoor operates as specified. We note that one might be tempted to pack the entries of y with constant size elements, since 2^n is still exponential in n and hence $Q(n) < 2^n$. However, the security proof relies heavily on the fact that the density (i.e., the number of entries that are packed) of y is bounded by some constant. Therefore, we must choose the size of the packed elements with care to make the overall scheme secure.

The final idea is carefully crafting a properly distributed challenge ciphertext. To be precise, the main issue is in the difficulty of creating a ciphertext that has errors that are properly distributed. This problem of generating a properly distributed challenge ciphertext was addressed in [Yam16] as well, however, they used the standard technique called the "smudging" or "noise flooding" technique which came at the cost of making the modulus size q super-polynomial in n. This was not a problem for them, since as we pointed out earlier, their scheme inherently needed a super-polynomial sized modulus to work. However, this tactic is inapplicable to our setting since we want to restrict ourselves to the polynomial sized modulus. To overcome this we devise a way to carefully craft the error term; a technique reminiscent of [GPV08, ACPS09]. First, assume we have $\mathsf{F}(\mathsf{ID}^\star) = 0$ for the challenge identity $\mathsf{ID}^\star$ and thus $\mathsf{H}(\mathsf{ID}) = \mathbf{A}\mathbf{R}_{\mathsf{ID}^\star}$. Note that for ease of understanding we explain the technique in the matrix form instead of the ring form. To prove security, we have to embed the LWE challenge $\mathbf{A}$ and $\mathbf{v}$ into the challenge ciphertext, where $\mathbf{v} = \mathbf{s}\mathbf{A} + \mathbf{x}$ or $\mathbf{v}$ a random vector. One natural way is to set

$$\mathbf{x}_1 = \mathbf{x}, \quad \mathbf{x}_2 = \mathbf{x}\mathbf{R}_{\mathsf{ID}^\star} \tag{5}$$

and compute the challenge ciphertext as

$$\mathbf{s}[\mathbf{A}|\mathsf{H}(\mathsf{ID}^\star)] + [\mathbf{x}_1|\mathbf{x}_2] = [\mathbf{v}|\mathbf{v}\mathbf{R}_{\mathsf{ID}^\star}].$$

However, one can not simply use the standard generalized leftover hash lemma for lattices presented in [ABB10]; a technique often used in proving such forms. This is because $\mathbf{R}_{\mathsf{ID}^\star}$ is not uniformly sampled as in the case of [ABB10], but instead highly correlated to the values of $y, \{y_{i,j}\}$ used during the simulation. Alternatively, we present a noise rerandomization technique and add a small extra noise to Eq.(5) and statistically hide $\mathbf{R}_{\mathsf{ID}}$. Namely, we sample noises $\mathbf{e}_1$ and $\mathbf{e}_2$ from a particular Gaussian distribution with variance computed from $\mathbf{R}_{\mathsf{ID}^\star}$ and set

$$\mathbf{x}_1 = \mathbf{x} + \mathbf{e}_1, \quad \mathbf{x}_2 = \mathbf{x}\mathbf{R}_{\mathsf{ID}^\star} + \mathbf{e}_2. \tag{6}$$

Thus the challenge ciphertext is created as above by further adding the new noise terms. Although the general idea of this technique has been around since [Reg05, GPV08] and has been used in contexts elsewhere, as far as we know, we believe this is a nice application for rerandomizing the noise without the need of adding a huge (super-polynomial sized) noise.

An Additional Idea. Working in the ring setting introduces some subtle yet crucial obstacles, which we did not have to address before. Namely, for q a prime and n a power of 2, the domain $R_q = \mathbb{Z}[X]/(q, X^n + 1)$ we work in is no longer a field as in the case of $\mathbb{Z}_q$. Additionally, if we use

a modulus q such that $q \equiv 1 \mod 2n$ as in [LPR10, LPR13], the ring R_q completely splits into n fields. In such a ring, each field only contains $q = \mathsf{poly}(n)$ elements so the Schwartz-Zippel lemma during our security proof can not be applied. We get around this by using a modulus q such that $q \equiv 3 \mod 8$ where it is known to split into only two fields. Then, since each field now contains $q^{n/2}$ elements and R_q acts roughly as a field, we are able to apply our proof techniques. As for the purpose of completeness, we prove the hardness of LWE over such rings by the straightforward combination of previous results in the Appendix E. We finally note that we also obtain a nice regularity lemma over such rings which helps us attain better parameters for the scheme.

We also employ some ideas to further optimize the sizes of the public parameters, secret keys and ciphertexts. Namely, we use the (ring version of the) $\mathbf{G}$-trapdoor where the base is set as n^η for some positive constant η. We use $\eta = \frac{1}{4}$ for our concrete parameter selection. By incorporating this idea, we can further reduce the size of the parameters by a factor of $\log n$. However, this comes at the cost of making the scheme less efficient, since the function $\mathbf{G}^{-1}(\cdot)$ has a slower running time for a larger base.

2.2 Construction from Bilinear Maps

Here, we explain our IBE scheme from bilinear maps. We start with a slightly modified version of Waters IBE [Wat05] and gradually modify it to obtain our scheme. Let us consider a group $\mathbb{G}$ with prime order p whose generator is g. The group is equipped with a efficiently computable bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$. The public parameters of the scheme contains rather long $\kappa + 3$ group elements $\{g^{w_i}\}_{i\in[0,\kappa]}$, g^α, g^β, and a randomness $\mathsf{rand} \in \{0,1\}^{|\mathbb{G}_T|}$ that is used to derive the Goldreich-Levin hardcore bit function $\mathsf{GL} : \{0,1\}^{|\mathbb{G}_T|} \times \{0,1\}^{|\mathbb{G}_T|} \to \{0,1\}$. The form of the ciphertexts and private keys in the scheme are as follows:

$$C \;=\; \left(g^s,\ g^{s\mathsf{H}(\mathsf{ID})},\ \mathsf{GL}\left(e(g^\alpha, g^\beta)^s, \mathsf{rand}\right) \oplus \mathsf{M} \right), \quad \mathsf{sk}_{\mathsf{ID}} = \left(g^{\alpha\beta} \cdot g^{r\mathsf{H}(\mathsf{ID})},\ g^{-r} \right)$$

where $\mathsf{M} \in \{0,1\}$ is the message to be encrypted, and s and r are random elements in $\mathbb{Z}_p$ that are picked during the encryption and key generation algorithms, respectively.

Here, $\mathsf{H} : \{0,1\}^\kappa \to \mathbb{Z}_p$ is defined as $\mathsf{H}(\mathsf{ID}) = w_0 + \sum_{\mathsf{ID}_i=1} w_i$ where ID_i is the i-th bit of ID. The reason why we use the hardcore bit function is to base the security of the scheme on the *computational* bilinear Diffie-Hellman (CBDH) assumption, rather than the stronger *decisional* bilinear Diffie-Hellman (DBDH) assumption which was used to prove the security of the original Waters IBE.

Next, we try to reduce the size of the public parameters using the idea of the Yamada IBE. A natural way to do this would be to introduce the injective map $S : \{0,1\}^\kappa \to 2^{[\ell]\times[\ell]}$ with $\ell = \lceil \kappa^{1/2} \rceil$, change the public parameters to be $g^{w_0}, \{g^{w_{i,j}}\}_{(i,j)\in[2]\times[\ell]}$, and modify the function H as

$$\mathsf{H}(\mathsf{ID}) = w_0 + \sum_{(i,j)\in S(\mathsf{ID})} w_{1,i} w_{2,j}. \tag{7}$$

Through this change, we can reduce the size of the public parameters from $O(\kappa)$ group elements to $O(\sqrt{\kappa})$, just in as [Yam16]. However, we come across an immediate problem: We cannot efficiently compute $g^{s\mathsf{H}(\mathsf{ID})}$ from the public parameters! A straightforward solution to this problem is to put "helper" terms $\{g^{w_{1,i}w_{2,j}}\}$ into the public parameters. However, this makes the size of the public parameters large again.

Our solution to this problem is to rely on the Boneh-Boyen technique [BB04a] to compute something similar to the problematic term. Namely, we compute

$$g^{s\mathsf{H}(\mathsf{ID})+\sum_{j\in S(\mathsf{ID})}\tilde{t}_j w_{2,j}}, \qquad \{\ g^{\tilde{t}_j}\ \}_{j\in[\ell]} \tag{8}$$

instead of computing only $g^{s\mathsf{H}(\mathsf{ID})}$. Here, $\{\tilde{t}_j\}$ are additional randomness introduced by the encryption algorithm. Accordingly, we change the form of the ciphertexts and private keys of our scheme as follows:

$$\begin{aligned} C &= \left(g^s,\ g^{s\mathsf{H}(\mathsf{ID})+\sum_{j\in[\ell]}\tilde{t}_j w_{2,j}},\ \{g^{\tilde{t}_j}\}_{j\in[\ell]},\ \mathsf{GL}\left(e(g^\alpha, g^\beta)^s, \mathsf{rand}\right) \oplus \mathsf{M}\ \right), \\ \mathsf{sk}_{\mathsf{ID}} &= \left(g^{\alpha\beta}\cdot g^{r\mathsf{H}(\mathsf{ID})},\ g^{-r},\ \{g^{rw_{2,j}}\}_{j\in[\ell]}\ \right). \end{aligned} \tag{9}$$

Note that although the size of the public parameters is smaller than the original scheme, the sizes of the ciphertexts and private keys are larger due to the additional terms. We now show that one can efficiently compute the ciphertext. In particular, we show that it is possible to generate the terms in Eq.(8). To see this, let us introduce the variables $\{t_j\}$ such that

$$\tilde{t}_j := t_j - s\left(\sum_{i\in\{i\in[1,\ell]|(i,j)\in S(\mathsf{ID})\}} w_{1,i}\right). \tag{10}$$

Then, we have

$$\begin{aligned} & s\mathsf{H}(\mathsf{ID}) + \sum_{j\in[\ell]}\tilde{t}_j w_{2,j} \\ =\ & s\mathsf{H}(\mathsf{ID}) + \sum_{j\in[\ell]} w_{2,j}\left(t_j - s\left(\sum_{i\in\{i\in[1,\ell]|(i,j)\in S(\mathsf{ID})\}} w_{1,i}\right)\right) \\ =\ & s\mathsf{H}(\mathsf{ID}) + \sum_{j\in[\ell]} w_{2,j}t_j - s\sum_{j\in[\ell]}\left(\sum_{i\in\{i\in[1,\ell]|(i,j)\in S(\mathsf{ID})\}} w_{1,i}w_{2,j}\right) \\ =\ & sw_0 + s\cancel{\sum_{(i,j)\in S(\mathsf{ID})} w_{1,i}w_{2,j}} + \sum_{j\in[\ell]} w_{2,j}t_j - s\cancel{\sum_{(i,j)\in S(\mathsf{ID})} w_{1,i}w_{2,j}} \\ =\ & sw_0 + \sum_{j\in[\ell]} w_{2,j}t_j. \end{aligned} \tag{11}$$

Since Eq.(10) and (11) are linear in w_0, $w_{i,j}$, it can be seen that the terms in Eq.(8) can be computed efficiently, as desired.

By substituting $\tilde{t}_j$ in Eq.(9) with the right-hand side of Eq.(8), we obtain our final scheme. As for the security, we can prove the adaptive security of the scheme from the 3-computational bilinear Diffie-Hellman exponent (3-CBDHE) assumption. We need to rely on this stronger assumption than the standard CBDH assumption, because of the different algebraic structure incorporated by the modified Waters IBE.

3 Preliminaries

Notations. We use non-italic bold lowercase letters (e.g., $\mathbf{v}$) for vectors with entries in $\mathbb{R}$ and italic bold lowercase letters (e.g., $\boldsymbol{v}$) for vectors with entries in rings or number fields. We view vectors

in the row form stated otherwise. Matrices are denoted by uppercase bold letters analogously. For a vector $\mathbf{v} \in \mathbb{R}^n$, denote $\|\mathbf{v}\|_p$ as the L_p-norm, where $p = 2$ is the standard Euclidean norm. For a matrix $\mathbf{R} \in \mathbb{R}^{n \times n}$, denote $\|\mathbf{R}\|_{\mathsf{GS}}$ as the longest column of the Gram-Schmidt orthogonalization of $\mathbf{R}$ and denote $s_1(\mathbf{R})$ as the largest singular value (spectral norm). We denote $[\cdot|\cdot]$ (resp. $[\cdot;\cdot]$) as the horizontal (resp. vertical) concatenation of vectors and matrices. We denote $[a, b]$ as the set $\{a, a+1, \ldots, b-1, b\}$ for any integers $a, b \in \mathbb{N}$ satisfying $a \leq b$, and for simplicity write $[b]$ for the special case $a = 1$. For a (quotient) polynomial ring R over $\mathbb{Z}$, we denote $[-b, b]_R \subseteq R$ as the set of elements in R with all coefficients in the interval $[-b, b]$. Statistical distance between two random variables X and Y with support Ω is defined as $\Delta(X;Y) = \frac{1}{2}\sum_{s\in\Omega} |\Pr[X = s] - \Pr[Y = s]|$. A function $f : \mathbb{N} \to \mathbb{R}_{\geq 0}$ is said to be negligible, if for all c, there exists λ_0 such that $f(\lambda) < 1/\lambda^c$ for all $\lambda > \lambda_0$. We denote by $\mathsf{negl}(\lambda)$ a negligible function in λ.

3.1 Identity-Based Encryption

Syntax. We use the standard syntax of IBE [BF01]. Let $\mathcal{ID}$ be the ID space of the scheme. If a collision resistant hash function $CRH : \{0,1\}^* \to \mathcal{ID}$ is available, one can use an arbitrary string as an identity. An IBE scheme is defined by the following four algorithms.

$\mathsf{Setup}(1^\lambda) \to (\mathsf{mpk}, \mathsf{msk})$: The setup algorithm takes as input a security parameter 1^λ and outputs a master public key mpk and a master secret key msk.

$\mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, \mathsf{ID}) \to \mathsf{sk}_{\mathsf{ID}}$: The key generation algorithm takes as input the master public key mpk, the master secret key msk, and an identity $\mathsf{ID} \in \mathcal{ID}$. It outputs a private key $\mathsf{sk}_{\mathsf{ID}}$. We assume that ID is implicitly included in $\mathsf{sk}_{\mathsf{ID}}$.

$\mathsf{Encrypt}(\mathsf{mpk}, \mathsf{ID}, \mathsf{M}) \to C$: The encryption algorithm takes as input a master public key mpk, an identity $\mathsf{ID} \in \mathcal{ID}$, and a message M, It outputs a ciphertext C.

$\mathsf{Decrypt}(\mathsf{mpk}, \mathsf{sk}_{\mathsf{ID}}, C) \to \mathsf{M}$ or $\perp$: The decryption algorithm takes as input the master public key mpk, a private key $\mathsf{sk}_{\mathsf{ID}}$, and a ciphertext C. It outputs the message M or $\perp$, which means that the ciphertext is not in a valid form.

Correctness. We require correctness of decryption: that is, for all λ, all $\mathsf{ID} \in \mathcal{ID}$, and all M in the specified message space,

$$\Pr[\mathsf{Decrypt}(\mathsf{mpk}, \mathsf{sk}_{\mathsf{ID}}, \mathsf{Encrypt}(\mathsf{mpk}, \mathsf{ID}, \mathsf{M})) = \mathsf{M}] = 1 - \mathsf{negl}(\lambda)$$

holds, where the probability is taken over the randomness used in $(\mathsf{mpk}, \mathsf{msk}) \xleftarrow{\$} \mathsf{Setup}(1^\lambda)$, $\mathsf{sk}_{\mathsf{ID}} \xleftarrow{\$} \mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, \mathsf{ID})$, and $\mathsf{Encrypt}(\mathsf{mpk}, \mathsf{ID}, \mathsf{M})$.

Security. We now define the security for an IBE scheme Π. This security notion is defined by the following game between a challenger and an adversary $\mathcal{A}$.

- Setup. At the outset of the game, the challenger runs $\mathsf{Setup}(1^\lambda) \to (\mathsf{mpk}, \mathsf{msk})$ and gives mpk to $\mathcal{A}$.

- Phase 1. $\mathcal{A}$ may adaptively make key-extraction queries. If $\mathcal{A}$ submits $\mathsf{ID} \in \mathcal{ID}$ to the challenger, the challenger returns $\mathsf{sk}_{\mathsf{ID}} \leftarrow \mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, \mathsf{ID})$.

- Challenge Phase. At some point, $\mathcal{A}$ outputs a message M and an identity $\mathsf{ID}^\star \in \mathcal{ID}$, on which it wishes to be challenged. Then, the challenger picks a random coin $\mathsf{coin} \xleftarrow{\$} \{0,1\}$ and a random ciphertext $C \xleftarrow{\$} \mathcal{C}$ from the ciphertext space. If $\mathsf{coin} = 0$, it runs $\mathsf{Encrypt}(\mathsf{mpk}, \mathsf{ID}^\star, \mathsf{M}) \to C^\star$ and gives the challenge ciphertext $C^\star$ to $\mathcal{A}$. If $\mathsf{coin} = 1$, it sets the challenge ciphertext as $C^\star = C$ and gives it to $\mathcal{A}$.

- Phase 2. After the challenge query, $\mathcal{A}$ may continue to make key-extraction queries, with the added restriction that $\mathsf{ID} \neq \mathsf{ID}^\star$.

- Guess. Finally, $\mathcal{A}$ outputs guess a $\widehat{\mathsf{coin}}$ for coin. The advantage of $\mathcal{A}$ is defined as

$$\mathsf{Adv}^{\mathsf{IBE}}_{\mathcal{A},\Pi} = \left| \Pr[\widehat{\mathsf{coin}} = \mathsf{coin}] - \frac{1}{2} \right|.$$

We say that Π is adaptively-anonymous secure, if the advantage of any PPT $\mathcal{A}$ is negligible. The term anonymous captures the fact that the ciphertext does not reveal the identity for which it was sent to.

We also define the standard adaptive security (without anonymity) as in [Wat05] for Π via a similar game to the above. To define adaptive security, we change the challenge phase as follows.

- Challenge Phase. $\mathcal{A}$ outputs two messages M_0, M_1 and an identity $\mathsf{ID}^\star \in \mathcal{ID}$, on which it wishes to be challenged. Then, the challenger picks a random coin $\mathsf{coin} \xleftarrow{\$} \{0,1\}$, runs $\mathsf{Encrypt}(\mathsf{mpk}, \mathsf{ID}^\star, \mathsf{M}_{\mathsf{coin}}) \to C^\star$, and gives the challenge ciphertext $C^\star$ to $\mathcal{A}$.

We also say that Π is adaptively secure, if the advantage of any PPT $\mathcal{A}$ is negligible. We note that adaptively-anonymous security implies adaptive security. Namely, the former is a stronger security notion.

3.2 Lattices and Gaussian Distributions

An n-dimensional (full rank) lattice $\Lambda \subseteq \mathbb{R}^n$ is the set of all integer linear combinations of some set of n linearly independent basis vectors $\mathbf{B} = \{\mathbf{b}_1, \ldots, \mathbf{b}_n\} \subseteq \mathbb{R}^n$, $\Lambda = \{\sum_{i\in[n]} z_i \mathbf{b}_i | \mathbf{z} \in \mathbb{Z}^n\}$. For positive integers q, n, m, a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and a vector $\mathbf{u} \in \mathbb{Z}_q^n$, the m-dimensional "shifted" integer lattice is defined as $\Lambda^\perp_{\mathbf{u}}(\mathbf{A}) = \{\mathbf{z} \in \mathbb{Z}^m | \mathbf{A}\mathbf{z}^T = \mathbf{u}^T \mod q\}$. We simply write $\Lambda^\perp(\mathbf{A})$ in case $\mathbf{u} = \mathbf{0}$.

For $s > 0$, the n-dimensional Gaussian function $\rho_s : \mathbb{R}^n \to (0,1]$ is defined as $\rho_s(\mathbf{x}) = \exp(-\pi\|\mathbf{x}\|_2^2/s^2)$. The (spherical) continuous Gaussian distribution D_s over $\mathbb{R}^n$ is the distribution with density function proportional to ρ_s. When the dimension n is not clear from context, we explicitly write it as D_s^n. More generally, for any matrix $\mathbf{B} \in \mathbb{R}^{n\times m}$, denote $D_{\mathbf{B}}$ as the distribution of $\mathbf{x}\mathbf{B}^T$ where $\mathbf{x}$ is distributed as D_1^m. A well known fact is that for any two matrices $\mathbf{B}_1, \mathbf{B}_2$, the sum of an independent sample from $D_{\mathbf{B}_1}$ and $D_{\mathbf{B}_2}$ is distributed as $D_{\mathbf{C}}$ where $\mathbf{C} = (\mathbf{B}_1\mathbf{B}_1^T + \mathbf{B}_2\mathbf{B}_2^T)^{1/2}$.

For a n-dimensional lattice Λ and a vector in $\mathbf{u} \in \mathbb{R}^n$, the discrete Gaussian distribution $D_{\Lambda+\mathbf{u},s}$ over the coset $\Lambda + \mathbf{u}$ is defined as $D_{\Lambda+\mathbf{u},s}(\mathbf{x}) = \rho_s(\mathbf{x})/\rho_s(\Lambda+\mathbf{u})$ for all $\mathbf{x} \in \Lambda + \mathbf{u}$. We also define the discrete Gaussian distribution $D^{\mathsf{coeff}}_{\Lambda+\mathbf{u},r}$ over a (quotient) polynomial ring R in X over $\mathbb{R}$. The discrete Gaussian distribution $D^{\mathsf{coeff}}_{\Lambda+\mathbf{u},r}$ is the distribution of $a = \sum_{i=0}^{n-1} \alpha_i X^i \in R$ where the coefficient vector $[\alpha_0, \ldots, \alpha_{n-1}] \in \mathbb{R}^n$ is sampled from the discrete Gaussian distribution $D_{\Lambda+\mathbf{u},r}$. This definition naturally extends to vectors $\boldsymbol{a} \in R^k$ in case of nk-dimensional lattices.

The following lemma on noise rerandomization plays an important role in the security proof of our scheme when creating a properly distributed challenge ciphertext. This allows us to simulate the challenge ciphertext without resorting to the noise flooding technique as in [Yam16]. The proof can be found in Appendix B.1.

Lemma 1 (Noise Rerandomization)**.** *Let q, ℓ, m be positive integers and r a positive real satisfying $r > \max\{\omega(\sqrt{\log m}), \omega(\sqrt{\log \ell})\}$. Let $\mathbf{b} \in \mathbb{Z}_q^m$ be arbitrary and $\mathbf{x}$ chosen from $D_{\mathbb{Z}^m,r}$. Then for any $\mathbf{V} \in \mathbb{Z}^{m\times\ell}$ and positive real $\sigma > s_1(\mathbf{V})$, there exists a PPT algorithm $\mathsf{ReRand}(\mathbf{V}, \mathbf{b}+\mathbf{x}, r, \sigma)$ that outputs $\mathbf{b}' = \mathbf{b}\mathbf{V} + \mathbf{x}' \in \mathbb{Z}_q^\ell$ where $\mathbf{x}'$ is distributed statistically close to $D_{\mathbb{Z}^\ell, 2r\sigma}$.*

Remark 1. *During the security proof we set* $\ell = 2m$, $\mathbf{V} = [\mathbf{I}_m | \mathbf{R}_{\mathsf{ID}}]$ *and* $\mathbf{b} + \mathbf{x}$ *as the LWE challenge. Note that we deal with the LWE challenge in a slightly different manner than usual by viewing the LWE challenge as either* $\mathbf{b} = \mathbf{sA}$ *or* $\mathbf{b}$ *a uniformly random element. This is only a conceptual difference. Then, for a valid LWE tuple, the output returned by the noise rerandomization algorithm* ReRand *is* $\mathbf{b}' = \mathbf{bV} + \mathbf{x}' = \mathbf{s}[\mathbf{A}|\mathbf{AR}_{\mathsf{ID}}] + \mathbf{x}'$, *where* $\mathbf{x}'$ *is distributed according to a spherical Gaussian distribution and* $\mathbf{b}'$ *is distributed exactly as in the real world. We also like to emphasize that the* ReRand *algorithm only needs to know the value* $\mathbf{b} + \mathbf{x}$ *and does not need to know the individual values* $\mathbf{b}$ *and* $\mathbf{x}$.

3.3 Rings and Ideal Lattices

We try to provide a minimum exposition of rings and ideal lattices to keep it self-contained. For further detail see Appendix E or refer to other works [LPR10, LPR13].

Preparation. Let n be a power of 2 and set $m = 2n$. Define the ring $R = \mathbb{Z}[X]/(\Phi_m(X))$, where $\Phi_m(X) = X^n + 1$ is the mth cyclotomic polynomial. For an integer q, denote R_q as $R/qR = \mathbb{Z}[X]/(q, \Phi_m(X))$. By viewing the elements in R as $n-1$ degree polynomials in $\mathbb{Z}[X]$, we can consider a natural coefficient embedding of R onto the integer lattice $\mathbb{Z}^n$. Namely, we define the coefficient embedding $\phi : R \to \mathbb{Z}^n$ that maps $a = \sum_{i=0}^{n-1} \alpha_i X^i \in R$ to $[\alpha_0, \alpha_1, \ldots, \alpha_{n-1}] \in \mathbb{Z}^n$. We extend the coefficient embedding naturally to vectors and matrices. On the other hand, we can also identify R as the subring of anti-circulant matrices in $\mathbb{Z}^{n \times n}$ by viewing each ring element $a \in R$ as a linear transformation $r \to a \cdot r$ of R. Concretely, we define the ring homomorphism $\mathrm{rot} : R \to \mathbb{Z}^{n \times n}$ that sends $a \in R$ to a matrix in $\mathbb{Z}^{n \times n}$ such that the i-th row is $\phi(a \cdot X^{i-1} \bmod \Phi_m(X)) \in \mathbb{Z}^n$. Note that the first row of $\mathrm{rot}(a)$ is $\phi(a)$. Similarly to above, the definition of the map rot naturally extends to vectors and matrices. We provide some useful formulas on ring elements in the Appendix A.

Norms in R. We define the Euclidean length for an element $a \in R$ and a vector $\boldsymbol{v} \in R^k$ by identifying R with $\mathbb{Z}^n$ through the coefficient embedding.[2] Therefore, when we say a vector $\boldsymbol{v}$ in R^k is "short", we mean that $\|\phi(\boldsymbol{v})\|_2$ is small. We also define the largest singular value of a matrix $\boldsymbol{R} \in R^{s \times t}$ by identifying the ring R with $\mathbb{Z}^{n \times n}$ through the map rot.[3] Namely, $s_1(\boldsymbol{R}) := \max_{\|\mathbf{z}\|_2 = 1} \|\mathbf{z} \cdot \mathrm{rot}(\boldsymbol{R})\|_2$. Note that this definition allows us to consider singular values of an element in R as well.

Properties for Elements in R. As with matrices with entries in $\mathbb{R}$, we have similar singular value bounds for matrices with elements in R. Namely, we can bound the singular value of a random matrix chosen from $[-b, b]_R^{s \times t}$. Recall that an element of $[-b, b]_R$ is an element in R with all of its coefficients in the interval $[-b, b]$.

Lemma 2 ([DM15], Special case of Fact 1)**.** *Let b be a positive integer and $\boldsymbol{R}$ be a $s \times t$ matrix chosen uniformly at random from $[-b, b]_R^{s \times t}$. Then, there exists a universal constant $C (\approx 1/\sqrt{2\pi})$ such that*

$$\Pr[s_1(\boldsymbol{R}) \geq C \cdot b\sqrt{n} \cdot (\sqrt{s} + \sqrt{t} + \omega(\sqrt{\log n}))] = \mathsf{negl}(n)$$

[2] We could have identified the Euclidean length by the *canonical* embedding as done in other works. However, for our special case where n is power of 2, the lengths are equivalent up to a factor of $\sqrt{n}$. (See Appendix E.2 for further detail.)

[3] For the special case where n is a power of 2, $s_1(\boldsymbol{R})$ defined by the coefficient and canonical embeddings are both equivalent to the one defined by the map rot. (See Appendix E.2 for further detail.)

We note that similarly to matrices with entries in $\mathbb{R}$, we have $s_1(\boldsymbol{R}_1\boldsymbol{R}_2) \le s_1(\boldsymbol{R}_1)s_1(\boldsymbol{R}_2)$ for all $\boldsymbol{R}_1, \boldsymbol{R}_2 \in R^{k\times k}$, which follows from the fact that rot is a ring homomorphism. Furthermore, it also holds when $\boldsymbol{R}_1$ is replaced by an element a in R.

Regularity Lemma. The former Lemma shows that there exists a quotient ring $R_q = R/(q, \Phi_m(X))$ that acts roughly as a field, or in other words, R_q has exponentially many invertible elements. The latter Lemma is a ring analogue of the standard lattice regularity lemma. The proof of the following Lemmas can be found in Appendix B.2 and B.3.

Lemma 3. *Let q be a prime such that $q \equiv 3 \mod 8$ and n be a power of 2. Then, $\Phi_{2n}(X) = X^n+1$ splits as $X^n + 1 \equiv t_1 t_2 \mod q$ for two irreducible polynomials $t_1 = X^{n/2} + uX^{n/4} - 1$ and $t_2 = X^{n/2} - uX^{n/4} - 1$ in $\mathbb{Z}_q[X]$ where $u^2 \equiv -2 \mod q$. Furthermore, all $x \in R_q$ satisfying $\|\phi(x)\|_2 < \sqrt{q}$ are invertible, i.e., $x \in R_q^*$.*

Lemma 4 (Regularity Lemma)**.** *Let n be a power of 2, q be a prime larger than $4n$ such that $q \equiv 3 \mod 8$, and ℓ, k', k, ρ be positive integers satisfying $\ell, k' \ge 1$, $k \ge 2$, $\rho < \frac{1}{2}\sqrt{q/n}$. Define the family of hash functions $\mathcal{H} = \{h_{\boldsymbol{A}}(\boldsymbol{x}) : [-\rho, \rho]_R^k \to R_q^{k'}\}$, where $h_{\boldsymbol{A}}(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x}$ for $\boldsymbol{A} \in R_q^{k'\times k}$, $\boldsymbol{x} \in R_q^{k\times 1}$. Then, $\mathcal{H}$ is a universal hash family. Furthermore, for $\boldsymbol{A} \xleftarrow{\$} R_q^{k'\times k}$ and $\boldsymbol{X} \xleftarrow{\$} [-\rho, \rho]_R^{k\times \ell}$, we have*

$$\Delta((\boldsymbol{A}, \boldsymbol{A}\boldsymbol{X}) \ ; \ (\boldsymbol{A}, U(R_q^{k'\times \ell}))) \le \frac{\ell}{2} \cdot \sqrt{\left(\frac{q^{k'}}{(2\rho+1)^k}\right)^n}.$$

Ring Learning with Errors. The ring LWE problem was introduced by Lyubashevsky et al. [LPR10]. They showed that solving it on the average is as hard as (quantumly) solving several standard problems on ideal lattices in the worst case.

Definition 1 (RLWE)**.** *For positive integers $n = n(\lambda)$, $k = k(n)$, a prime integer $q = q(n) > 2$, an error distribution $\chi = \chi(n)$ over R_q, and an PPT algorithm $\mathcal{A}$, an advantage for the RLWE problem $\mathsf{RLWE}_{n,k,q,\chi}$ of $\mathcal{A}$ is defined as follows:*

$$\mathsf{Adv}_{\mathcal{A}}^{\mathsf{RLWE}_{n,k,q,\chi}} = |\Pr[\mathcal{A}(\{(a_i, v_i)\}_{i=1}^k) \to 1] - \Pr[\mathcal{A}(\{(a_i, a_i s + e_i)\}_{i=1}^k) \to 1]|$$

where $a_1, \ldots, a_k, v_1, \ldots, v_k, s \xleftarrow{\$} R_q$ and $e_1, \ldots, e_k \xleftarrow{\$} \chi$. We say that $\mathsf{RLWE}_{n,k,q,\chi}$ assumption holds if $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{RLWE}_{n,k,q,\chi}}$ is negligible for all PPT $\mathcal{A}$.

Theorem 1. *Let α be a positive real, m be a power of 2, ℓ be an integer, $\Phi_m(X) = X^n + 1$ be the mth cyclotomic polynomial where $m = 2n$, and $R = \mathbb{Z}[X]/(\Phi_m(X))$. Let $q \equiv 3 \mod 8$ be a (polynomial size) prime such that there is another prime $p \equiv 1 \mod m$ satisfying $p \le q \le 2p$. Let also $\alpha q \ge n^{3/2}k^{1/4}\omega(\log^{9/4} n)$. Then, there is a probabilistic polynomial-time quantum reduction from $\tilde{O}(\sqrt{n}/\alpha)$-approximate SIVP (or SVP) to $\mathsf{RLWE}_{n,k,q,\chi}$ with $\chi = D_{\mathbb{Z}^n,\alpha q}^{\mathsf{coeff}}$.*

Although the proof is obtained by combining many of the previous results, we nevertheless include the proof in Appendix E for completeness. Due to the Linnik's theorem and Dirichlet's theorem on arithmetic progressions, we have that there are sufficiently many primes p and q satisfying the assumption of the theorem.

Trapdoors for Rings. Define the gadget matrix $\boldsymbol{g}_b = [1|b|\cdots|b^{k'-1}|\boldsymbol{0}] \in R_q^k$, where b is a positive integer and $k \ge k' = \lceil \log_b q \rceil$. When $k = k'$ and $b = 2$, this corresponds to the matrix representation of the gadget matrix $\mathbf{G} \in \mathbb{Z}_q^{n\times nk}$ often used in the literatures by properly rearranging

the rows and columns of $\mathrm{rot}(\boldsymbol{g}_2)$. The following algorithms are simple modification of traditional lattice based algorithms, however, owing to the conversion to the ring setting and the fact that we view vectors in their row form, it may seem unclear at first. We provide some supplementary notes in Appendix A and B.4.

Lemma 5. *Let n be a power of 2, q be a prime larger than $4n$ such that $q \equiv 3 \mod 8$, and b, ρ be a positive integer satisfying $\rho < \frac{1}{2}\sqrt{q/n}$. Furthermore, define $\log_1(\cdot) := \log_2(\cdot)$. Then, there exist polynomial time algorithms with the properties below:*

- $\mathsf{TrapGen}(1^n, 1^k, q, \rho) \to (\boldsymbol{a}, \boldsymbol{T_a})$ ([MP12], Lemma 5.3): *a randomized algorithm that, when $k \geq 2\log_\rho q$, outputs a vector $\boldsymbol{a} \in R_q^k$ and a matrix $\boldsymbol{T_a} \in R^{k\times k}$, where $\mathrm{rot}(\boldsymbol{a}^T)^T \in \mathbb{Z}_q^{n\times nk}$ is a full-rank matrix and $\mathrm{rot}(\boldsymbol{T_a}) \in \mathbb{Z}^{nk\times nk}$ is a basis for $\Lambda^\perp(\mathrm{rot}(\boldsymbol{a}^T)^T)$ such that $\boldsymbol{a}$ is $\mathsf{negl}(n)$-close to uniform and $\|\mathrm{rot}(\boldsymbol{T_a})\|_{\mathsf{GS}} = O(b\rho \cdot \sqrt{n\log_\rho q})$.*[4]
- $\mathsf{SampleLeft}(\boldsymbol{a}, \boldsymbol{b}, u, \boldsymbol{T_a}, \sigma) \to \boldsymbol{e}$ ([CHKP10]): *a randomized algorithm that, given vectors $\boldsymbol{a}, \boldsymbol{b} \in R_q^k$ where $\mathrm{rot}(\boldsymbol{a}^T)^T, \mathrm{rot}(\boldsymbol{b}^T)^T \in \mathbb{Z}_q^{n\times nk}$ are full-rank, an element $u \in R_q$, a matrix $\boldsymbol{T_a} \in R^{k\times k}$ such that $\mathrm{rot}(\boldsymbol{T_a}) \in \mathbb{Z}^{nk\times nk}$ is a basis for $\Lambda^\perp(\mathrm{rot}(\boldsymbol{a}^T)^T)$, and a Gaussian parameter $\sigma > \|\mathrm{rot}(\boldsymbol{T_a})\|_{\mathsf{GS}} \cdot \omega(\sqrt{\log nk})$, outputs a vector $\boldsymbol{e} \in R^{2k}$ sampled from a distribution which is $\mathsf{negl}(n)$-close to $D^{\mathsf{coeff}}_{\Lambda^\perp_{\phi(u)}([\mathrm{rot}(\boldsymbol{a}^T)^T|\mathrm{rot}(\boldsymbol{b}^T)^T]),\sigma}$, i.e., $[\boldsymbol{a}|\boldsymbol{b}]\boldsymbol{e}^T = u$ and $\phi(\boldsymbol{e}) \in \mathbb{Z}^{2nk}$ is distributed according to $D_{\Lambda^\perp_{\phi(u)}([\mathrm{rot}(\boldsymbol{a}^T)^T|\mathrm{rot}(\boldsymbol{b}^T)^T]),\sigma}$.*
- $\mathsf{SampleRight}(\boldsymbol{a}, \boldsymbol{g}_b, \boldsymbol{R}, y, u, \boldsymbol{T}_{\boldsymbol{g}_b}, \sigma) \to \boldsymbol{e}$ *where* $\boldsymbol{b} = \boldsymbol{a}\boldsymbol{R} + y\boldsymbol{g}_b$ ([ABB10]): *a randomized algorithm that, given vectors $\boldsymbol{a}, \boldsymbol{g}_b \in R_q^k$ such that $\mathrm{rot}(\boldsymbol{a}^T)^T, \mathrm{rot}(\boldsymbol{g}_b)$*[5] *$\in \mathbb{Z}_q^{n\times nk}$ are full-rank matrices, elements $y \in R_q^*, u \in R_q$, a matrix $\boldsymbol{R} \in R^{k\times k}$, a matrix $\boldsymbol{T}_{\boldsymbol{g}_b} \in R^{k\times k}$ such that $\mathrm{rot}(\boldsymbol{T}_{\boldsymbol{g}_b}) \in \mathbb{Z}^{nk\times nk}$ is a basis for $\Lambda^\perp(\mathrm{rot}(\boldsymbol{g}_b))$, and a Gaussian parameter $\sigma > s_1(\boldsymbol{R}) \cdot \|\mathrm{rot}(\boldsymbol{T}_{\boldsymbol{g}_b})\|_{\mathsf{GS}} \cdot \omega(\sqrt{\log nk})$, outputs a vector $\boldsymbol{e} \in R^{2k}$ sampled from a distribution which is $\mathsf{negl}(n)$-close to $D^{\mathsf{coeff}}_{\Lambda^\perp_{\phi(u)}([\mathrm{rot}(\boldsymbol{a}^T)^T|\mathrm{rot}(\boldsymbol{b}^T)^T]),\sigma}$, i.e., $[\boldsymbol{a}|\boldsymbol{b}]\boldsymbol{e}^T = u$ and $\phi(\boldsymbol{e}) \in \mathbb{Z}^{2nk}$ is distributed according to $D_{\Lambda^\perp_{\phi(u)}([\mathrm{rot}(\boldsymbol{a}^T)^T|\mathrm{rot}(\boldsymbol{b}^T)^T]),\sigma}$.*
- ([MP12]:) *Let $k \geq \lceil \log_b q \rceil$. There exists a publicly known matrix $\boldsymbol{T}_{\boldsymbol{g}_b}$ such that $\mathrm{rot}(\boldsymbol{T}_{\boldsymbol{g}_b}) \in \mathbb{Z}^{nk\times nk}$ is a basis for the lattice $\Lambda^\perp(\mathrm{rot}(\boldsymbol{g}_b))$ and $\|\mathrm{rot}(\boldsymbol{T}_{\boldsymbol{g}_b})\|_{\mathsf{GS}} \leq \sqrt{b^2+1}$. Furthermore, there exists a deterministic polynomial time algorithm $\boldsymbol{g}_b^{-1}$ which takes input $\boldsymbol{u} \in R_q^k$ and outputs $\boldsymbol{R} = \boldsymbol{g}_b^{-1}(\boldsymbol{u})$ such that $\boldsymbol{R} \in [-b, b]_R^{k\times k}$ and $\boldsymbol{g}_b\boldsymbol{R} = \boldsymbol{u}$.*

Note that we abuse the notation $\boldsymbol{g}_b^{-1}$ by viewing it as a function rather than a vector. Namely, for any $\boldsymbol{u} \in R_q^k$ there are many choices for $\boldsymbol{R} \in R^{k\times k}$ such that $\boldsymbol{g}_b\boldsymbol{R} = \boldsymbol{u}$, and $\boldsymbol{g}_b^{-1}(\boldsymbol{u})$ is a function that deterministically outputs a particular short matrix from the possible candidates. Since we have $s_1(\boldsymbol{R}) \leq b \cdot nk$ for any $\boldsymbol{R} \in [-b, b]_R^{k\times k}$, $s_1(\boldsymbol{g}_b^{-1}(\boldsymbol{u})) \leq bnk$ holds for arbitrary $\boldsymbol{u} \in R_q^k$.

Homomorphic Computation. Let d be a natural number. We introduce the function $\mathsf{PubEval}_d : (R_q^k)^d \to R_q^k$ as in [Yam16], which takes a set of vectors $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_d \in R_q^k$ as inputs and outputs a vector in R_q^k. This function wil be used to hash identities to R_q^k in our lattice-based IBE construction. The function is defined recursively as follows:

$$\mathsf{PubEval}_d(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d) = \begin{cases} \boldsymbol{b}_1 & \text{if } d = 1 \\ \boldsymbol{b}_1 \cdot \boldsymbol{g}_b^{-1}\big(\mathsf{PubEval}_{d-1}(\boldsymbol{b}_2, \ldots, \boldsymbol{b}_d)\big) & \text{if } d \geq 2. \end{cases}$$

[4] We combine several lemmas from [MP12] and the regularity lemma (Lemma 4) to show correctness of $\mathsf{TrapGen}$. See Appendix B.4 for further detail. Further, the unusual lattice $\Lambda^\perp(\mathrm{rot}(\boldsymbol{a}^T)^T)$ is used only to be consistent with the other algorithms. Namely, we could have instead defined the trapdoor for the lattice $\Lambda^\perp(\mathrm{rot}(\boldsymbol{a}))$.

[5] We have $\mathrm{rot}(\boldsymbol{g}_b^T)^T = \mathrm{rot}(\boldsymbol{g}_b)$ since all the entries of $\boldsymbol{g}_b$ are integers.

The proof of the following lemma can be found in Appendix B.5.

Lemma 6. *Let $y_1, \ldots, y_d$ be elements in R, $\boldsymbol{a}, \boldsymbol{b}_1, \ldots, \boldsymbol{b}_d$ be vectors in R_q^k and $\boldsymbol{R}_1, \ldots, \boldsymbol{R}_d$ be matrices in $R^{k\times k}$ such that $\boldsymbol{b}_i = \boldsymbol{a}\boldsymbol{R}_i + y_i\boldsymbol{g}_b$ for $i \in [d]$. Furthermore, we assume that $s_1(\boldsymbol{R}_i) \leq B, \|\phi(y_i)\|_1 \leq \delta$ for $i \in [d]$. Then, there exists an efficient algorithm $\mathsf{TrapEval}_d$ that takes $\boldsymbol{R}_1, \ldots, \boldsymbol{R}_d$, $y_1, \ldots, y_d$ as inputs and outputs $\boldsymbol{R}' \in R^{k\times k}$ such that*

$$\mathsf{PubEval}_d(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_d) = \boldsymbol{a}\boldsymbol{R}' + y_1 \cdots y_d \boldsymbol{g}_b \in R_q^k \tag{12}$$

and $s_1(\boldsymbol{R}') \leq B\delta^{d-1} + Bbnk\left(\frac{\delta^{d-1}-1}{\delta-1}\right)$.

3.4 Other Facts

The proof of the following lemmas will appear in Appendix B.6 and B.7.

Lemma 7 (Expansion of Coefficients)**.** *Let $c_1, c_2, B_1, B_2 \in \mathbb{N}$. Let also $u = u_0 + u_1 X + \cdots u_{c_1-1}X^{c_1-1} \in R$ and $v = v_0 + v_1 X + \cdots v_{c_2-1}X^{c_2-1} \in R$ be ring elements. We further assume that $c_1 + c_2 < n$ and $\|\phi(u)\|_\infty < B_1$ and $\|\phi(v)\|_\infty < B_2$. Then we have $\|\phi(uv)\|_\infty \leq \min\{c_1, c_2\} \cdot B_1 B_2$.*

The following Lemma addresses a general statement for bounding the success probability of an adversary engaging with the security game of IBE. In more detail, when the partitioning technique is used to prove security, the guess returned by the adversary is correlated with the key extraction queries it has made. Therefore, we need to argue with care to obtain a meaningful bound on the success probability that holds for arbitrary key extraction queries.

Lemma 8 (Implicit in [BR09, Yam16])**.** *Let us consider an IBE scheme and an adversary $\mathcal{A}$ that breaks adaptive security (adaptively-anonymous security) with advantage ϵ. Let us also consider a map γ that maps a sequence of identities to a value in $[0, 1]$. We consider the following experiment. We first execute the security game for $\mathcal{A}$. Let $\mathsf{ID}^\star$ be the challenge identity and $\mathsf{ID}_1, \ldots, \mathsf{ID}_Q$ be the identities for which key extraction queries were made. We denote $\mathbb{ID} = (\mathsf{ID}^\star, \mathsf{ID}_1, \ldots, \mathsf{ID}_Q)$. At the end of the game, we set $\mathsf{coin}' \in \{0, 1\}$ as $\mathsf{coin}' = \widehat{\mathsf{coin}}$ with probability $\gamma(\mathbb{ID})$ and $\mathsf{coin}' \xleftarrow{\$} \{0, 1\}$ with probability $1 - \gamma(\mathbb{ID})$. Then, the following holds.*

$$\left|\Pr[\mathsf{coin}' = \mathsf{coin}] - \frac{1}{2}\right| \geq \gamma_{\min} \cdot \epsilon - \frac{\gamma_{\max} - \gamma_{\min}}{2}$$

where $\gamma_{\min}$ (resp. $\gamma_{\max}$) is the maximum (resp. minimum) of $\gamma(\mathbb{ID})$ taken over all possible $\mathbb{ID}$.

Injective map. Let d and κ be some integers. Furthermore, let ℓ be $\ell = \lceil \kappa^{1/d} \rceil$. Then, an element of $[1, \kappa]$ can be written as an element of $[1, \ell]^d$ using some canonical map. Furthermore, it is also possible to write a subset of $[1, \kappa]$ as a subset of $[1, \ell]^d$ by naturally extending the canonical map. By identifying a bit string in $\{0, 1\}^\kappa$ with a subset of $[1, \kappa]$ (for example, by regarding the former as the indicator vector of a subset of $[1, \kappa]$), we can define an efficiently computable injective map S that maps a bit string $\mathsf{ID} \in \{0, 1\}^\kappa$ to a subset $S(\mathsf{ID})$ of $[1, \ell]^d$.

3.5 Core Lemma for Our Partitioning

We make a general statement concerning the partitioning technique for IBEs, which we use during the security analysis for both our lattice and bilinear map based constructions. Namely, we use the following Lemma in order to argue that the probability of the hash value for identities corresponding to the key extraction queries being invertible and the hash value for the challenge identity being zero is non-negligible.

Lemma 9. *Let $\nu, \mu, d, Q \geq 1$ be any integers. Let Φ be a ring and $\Omega_1, \ldots, \Omega_\nu$ be a set of fields equipped with homomorphisms $\pi_j : \Phi \to \Omega_j$ for $j \in [\nu]$. Assume that the map Π defined as $\Pi : \Phi \ni y \mapsto (\pi_1(y), \ldots, \pi_\nu(y)) \in \Omega_1 \times \cdots \times \Omega_\nu$ is an isomorphism. Let S_0 and S_1 be subsets of Φ with finite cardinality. Let us consider a set of multivariate polynomials $f_i(Y_1, \ldots, Y_\mu) \in \Phi[Y_1, \ldots, Y_\mu]$ for $i \in [0, Q]$ We further assume the following properties:*

1. *The map π_j is injective on S_1 for all $j \in [\nu]$.*
2. *We have $\pi_j(f_0) - \pi_j(f_i)$ is a non-zero polynomial with degree d for all $i \in [Q]$ and $j \in [\nu]$. Here π_j is extended to $\pi_j : \Phi[X] \to \Omega_j[X]$ in a natural way.*
3. *We have $S_0 \supseteq \cup_{i \in [0,Q]} \{-f_i(y_1, \ldots, y_\mu) | y_1, \ldots, y_\mu \in S_1\}$.*

Then, for $y_0 \xleftarrow{\$} S_0$ and $y_1, \ldots, y_\mu \xleftarrow{\$} S_1$, we have

$$\frac{1}{|S_0|}\left(1 - \frac{d\nu Q}{|S_1|}\right) \leq \Pr_{y_0, \boldsymbol{y}'}[\ y_0 + f_0(\boldsymbol{y}') = 0 \ \wedge \ y_0 + f_1(\boldsymbol{y}') \in \Phi^* \ \wedge \cdots \wedge \ y_0 + f_Q(\boldsymbol{y}') \in \Phi^*] \leq \frac{1}{|S_0|}$$

where we denote $\boldsymbol{y}' = (y_1, \ldots, y_\mu)$ and $\Phi^* = \Pi^{-1}(\Omega_1^* \times \cdots \times \Omega_\nu^*)$.

Proof. Let us denote $\gamma := \Pr_{y_0, \boldsymbol{y}'}[\ y_0 + f_0(\boldsymbol{y}') = 0 \ \wedge \ y_0 + f_1(\boldsymbol{y}') \in \Phi^* \ \wedge \cdots \wedge \ y_0 + f_Q(\boldsymbol{y}') \in \Phi^*]$ where the probability is taken over $y_0 \xleftarrow{\$} S_0$ and $y_1, \ldots, y_\mu \xleftarrow{\$} S_1$. We first show the upper bound. We have

$$\gamma \leq \Pr_{y_0, \boldsymbol{y}'}[y_0 + f_0(\boldsymbol{y}') = 0] = \Pr_{y_0, \boldsymbol{y}'}[y_0 = -f_0(\boldsymbol{y}')] = \frac{1}{|S_0|}.$$

The last equation follows since there exists unique $y_0 \in S_0$ such that $y_0 = -f_0(\boldsymbol{y}')$, for any $\boldsymbol{y}' \in S_1^\mu$ from our third assumption. We then proceed to show the lower bound. We have

$$\begin{aligned}
\gamma &= \Pr_{y_0, \boldsymbol{y}'}[\ y_0 + f_0(\boldsymbol{y}') = 0 \ \wedge \ y_0 + f_1(\boldsymbol{y}') \in \Phi^* \ \wedge \cdots \wedge \ y_0 + f_Q(\boldsymbol{y}') \in \Phi^*] \\
&= \Pr_{y_0, \boldsymbol{y}'}[\ y_0 + f_0(\boldsymbol{y}') = 0\ ] \\
&\quad - \Pr_{y_0, \boldsymbol{y}'}[y_0 + f_0(\boldsymbol{y}') = 0 \wedge \neg\left(y_0 + f_1(\boldsymbol{y}') \in \Phi^* \wedge \cdots \wedge y_0 + f_Q(\boldsymbol{y}') \in \Phi^*\right)] && (13) \\
&= \Pr_{y_0, \boldsymbol{y}'}[y_0 + f_0(\boldsymbol{y}') = 0] - \Pr_{y_0, \boldsymbol{y}'}[\bigvee_{i=1}^{Q}\left(y_0 + f_0(\boldsymbol{y}') = 0 \wedge y_0 + f_i(\boldsymbol{y}') \notin \Phi^*\right)] && (14) \\
&\geq \Pr_{y_0, \boldsymbol{y}'}[y_0 + f_0(\boldsymbol{y}') = 0] - \sum_{i \in [Q]} \Pr_{y_0, \boldsymbol{y}'}[\ y_0 + f_0(\boldsymbol{y}') = 0 \ \wedge \ y_0 + f_i(\boldsymbol{y}') \notin \Phi^*\ ] && (15) \\
&= \frac{1}{|S_0|} - \sum_{i \in [Q]} \underbrace{\Pr_{y_0, \boldsymbol{y}'}[\ y_0 + f_0(\boldsymbol{y}') = 0 \ \wedge \ y_0 + f_i(\boldsymbol{y}') \notin \Phi^*\ ]}_{:= \gamma_i'} && (16)
\end{aligned}$$

where Eq.(13) is a general equation that holds for any event, Eq.(14) follows from De morgan's laws and the distributive property, Eq.(15) follows from the union bound, Eq.(16) is again from our third assumption. We then have to show an upper bound for γ_i'.

$$\begin{aligned}
\gamma_i' &= \Pr_{y_0, \boldsymbol{y}'}[\ y_0 + f_0(\boldsymbol{y}') = 0 \ \wedge \ y_0 + f_i(\boldsymbol{y}') \notin \Phi^*\ ] \\
&= \Pr_{y_0, \boldsymbol{y}'}[\ y_0 + f_0(\boldsymbol{y}') = 0 \ \wedge \ f_0(\boldsymbol{y}') - f_i(\boldsymbol{y}') \notin \Phi^*\ ] && (17)
\end{aligned}$$

$$
\begin{aligned}
&= \Pr_{y_0,\boldsymbol{y}'}[\, y_0 = -f_0(\boldsymbol{y}') \mid f_0(\boldsymbol{y}') - f_i(\boldsymbol{y}') \notin \Phi^* \,] \cdot \Pr_{y_0,\boldsymbol{y}'}[\, f_0(\boldsymbol{y}') - f_i(\boldsymbol{y}') \notin \Phi^* \,] && (18)\\
&= \frac{1}{|S_0|} \cdot \Pr_{y_0,\boldsymbol{y}'}[\, f_0(\boldsymbol{y}') - f_i(\boldsymbol{y}') \notin \Phi^* \,] && (19)\\
&= \frac{1}{|S_0|} \cdot \underbrace{\Pr_{\boldsymbol{y}'}[\, f_0(\boldsymbol{y}') - f_i(\boldsymbol{y}') \notin \Phi^* \,]}_{:=\gamma_i''} && (20)
\end{aligned}
$$

where Eq.(17) is just an equivalent expression, Eq.(18) is trivial, Eq.(19) is from the fact that for any $\boldsymbol{y}' \in S_1^\mu$ there exists unique $y_0 \in S_0$ such that $y_0 = -f_0(\boldsymbol{y}')$ (from our third assumption), and in Eq.(20) we omit y_0 since it is independent of the probability. It suffices to show an upper bound for γ_i''. We have

$$
\begin{aligned}
\gamma_i'' &= \Pr_{\boldsymbol{y}' \xleftarrow{\$} S_1^\mu}\left[\, f_0(\boldsymbol{y}') - f_i(\boldsymbol{y}') \notin \Phi^* \,\right] && (21)\\
&= \Pr_{\boldsymbol{y}' \xleftarrow{\$} S_1^\mu}\left[\bigvee_{j=1}^{\nu} \Pi(f_0(\boldsymbol{y}') - f_i(\boldsymbol{y}')) \in \Omega_1 \times \cdots \times \Omega_{j-1} \times \{0\} \times \Omega_{j+1} \times \cdots \times \Omega_\nu\right] && (22)\\
&\le \sum_{j=1}^{\nu} \Pr_{\boldsymbol{y}' \xleftarrow{\$} S_1^\mu}[\Pi(f_0(\boldsymbol{y}') - f_i(\boldsymbol{y}')) \in \Omega_1 \times \cdots \times \Omega_{j-1} \times \{0\} \times \Omega_{j+1} \times \cdots \times \Omega_\nu] && (23)\\
&= \sum_{j=1}^{\nu} \Pr_{\boldsymbol{y}' \xleftarrow{\$} S_1^\mu}\left[\pi_j(f_0(\boldsymbol{y}') - f_i(\boldsymbol{y}')) = 0\right] && (24)\\
&= \sum_{j=1}^{\nu} \Pr_{\boldsymbol{y}' \xleftarrow{\$} S_1^\mu}\left[\big(\pi_j(f_0 - f_i)\big)(\pi_j(\boldsymbol{y}')) = 0\right] && (25)\\
&= \sum_{j=1}^{\nu} \Pr_{\boldsymbol{z} \xleftarrow{\$} \pi_j(S_1)^\mu}\left[\big(\pi_j(f_0 - f_i)\big)(\boldsymbol{z}) = 0\right] && (26)\\
&\le \sum_{j=1}^{\nu} \frac{d}{|\pi_j(S_1)|} && (27)\\
&= \frac{d\nu}{|S_1|} && (28)
\end{aligned}
$$

where in Eq.(21) we made the distribution of $\boldsymbol{y}'$ explicit, Eq.(22) is from the fact that $\Phi \backslash \Phi^* = \Pi^{-1}\left(\cup_{j=1}^{\nu}(\Omega_1 \times \cdots \times \Omega_{j-1} \times \{0\} \times \Omega_{j+1} \times \cdots \times \Omega_\nu)\right)$, Eq.(23) follows from the union bound, Eq.(24) is by the definition of Π, Eq.(25) follows since π_j is a homomorphism, Eq.(26) follows since π_j is injective on S_1 (our first assumption), Eq.(27) is from the fact that $\pi_j(f_0 - f_i) \in \Omega_j[X]$ is a non-zero polynomial with degree d (our second assumption) and the Schwartz-Zippel lemma, and Eq.(28) follows since π_j is injective on S_1.

□

4 Construction from RLWE

In this section, we show our IBE scheme from the RLWE assumption. Let d be a (flexible) constant number. In addition, let the identity space of the scheme be $\mathcal{ID} = \{0,1\}^\kappa$ for some $\kappa \in \mathbb{N}$ and

the message space be $\{0,1\}^n \subset R$.[6] For our construction, we consider an efficiently computable injective map S that maps an identity $\mathsf{ID} \in \{0,1\}^\kappa$ to a subset $S(\mathsf{ID})$ of $[1,\ell]^d$, where $\ell = \lceil \kappa^{1/d} \rceil$. Such a map can be constructed easily as we explained in Sec. 3.4. Let $n := n(\lambda)$, $b := b(n)$, $\rho := \rho(n)$, $m := 2n$, $k := k(n)$, $q := q(n)$, $\ell := \ell(n)$, $\alpha := \alpha(n)$, $\alpha' := \alpha'(n)$, and $\sigma := \sigma(n)$ be parameters that are specified later. Let also $\Phi_m(X) = X^n + 1$ be the mth cyclotomic polynomial and $R = \mathbb{Z}[X]/(\Phi_m(X))$.

$\mathsf{Setup}(1^\lambda)$: On input 1^λ, it first runs $(\boldsymbol{a}, \boldsymbol{T}_{\boldsymbol{a}}) \xleftarrow{\$} \mathsf{TrapGen}(1^n, 1^k, q, \rho)$ to obtain $\boldsymbol{a} \in R_q^k$ and $\boldsymbol{T}_{\boldsymbol{a}} \in R^{k \times k}$. It also picks $u \xleftarrow{\$} R_q$, $\boldsymbol{b}_0, \boldsymbol{b}_{i,j} \xleftarrow{\$} R_q^k$ for $(i,j) \in [d] \times [\ell]$ and outputs

$$\mathsf{mpk} = (\boldsymbol{a}, \boldsymbol{b}_0, \{\boldsymbol{b}_{i,j}\}_{(i,j)\in[d]\times[\ell]}, u) \quad \text{and} \quad \mathsf{msk} = \boldsymbol{T}_{\boldsymbol{a}}.$$

In the following, we use a deterministic function $\mathsf{H} : \mathcal{ID} \to R_q^k$ defined as

$$\mathsf{H}(\mathsf{ID}) = \boldsymbol{b}_0 + \sum_{(j_1,\ldots,j_d)\in S(\mathsf{ID})} \mathsf{PubEval}_d(\boldsymbol{b}_{1,j_1}, \boldsymbol{b}_{2,j_2}, \ldots, \boldsymbol{b}_{d,j_d}) \in R_q^k.$$

$\mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, \mathsf{ID})$: It first computes $\mathsf{H}(\mathsf{ID})$ and picks $\boldsymbol{e} \in R^{2k}$ such that

$$[\boldsymbol{a}|\mathsf{H}(\mathsf{ID})] \cdot \boldsymbol{e}^T = u$$

using $\mathsf{SampleLeft}(\boldsymbol{a}, \mathsf{H}(\mathsf{ID}), u, \mathbf{T}_{\boldsymbol{a}}, \sigma) \to \boldsymbol{e}$. It returns $\mathsf{sk}_{\mathsf{ID}} = \boldsymbol{e}$.

$\mathsf{Encrypt}(\mathsf{mpk}, \mathsf{ID}, \mathsf{M})$: To encrypt a message $\mathsf{M} \in \{0,1\}^n \subset R$, it first picks $s \xleftarrow{\$} R_q$, $x_0 \xleftarrow{\$} D^{\mathsf{coeff}}_{\mathbb{Z}^n,\alpha q}$, $\boldsymbol{x}_1, \boldsymbol{x}_2 \xleftarrow{\$} \left(D^{\mathsf{coeff}}_{\mathbb{Z}^n,\alpha'}\right)^k$. Then it computes

$$c_0 = su + x_0 + \lfloor q/2 \rceil \cdot \mathsf{M}, \quad \boldsymbol{c}_1 = s[\boldsymbol{a}|\mathsf{H}(\mathsf{ID})] + [\boldsymbol{x}_1|\boldsymbol{x}_2].$$

Finally, it outputs the ciphertext $C = (c_0, \boldsymbol{c}_1) \in R_q \times R_q^{2k}$.

$\mathsf{Decrypt}(\mathsf{mpk}, \mathsf{sk}_{\mathsf{ID}}, C)$: To decrypt a ciphertext $C = (c_0, \boldsymbol{c}_1)$ using a private key $\mathsf{sk}_{\mathsf{ID}} = \boldsymbol{e}$, it computes

$$\left(\lfloor (2/q) \cdot \phi(c_0 - \boldsymbol{c}_1 \boldsymbol{e}^T) \rceil \mod 2\right) = m. \tag{29}$$

Here, the rounding function $\lfloor \cdot \rceil$ is applied componentwise.

4.1 Correctness and Parameter Selection.

The proof of the following lemma can be found in Appendix C.

Lemma 10 (Correctness). *Assume $\alpha q \omega(\sqrt{\log n}) + \sqrt{nk}\alpha'\sigma\omega(\sqrt{\log nk}) \le q/5$ holds with over whelming probability. Then the above scheme has negligible decryption error.*

Parameter selection. To satisfy the correctness requirement and make the security proof follow through, we need the following:

- the error term is less than $q/5$ with overwhelming probability (i.e., $\alpha q \omega(\sqrt{\log n}) + \sqrt{nk}\alpha'\sigma\omega(\sqrt{\log nk})$. See Lemma 10,),

[6] Note that we regard m as an elements in R via $\phi^{-1} : \mathbb{Z}^n \to R$ (the inversion of coefficient embedding).

– $\mathsf{TrapGen}$ can operate (i.e., $\rho < \frac{1}{2}\sqrt{q/n}$ and $k \geq 2\log_\rho q$. See Lemma 5.),

– the gadget matrix $\boldsymbol{g}_b$ can be defined (i.e., $k \geq \lceil \log_b q \rceil$. See Lemma 5.),

– the regularity lemma (Lemma 4) can be applied in the security proof (i.e., $\frac{k}{2}\left(\frac{q^2}{(2\rho+1)^k}\right)^{\frac{n}{2}} = \mathsf{negl}(n)$.),

– σ is sufficiently large so that $\mathsf{SampleLeft}$ and $\mathsf{SampleRight}$ work (i.e., $\sigma > O(b\rho \cdot \sqrt{n\log_\rho q}) \cdot \omega(\sqrt{\log nk})$ and $\sigma > s_1(\boldsymbol{R})\sqrt{b^2+1}\cdot\omega(\sqrt{\log n})$, where $s_1(\boldsymbol{R}) \leq C''\cdot\kappa\rho\sqrt{n}(\sqrt{k}+\omega(\sqrt{\log n}))\left((cn)^{d-1}+bnk\frac{(cn)^{d-1}-1}{cn-1}\right)$ for some absolute constant C''. See Eq.(37). The latter condition turns out to be more restrictive.),

– ReRand algorithm in the security proof works (i.e., $\alpha' > 2\alpha q(s_1(\boldsymbol{R})+1)$, $\alpha q > \omega(\sqrt{\log nk})$ where $s_1(\boldsymbol{R})$ is the same as the one defined above. See Lemma 1.),

– the worst case to average case reduction works (i.e., $\alpha q \geq n^{3/2}k^{1/4}\omega(\log^{9/4} n)$. See Theorem 1.).

Recall that d is a (flexible) constant which may be set very small (e.g., $d = 2$ or 3) in a typical setting, and $\kappa(n) = n$ is the size of the identity space ID. To satisfy the above requirements, we propose two candidate parameter selections as follows:

Type 1 IBE. For this construction we set $b = 2$ and $\rho = 1$ in order to reduce the modulus size q. Recalling that we defined $\log_1 q := \log_2 q$, we can set the parameters as follows:

$$k = 4(d+1)\log n, \qquad q = n^{2d+2}, \qquad b = 2, \qquad \rho = 1,$$
$$\sigma = n^{d-\frac{1}{2}}\cdot\omega(\log n), \qquad \alpha = n^{-2d+\frac{1}{2}}\cdot\omega(\log^{\frac{9}{2}} n)^{-1}, \qquad \alpha' = n^{d+2\eta+2}\cdot\omega(\log^3 n)^{-1}.$$

We denote this specific instantiation as the Type 1 IBE scheme.

Type 2 IBE. For this construction we set $b = \rho = n^\eta$ for an arbitrary positive real η in order to reduce the size of the public parameters, private keys, and ciphertexts. Namely, one way to set the parameters is as follows:

$$k = 4 + \frac{2d+2}{\eta}, \qquad q = n^{2d+2+4\eta}, \qquad b = \rho = n^\eta,$$
$$\sigma = n^{d+2\eta-\frac{1}{2}}\cdot\omega(\log n), \qquad \alpha = n^{-2d-\frac{7}{2}\eta+\frac{1}{2}}\cdot\omega(\log^2 n)^{-1}, \qquad \alpha' = n^{d+2\eta+2}\cdot\omega(\log^{\frac{3}{4}} n)^{-1}.$$

By plugging in $\eta = \frac{1}{4}$ we obtain the following concrete parameter selection:

$$k = 8d+12, \qquad q = n^{2d+3}, \qquad b = \rho = n^{\frac{1}{4}},$$
$$\sigma = n^d\cdot\omega(\log n), \qquad \alpha = n^{-2d-\frac{3}{8}}\cdot\omega(\log^2 n)^{-1}, \qquad \alpha' = n^{d+\frac{5}{2}}\cdot\omega(\log^{\frac{3}{4}} n)^{-1}.$$

We denote this specific instantiation as the Type 2 IBE scheme.

4.2 Security Proof for the Scheme

The following theorem addresses the security of the scheme. The proof proceeds in a similar manner as in [Yam16], but we incorporate several novel ideas as we explained in Sec. 2.

Theorem 2. *The above IBE scheme is adaptively-anonymous secure assuming* $\mathsf{RLWE}_{n,k+1,q,D^{\mathsf{coeff}}_{\mathbb{Z}^n,\alpha q}}$ *is hard, where the ciphertext space is* $\mathcal{C} = R_q \times R_q^{2k}$.

Proof. Let $\mathcal{A}$ be a PPT adversary that breaks the adaptively-anonymous security of the scheme. In addition, let $\epsilon = \epsilon(n)$ and $Q = Q(n)$ be its advantage and the upper bound of the number of key extraction queries, respectively.

Since $\mathcal{A}$ is PPT and λ and n are polynomially related (namely, $n = O(\lambda^\delta)$ for some constant δ), there exists a constant number $c_1 \in \mathbb{N}$ such that $4(dQ+1) \leq n^{c_1}$ for all n that are sufficiently large. Similarly, since $\mathcal{A}$ breaks the security of the scheme, there exists $c_2 \in \mathbb{N}$ such that $2\epsilon \geq n^{-c_2}$ holds for infinitely many n. By setting $c = c_1 + c_2$, we have that

$$4dQ \leq n^c \text{ for all } n \in \mathbb{N} \quad \text{and} \quad \frac{\epsilon}{2(dQ+1)} \geq \frac{1}{n^c} \quad \text{for infinitely many } n \in \mathbb{N}. \tag{30}$$

In the proof, we will assume $d(c-1) < n$. Since both c and d are constant numbers, this holds for sufficiently large n.

We show the security of the scheme via the following games. In each game, a value $\mathsf{coin}' \in \{0,1\}$ is defined. While it is set $\mathsf{coin}' = \widehat{\mathsf{coin}}$ in the first game, these values might be different in the later games. In the following, we define X_i to be the event that $\mathsf{coin}' = \mathsf{coin}$.

Game_0 : This is the real security game. Recall that since the ciphertext space is $\mathcal{C} = R_q \times R_q^{2k}$, in the challenge phase, the challenge ciphertext is set as $C^\star = (c_0, \boldsymbol{c}_1) \xleftarrow{\$} R_q \times R_q^{2k}$ if $\mathsf{coin} = 1$. At the end of the game, $\mathcal{A}$ outputs a guess $\widehat{\mathsf{coin}}$ for coin. Finally, the challenger sets $\mathsf{coin}' = \widehat{\mathsf{coin}}$. By definition, we have

$$\left|\Pr[X_0] - \frac{1}{2}\right| = \left|\Pr[\mathsf{coin}' = \mathsf{coin}] - \frac{1}{2}\right| = \left|\Pr[\widehat{\mathsf{coin}} = \mathsf{coin}] - \frac{1}{2}\right| = \epsilon.$$

Game_1 : For integers $t_0, t_1 \in \mathbb{Z}$ such that $t_0 \leq t_1$ and positive integer $c \in \mathbb{N}$, let us denote $[t_0, t_1]_{R,c}$ as

$$[t_0, t_1]_{R,c} := \left\{ \sum_{i=0}^{c-1} a_i X^i \;\middle|\; a_i \in [t_0, t_1] \text{ for all } i \in [0, c-1] \right\} \subseteq R. \tag{31}$$

In words, $[t_0, t_1]_{R,c}$ denotes the set of polynomials of degree less then $c-1$ with all of its coefficients in the interval $[t_0, t_1]$. Note that c is the constant defined in Eq.(30). In this game, we change Game_0 so that the challenger performs the following additional step at the end of the game. First, the challenger picks $\boldsymbol{y} = (y_0, \{y_{i,j}\}_{(i,j)\in[d,\ell]})$ as

$$y_0 \xleftarrow{\$} [-\kappa(cn)^d, -1]_{R,(c-1)d+1} \qquad \text{and} \qquad y_{i,j} \xleftarrow{\$} [1, n]_{R,c} \tag{32}$$

for $(i,j) \in [d] \times [\ell]$. Recall κ is the length of the identities. We then define a function $\mathsf{F}_{\boldsymbol{y}} : \mathcal{ID} \to R_q$ as follows:

$$\mathsf{F}_{\boldsymbol{y}}(\mathsf{ID}) = y_0 + \sum_{(j_1,\ldots,j_d)\in S(\mathsf{ID})} y_{1,j_1} \cdots y_{d,j_d}.$$

Then the challenger checks whether the following condition holds:

$$\mathsf{F}_{\boldsymbol{y}}(\mathsf{ID}^\star) = 0 \;\wedge\; \mathsf{F}_{\boldsymbol{y}}(\mathsf{ID}_1) \in R_q^* \;\wedge\; \cdots \;\wedge\; \mathsf{F}_{\boldsymbol{y}}(\mathsf{ID}_Q) \in R_q^*, \tag{33}$$

where $\mathsf{ID}^\star$ is the challenge identity, and $\mathsf{ID}_1, \ldots, \mathsf{ID}_Q$ are identities for which $\mathcal{A}$ has made key extraction queries. If it does not hold, the challenger ignores the output $\widehat{\mathsf{coin}}$ of $\mathcal{A}$, and sets $\mathsf{coin}' \xleftarrow{\$} \{0,1\}$. In this case, we say that the challenger aborts. If condition (33) holds, the challenger sets $\mathsf{coin}' = \widehat{\mathsf{coin}}$. As we will show in Lemma 11, we have

$$\left|\Pr[X_1] - \frac{1}{2}\right| \geq \frac{1}{(\kappa c^d n^d)^{(c-1)d+1}} \left(\frac{\epsilon}{2} - \frac{dQ}{n^c}\right).$$

So as not to interrupt the proof of Theorem 2, we intentionally skip the proof for the time being.

Game_2 : In this game, we change the way $\boldsymbol{b}_0$ and $\boldsymbol{b}_{i,j}$ are chosen. At the beginning of the game, the challenger picks $\boldsymbol{R}_0, \boldsymbol{R}_{i,j} \xleftarrow{\$} [-\rho, \rho]_R^{k\times k}$ for $(i,j) \in [d] \times [\ell]$. It also picks $\boldsymbol{y}$ as in Game_1. Then, $\boldsymbol{a}$, $\boldsymbol{b}_0$, and $\boldsymbol{b}_{i,j}$ are defined as

$$\boldsymbol{b}_0 = \boldsymbol{a}\boldsymbol{R}_0 + y_0\boldsymbol{g}_b, \qquad \boldsymbol{b}_{i,j} = \boldsymbol{a}\boldsymbol{R}_{i,j} + y_{i,j}\boldsymbol{g}_b, \tag{34}$$

for $(i,j) \in [d] \times [\ell]$. The rest of the game is the same as in Game_1.

Now, we bound $|\Pr[X_2] - \Pr[X_1]|$. By Lemma 4, the distributions

$$\left(\boldsymbol{a}, \boldsymbol{a}\boldsymbol{R}_0 + y_0\boldsymbol{g}_b, \{\boldsymbol{a}\boldsymbol{R}_{i,j} + y_{i,j}\boldsymbol{g}_b\}_{(i,j)\in[d]\times[\ell]} \right) \text{ and } \left(\boldsymbol{a}, \boldsymbol{b}_0, \{\boldsymbol{b}_{i,j}\}_{(i,j)\in[d]\times[\ell]} \right)$$

are $\mathsf{negl}(n)$-close, where $\boldsymbol{b}_0, \boldsymbol{b}_{i,j} \xleftarrow{\$} R_q^k$. Therefore, we have $|\Pr[X_1] - \Pr[X_2]| = \mathsf{negl}(n)$.

Game_3 Recall that in the previous game, the challenger aborts at the end of the game if condition (33) is not satisfied. In this game, we change the game so that the challenger aborts as soon as the abort condition becomes true. Since this is only a conceptual change, we have $\Pr[X_2] = \Pr[X_3]$.

Before describing the next game, we define $\boldsymbol{R}_{\mathsf{ID}} \in R^{k\times k}$ for an identity $\mathsf{ID} \in \mathcal{ID}$ as

$$\boldsymbol{R}_{\mathsf{ID}} = \boldsymbol{R}_0 + \sum_{(j_1,\ldots,j_d)\in S(\mathsf{ID})} \mathsf{TrapEval}_d(\boldsymbol{R}_{1,j_1}, \ldots, \boldsymbol{R}_{d,j_d}, y_{1,j_1}, \ldots, y_{d,j_d}). \tag{35}$$

Note that by the definition of $\boldsymbol{R}_{\mathsf{ID}}$, $\mathsf{H}(\mathsf{ID})$, $\mathsf{PubEval}$ and $\mathsf{TrapEval}$ (Lemma 6) we have

$$\begin{aligned} \mathsf{H}(\mathsf{ID}) &= \boldsymbol{b}_0 + \sum_{(j_1,\ldots,j_d)\in S(\mathsf{ID})} \mathsf{PubEval}_d(\boldsymbol{b}_{1,j_1}\boldsymbol{b}_{2,j_2}, \ldots, \boldsymbol{b}_{d,j_d}) \\ &= \boldsymbol{a}\boldsymbol{R}_{\mathsf{ID}} + \mathsf{F}_{\boldsymbol{y}}(\mathsf{ID})\boldsymbol{g}_b. \end{aligned} \tag{36}$$

Since $\boldsymbol{R}_0, \boldsymbol{R}_{i,j} \xleftarrow{\$} [-\rho, \rho]_R^{k\times k}$, from Lemma 2 we have $s_1(\boldsymbol{R}_0), s_1(\boldsymbol{R}_{i,j}) \leq B$ with all but negligible probability where $B = C' \cdot \rho\sqrt{n}(\sqrt{k} + \omega(\sqrt{\log n}))$ for some positive absolute constant C'. Furthermore, we have $\|y_{i,j}\|_1 \leq cn$ from Eq. (32). Therefore by Lemma 6, we have

$$\begin{aligned} s_1(\boldsymbol{R}_{\mathsf{ID}}) &\leq s_1(\boldsymbol{R}_0) + \sum_{(j_1,\ldots,j_d)\in S(\mathsf{ID})} s_1(\mathsf{TrapEval}_d(\boldsymbol{R}_{1,j_1}, \ldots, \boldsymbol{R}_{d,j_d}, y_{1,j_1}, \ldots, y_{d,j_d})) \\ &\leq B\left(1 + \kappa(cn)^{d-1} + \kappa bnk\frac{(cn)^{d-1} - 1}{cn - 1}\right), \end{aligned} \tag{37}$$

for any $\mathsf{ID} \in \mathcal{ID}$ with all but negligible probability.

Game_4 In this game, we change the way the vector $\boldsymbol{a}$ is sampled. Namely, Game_4 challenger picks $\boldsymbol{a} \xleftarrow{\$} R_q^k$ instead of generating it with a trapdoor. By Lemma 5, this makes only negligible difference. Furthermore, we also change the way the key extraction queries are answered. When $\mathcal{A}$ makes a key extraction query for an identity ID, the challenger first computes $\boldsymbol{R}_{\mathsf{ID}}$ as in Eq.(35). It aborts if $\mathsf{F}_{\boldsymbol{y}}(\mathsf{ID}) \notin R_q^*$ as in the previous game and runs

$$\mathsf{SampleRight}(\boldsymbol{a}, \boldsymbol{g}_b, \boldsymbol{R}_{\mathsf{ID}}, \mathsf{F}_{\mathbf{y}}(\mathsf{ID}), u, \mathbf{T}_{\boldsymbol{g}_b}, \sigma) \to \boldsymbol{e},$$

otherwise. Note that in the previous game the private key was sampled as

$$\mathsf{SampleLeft}(\boldsymbol{a}, \mathsf{H}(\mathsf{ID}), u, \mathbf{T}_{\boldsymbol{a}}, \sigma) \to \boldsymbol{e}.$$

By Eq.(37) and for our choice of σ, the output distribution of $\mathsf{SampleRight}$ is $\mathsf{negl}(n)$-close to $D^{\mathsf{coeff}}_{\Lambda^{\perp}_{\phi(u)}([\mathrm{rot}(\boldsymbol{a}^T)^T | \mathrm{rot}(\mathsf{H}(\mathsf{ID})^T)^T]), \sigma}$. Furthermore, by the choice of σ, this distribution is $\mathsf{negl}(n)$-close to the output distribution of $\mathsf{SampleLeft}$. Therefore, the above change alters the view of $\mathcal{A}$ only negligibly. Thus, we have $|\Pr[X_3] - \Pr[X_4]| = \mathsf{negl}(n)$.

Game_5 : In this game, we change the way the challenge ciphertext is created when $\mathsf{coin} = 0$. Recall in the previous games when $\mathsf{coin} = 0$, we created a valid challenge ciphertext as in the real scheme. If $\mathsf{coin} = 0$ and $\mathsf{F}_{\boldsymbol{y}}(\mathsf{ID}^\star) = 0$ (i.e., if it does not abort), to create the challenge ciphertext Game_5 challenger first picks $s \xleftarrow{\$} R_q$ and $\boldsymbol{x} \xleftarrow{\$} (D^{\mathsf{coeff}}_{\mathbb{Z}^n, \alpha q})^k$ and computes $\boldsymbol{v} = s\boldsymbol{a} + \boldsymbol{x} \in R^k$. It then runs the algorithm

$$\mathsf{ReRand}\left(\mathrm{rot}\big([\boldsymbol{I}_k | \boldsymbol{R}_{\mathsf{ID}^\star}]\big), \phi(\boldsymbol{v}), \alpha q, \frac{\alpha'}{2\alpha q}\right) \to \mathbf{c} \in \mathbb{Z}_q^{2nk}$$

from Lemma 1, where $\boldsymbol{I}_k \in R^{k \times k}$ is the identity matrix of size $k \times k$. Finally, it picks $x_0 \xleftarrow{\$} D^{\mathsf{coeff}}_{\mathbb{Z}^n, \alpha q}$ and sets the challenge ciphertext as

$$C^\star = \left(c_0 = v_0 + \lfloor q/2 \rceil \cdot \mathsf{M},\ \boldsymbol{c}_1 = \phi^{-1}(\mathbf{c}) \right) \in R_q \times R_q^{2k}, \tag{38}$$

where $v_0 = su + x_0$ and M is the message chosen by $\mathcal{A}$. We claim that this change alters the view of $\mathcal{A}$ only negligibly. To show this, observe that the input to ReRand is $\mathrm{rot}\big([\boldsymbol{I}_k | \boldsymbol{R}_{\mathsf{ID}^\star}]\big) \in \mathbb{Z}_q^{nk \times 2nk}$ and

$$\phi(\boldsymbol{v}) = \phi(s\boldsymbol{a} + \boldsymbol{x}) = \phi(s)\mathrm{rot}(\boldsymbol{a}) + \phi(\boldsymbol{x}) \in \mathbb{Z}_q^{nk},$$

where $\phi(\boldsymbol{x})$ is distributed as $\phi(\boldsymbol{x}) \xleftarrow{\$} D_{\mathbb{Z}^{nk}, \alpha q}$. Therefore, by the property of ReRand and our choice of α and α', the output $\mathbf{c} \in \mathbb{Z}_q^{2nk}$ is

$$\begin{aligned} \mathbf{c} &= \Big(\phi(s)\mathrm{rot}(\boldsymbol{a})\Big) \cdot \mathrm{rot}\big([\boldsymbol{I}_k | \boldsymbol{R}_{\mathsf{ID}^\star}]\big) + \mathbf{x}' \\ &= \phi(s) \cdot \mathrm{rot}\big([\boldsymbol{a} | \mathsf{H}(\mathsf{ID}^\star)]\big) + \mathbf{x}' \\ &= \phi\big(s\big[\boldsymbol{a} | \mathsf{H}(\mathsf{ID}^\star)\big]\big) + \mathbf{x}', \end{aligned}$$

where the distribution of $\mathbf{x}'$ is within negligible distance from $\mathbf{x}' \xleftarrow{\$} D_{\mathbb{Z}^{2nk}, \alpha'}$ due to Lemma 1. Here, we use the fact that $\mathsf{H}(\mathsf{ID}^\star) = \boldsymbol{a}\boldsymbol{R}_{\mathsf{ID}^\star}$ holds since $\mathsf{F}_{\boldsymbol{y}}(\mathsf{ID}^\star) = 0$. It can be readily seen that the distribution of $\boldsymbol{c}_1 = \phi^{-1}(\mathbf{c})$ in Game_5 is statistically close to that in Game_4. Therefore, we have $|\Pr[X_4] - \Pr[X_5]| = \mathsf{negl}(n)$.

Game_6 In this game, we change the way the challenge ciphertext is created when $\mathsf{coin} = 0$. If $\mathsf{coin} = 0$ and the abort condition is not satisfied, to create the challenge ciphertext for identity $\mathsf{ID}^\star$ and message M, Game_6 challenger first picks $v_0 \xleftarrow{\$} R_q$, $\boldsymbol{v}' \xleftarrow{\$} R_q^k$ and $\boldsymbol{x} \xleftarrow{\$} (D^{\mathsf{coeff}}_{\mathbb{Z}^n,\alpha q})^k$, and runs

$$\mathsf{ReRand}\left(\mathrm{rot}\big([\boldsymbol{I}_k|\boldsymbol{R}_{\mathsf{ID}^\star}]\big), \phi(\boldsymbol{v}), \alpha q, \frac{\alpha'}{2\alpha q}\right) \to \mathbf{c} \in \mathbb{Z}_q^{2nk}, \tag{39}$$

where $\boldsymbol{v} = \boldsymbol{v}' + \boldsymbol{x}$. Then, the challenge ciphertext is set as in Eq.(38). As we will show in Lemma 12, assuming $\mathsf{RLWE}_{n,k+1,q,D^{\mathsf{coeff}}_{\mathbb{Z}^n,\alpha q}}$ is hard, we have $|\Pr[X_5] - \Pr[X_6]| = \mathsf{negl}(n)$.

Game_7 In this game, we further change the way the challenge ciphertext is created. When $\mathsf{coin} = 0$ and the abort condition is not satisfied, the challenge ciphertext for $\mathsf{ID}^\star$ is created as

$$C^\star = \left(c_0 = v_0 + \lfloor q/2 \rceil \cdot \mathsf{M},\ \boldsymbol{c}_1 = [\boldsymbol{v}'|\boldsymbol{v}'\boldsymbol{R}_{\mathsf{ID}^\star}] + [\boldsymbol{x}_1|\boldsymbol{x}_2] \right) \in R_q \times R^{2k},$$

where $v_0 \xleftarrow{\$} R_q$, $\boldsymbol{v}' \xleftarrow{\$} R_q^k$ and $\boldsymbol{x}_1, \boldsymbol{x}_2 \xleftarrow{\$} (D^{\mathsf{coeff}}_{\mathbb{Z}^n,\alpha'})^k$.

We claim that this change alters the view of $\mathcal{A}$ only negligibly. This can be seen by a similar argument to that we made in the step from Game_3 to Game_4. We first observe that in Game_6 the input to ReRand is $\mathrm{rot}\big([\boldsymbol{I}_k|\boldsymbol{R}_{\mathsf{ID}^\star}]\big) \in \mathbb{Z}_q^{nk \times 2nk}$ and

$$\phi(\boldsymbol{v}) = \phi(\boldsymbol{v}' + \boldsymbol{x}) = \phi(\boldsymbol{v}') + \phi(\boldsymbol{x}) \in \mathbb{Z}_q^{nk}, \tag{40}$$

where $\phi(\boldsymbol{x})$ is distributed as $D_{\mathbb{Z}^{nk},\alpha q}$. Therefore, the output $\mathbf{c} \in \mathbb{Z}_q^{2nk}$ of ReRand is

$$\mathbf{c} = \phi(\boldsymbol{v}') \cdot \mathrm{rot}\big([\boldsymbol{I}_k|\boldsymbol{R}_{\mathsf{ID}^\star}]\big) + \mathbf{x}' = \phi\big([\boldsymbol{v}'|\boldsymbol{v}'\boldsymbol{R}_{\mathsf{ID}^\star}]\big) + \mathbf{x}',$$

where the distribution of $\mathbf{x}'$ is within negligible distance from $\mathbf{x}' \xleftarrow{\$} D_{\mathbb{Z}^{2nk},\alpha'}$ due to Lemma 1. Hence, the distribution of $\boldsymbol{c}_1 = \phi^{-1}(\mathbf{c})$ in Game_6 is statistically close to that in Game_7. Therefore, we have $|\Pr[X_6] - \Pr[X_7]| = \mathsf{negl}(n)$.

Game_8 In this game, we change the way the key extraction queries are answered. Instead of running $\mathsf{SampleLeft}$ or $\mathsf{SampleRight}$, the (possibly inefficient) challenger directly picks a secret key $\mathsf{sk}_{\mathsf{ID}}$ for identity ID as $\mathsf{sk}_{\mathsf{ID}} \xleftarrow{\$} D^{\mathsf{coeff}}_{\Lambda^\perp_{\phi(u)}([\mathrm{rot}(\boldsymbol{a}^T)^T|\mathrm{rot}(\mathsf{H}(\mathsf{ID})^T)^T]),\sigma}$ without using $\boldsymbol{R}_{\mathsf{ID}}$. Similarly to the change from Game_3 to Game_4, by the choice of σ and Eq.(37), this alters the view of $\mathcal{A}$ only negligibly. Therefore, we have $|\Pr[X_7] - \Pr[X_8]| = \mathsf{negl}(n)$. Note that this is only a conceptual game in order to get rid of any (negligible) correlation between the secret key and $\boldsymbol{R}_{\mathsf{ID}}$ so as not to interfere with the statistical argument using $\boldsymbol{R}_{\mathsf{ID}^\star}$ in the following game.

Game_9 In this game, we change the challenge ciphertext to be a random vector, regardless of whether $\mathsf{coin} = 0$ or $\mathsf{coin} = 1$. Namely, Game_9 challenger generates the challenge ciphertext $C^\star = (c_0, \boldsymbol{c}_1)$ as

$$c_0 \xleftarrow{\$} R_q, \qquad \text{and} \qquad \boldsymbol{c}_1 \xleftarrow{\$} R_q^{2k}.$$

We now proceed to bound $|\Pr[X_8] - \Pr[X_9]|$. Since Game_8 and Game_9 differ only in the creation of the challenge ciphertext when $\mathsf{coin} = 0$, we focus on this case. First, it is easy to see that c_0 is uniformly random over R_q in both of Game_8 and Game_9. Therefore, we only

need to show that the distribution of $\boldsymbol{c}_1$ in Game_8 is $\mathsf{negl}(n)$-close to the uniform distribution over R_q^{2k}. To see this, it suffices to show that $[\boldsymbol{v}'|\boldsymbol{v}'\boldsymbol{R}_{\mathsf{ID}^\star}]$ is distributed statistically close to the uniform distribution over R_q^{2k}. First, observe that the following distributions are $\mathsf{negl}(n)$-close:

$$(\boldsymbol{a}, \boldsymbol{a}\boldsymbol{R}_0, \boldsymbol{v}', \boldsymbol{v}'\boldsymbol{R}_0) \approx (\boldsymbol{a}, \boldsymbol{a}', \boldsymbol{v}', \boldsymbol{v}'') \approx (\boldsymbol{a}, \boldsymbol{a}\boldsymbol{R}_0, \boldsymbol{v}', \boldsymbol{v}''), \tag{41}$$

where $\boldsymbol{a}, \boldsymbol{a}' \xleftarrow{\$} R_q^k$, $\boldsymbol{R}_0 \xleftarrow{\$} [-\rho, \rho]_R^{k\times k}$, $\boldsymbol{v}', \boldsymbol{v}'' \xleftarrow{\$} R_q^k$. It can be seen that the first and the second distributions are $\mathsf{negl}(n)$-close, by applying Lemma 4 for $[\boldsymbol{a}; \boldsymbol{v}'] \in R_q^{2\times k}$ and $\boldsymbol{R}_0$. It can also be seen that the second and the third distributions are $\mathsf{negl}(n)$-close, by applying the same lemma for $\boldsymbol{a}$ and $\boldsymbol{R}_0$. From the above, the following distributions are statistically close:

$$\begin{aligned}
& (\boldsymbol{a}, \boldsymbol{a}\boldsymbol{R}_0, \boldsymbol{v}', \boldsymbol{v}'\boldsymbol{R}_{\mathsf{ID}^\star}) \\
= \; & \left(\boldsymbol{a}, \boldsymbol{a}\boldsymbol{R}_0, \boldsymbol{v}', \boldsymbol{v}'\left(\boldsymbol{R}_0 + \sum_{(j_1,\ldots,j_d)\in S(\mathsf{ID})} \mathsf{TrapEval}_d(\boldsymbol{R}_{1,j_1}, \ldots, \boldsymbol{R}_{d,j_d}, y_{1,j_1}, \ldots, y_{d,j_d})\right)\right) \\
\approx \; & \left(\boldsymbol{a}, \boldsymbol{a}\boldsymbol{R}_0, \boldsymbol{v}', \boldsymbol{v}'' + \boldsymbol{v}'\left(\sum_{(j_1,\ldots,j_d)\in S(\mathsf{ID})} \mathsf{TrapEval}_d(\mathbf{R}_{1,j_1}, \ldots, \mathbf{R}_{d,j_d}, y_{1,j_1}, \ldots, y_{d,j_d})\right)\right) \\
\approx \; & (\boldsymbol{a}, \boldsymbol{a}\boldsymbol{R}_0, \boldsymbol{v}', \boldsymbol{v}'')
\end{aligned}$$

where $\boldsymbol{a}, \boldsymbol{a}' \xleftarrow{\$} R_q^k$, $\boldsymbol{R}_0 \xleftarrow{\$} [-\rho, \rho]_R^{k\times k}$, $\boldsymbol{v}', \boldsymbol{v}'' \xleftarrow{\$} R_q^k$. The second and the third distributions above are $\mathsf{negl}(n)$-close by Eq.(41). Note that we intentionally ignored all the $\boldsymbol{a}\boldsymbol{R}_{i,j}$ terms to keep the argument simple, since focusing on the $\boldsymbol{a}\boldsymbol{R}_0$ term is enough to prove randomness of $[\boldsymbol{v}'|\boldsymbol{v}'\boldsymbol{R}_{\mathsf{ID}^\star}]$. Therefore, we conclude that $|\Pr[X_8] - \Pr[X_9]| = \mathsf{negl}(n)$.

Analysis. From the above, we have

$$\begin{aligned}
\left|\Pr[X_9] - \frac{1}{2}\right| &= \left|\Pr[X_1] - \frac{1}{2} + \sum_{i=1}^{8}(\Pr[X_{i+1}] - \Pr[X_i])\right| \\
&\geq \left|\Pr[X_1] - \frac{1}{2}\right| - \sum_{i=1}^{8}|\Pr[X_{i+1}] - \Pr[X_i]| \\
&\geq \frac{1}{(\kappa c^d n^d)^{(c-1)d+1}}\left(\frac{\epsilon}{2} - \frac{dQ}{n^c}\right) - \mathsf{negl}(n) \\
&= \frac{1}{\mathsf{poly}(n)}\left(\frac{\epsilon}{2} - \frac{dQ}{n^c}\right) - \mathsf{negl}(n)
\end{aligned} \tag{42}$$

where the last equality follows from the facts that c and d are constants and $\kappa = \mathsf{poly}(n)$. Since the challenge ciphertext is independent from the value of coin in Game_9, we have $\Pr[X_9] = 1/2$ and thus $|\Pr[X_9] - 1/2| = 0$. Therefore, we have that $\epsilon/2 - dQ/n^c$ is negligible. However, by Eq.(30),

$$\frac{\epsilon}{2} - \frac{dQ}{n^c} \geq \frac{dQ+1}{n^c} - \frac{dQ}{n^c} = \frac{1}{n^c}$$

holds for infinitely many n, which is a contradiction. □

To complete the proof of Theorem 2, it remains to prove Lemma 11 and 12.

Lemma 11. *For any PPT adversary $\mathcal{A}$, we have*

$$\left|\Pr[X_1] - \frac{1}{2}\right| \geq \frac{1}{(\kappa c^d n^d)^{(c-1)d+1}} \left(\frac{\epsilon}{2} - \frac{dQ}{n^c}\right).$$

Proof. For a sequence of identities $\mathbb{ID} = (\mathsf{ID}^\star, \mathsf{ID}_1, \ldots, \mathsf{ID}_Q) \in \mathcal{ID}^{Q+1}$, we define $\gamma(\mathbb{ID})$ as

$$\gamma(\mathbb{ID}) = \Pr_{\boldsymbol{y}}[\mathsf{F}_{\boldsymbol{y}}(\mathsf{ID}^\star) = 0 \wedge \mathsf{F}_{\boldsymbol{y}}(\mathsf{ID}_1) \neq 0 \wedge \mathsf{F}_{\boldsymbol{y}}(\mathsf{ID}_2) \neq 0 \wedge \cdots \wedge \mathsf{F}_{\boldsymbol{y}}(\mathsf{ID}_Q) \neq 0]$$

where the probability is taken over $\boldsymbol{y} = (y_0, \{y_{i,j}\}_{(i,j)\in[d,\ell]})$, which is chosen as specified in Game_1. Then, it suffices to show

$$\frac{1}{(\kappa c^d n^d)^{(c-1)d+1}} \left(1 - \frac{2dQ}{n^c}\right) \leq \gamma(\mathbb{ID}) \leq \frac{1}{(\kappa c^d n^d)^{(c-1)d+1}} \tag{43}$$

since by Lemma 8, this implies

$$\begin{aligned}
\left|\Pr[X_1] - \frac{1}{2}\right| &\geq \frac{\epsilon}{(\kappa c^d n^d)^{(c-1)d+1}} \left(1 - \frac{2dQ}{n^c}\right) - \frac{1}{2(\kappa c^d n^d)^{(c-1)d+1}} \left(1 - \left(1 - \frac{2dQ}{n^c}\right)\right) \\
&= \frac{1}{(\kappa c^d n^d)^{(c-1)d+1}} \left(\epsilon \left(1 - \frac{2dQ}{n^c}\right) - \frac{dQ}{n^c}\right) \\
&\geq \frac{1}{(\kappa c^d n^d)^{(c-1)d+1}} \left(\frac{\epsilon}{2} - \frac{dQ}{n^c}\right)
\end{aligned}$$

where the last inequality follows from Eq.(30). In the following, we will prove Eq.(43) by applying Lemma 9. We set

$$\begin{aligned}
&\nu = 2,\ \mu = d\ell && \Phi = R_q, \\
&\Omega_j = R_q/\langle t_j \rangle, && \pi_j : R_q \to R_q/\langle t_j \rangle, && \text{for } j \in [2], \\
&S_0 = [-\kappa(cn)^d, -1]_{R,(c-1)d+1}, && S_1 = [1, n]_{R,c}
\end{aligned}$$

where π_j is a natural homomorphism and t_1, t_2 are elements in R_q as defined in Lemma 3. Therefore, the map $\Pi : \Phi \ni y \mapsto (\pi_1(y), \pi_2(y)) \in \Omega_1 \times \Omega_2$ is an isomorphism. We define $f_i(\{Y_{j,j'}\}_{(j,j')\in[d]\times[\ell]})$ for $i \in [0, Q]$ as

$$f_i\left(\{Y_{j,j'}\}_{(j,j')\in[d]\times[\ell]}\right) = \sum_{(j'_1,\ldots,j'_d)\in S(\mathsf{ID}_i)} Y_{1,j'_1} Y_{2,j'_2} \cdots Y_{d,j'_d}$$

where we define $\mathsf{ID}_0 := \mathsf{ID}^\star$. Note that we have $\mathsf{F}_{\boldsymbol{y}}(\mathsf{ID}_i) = y_0 + f_i(\{y_{i,j}\}_{(i,j)\in[d]\times[\ell]})$. We now check that the three conditions for Lemma 9 hold.

- We prove that π_j is injective on S_1 for $j \in \{1, 2\}$. Assume for contradiction that there are $a_1, a_2 \in S_1$ with $a_1 \neq a_2$ and $\pi_j(a_1) = \pi_j(a_2) \Leftrightarrow \pi_j(a_1 - a_2) = 0$. We then have $a_1 - a_2 \notin R_q^*$. On the other hand, we have $\|\phi(a_1 - a_2)\|_2 \leq \sqrt{c}n < \sqrt{q}$. However, this contradicts Lemma 3.

- For $i \in [1, Q]$, we have

$$f_0\left(\{Y_{j,j'}\}\right) - f_i\left(\{Y_{j,j'}\}\right) = \sum_{(j'_1,\ldots,j'_d)\in S(\mathsf{ID}^\star)} Y_{1,j'_1}Y_{2,j'_2}\cdots Y_{d,j'_d} - \sum_{(j'_1,\ldots,j'_d)\in S(\mathsf{ID}_i)} Y_{1,j'_1}Y_{2,j'_2}\cdots Y_{d,j'_d}.$$

  Since $\mathsf{ID}^\star \neq \mathsf{ID}_i$ and S is an injective map, we have $S(\mathsf{ID}^\star) \neq S(\mathsf{ID}_i)$. Therefore, there exists $(j_1^\star, \ldots, j_d^\star) \in [\ell]^d$ such that $(j_1^\star, \ldots, j_d^\star) \in S(\mathsf{ID}^\star) \triangle S(\mathsf{ID}_i)$, where $S(\mathsf{ID}^\star) \triangle S(\mathsf{ID}_i)$ denotes the symmetric difference of $S(\mathsf{ID}^\star)$ and $S(\mathsf{ID}_i)$. Thus, the above polynomial is a non-zero polynomial with degree d. Since the coefficients of $f_0 - f_i$ are all in $\{-1, 0, 1\}$ and $\pi_j(\pm 1) = \pm 1$, $\pi_j(f_0 - f_i)$ is a non-zero polynomial for $j \in \{1, 2\}$ as well.

- We prove $S_0 \supseteq \{-f_i(\{y_{j,j'}\}_{(j,j')\in[d]\times[\ell]}) | y_{1,1}, \ldots, y_{d,\ell} \in S_1\}$ for all $i \in [0, Q]$. By our assumption $d(c-1) < n$ and by regarding elements $y_{j,j'}$ as polynomials in $\mathbb{Z}[X]/(X^n+1)$ with degree $c-1$, we have $f_i(\{y_{j,j'}\})$ are all in $[*, *]_{R,d(c-1)+1}$ where $*$ represents some integer. It then suffices to show $\|\phi(f_i(\{y_{j,j'}\}_{(j,j')\in[d]\times[\ell]}))\|_\infty \leq \kappa(cn)^d$. For any $\{y_{j,j'}\}_{(j,j')\in[d]\times[\ell]}$, we have

$$\begin{aligned}
\|\phi(f_i(\{y_{j,j'}\}_{(j,j')\in[d]\times[\ell]}))\|_\infty &= \left\|\phi\left(\sum_{(j'_1,\ldots,j'_d)\in S(\mathsf{ID}_i)} y_{1,j'_1}y_{2,j'_2}\cdots y_{d,j'_d}\right)\right\|_\infty && (44)\\
&= \left\|\sum_{(j'_1,\ldots,j'_d)\in S(\mathsf{ID}_i)} \phi(y_{1,j'_1}y_{2,j'_2}\cdots y_{d,j'_d})\right\|_\infty && (45)\\
&\leq \sum_{(j'_1,\ldots,j'_d)\in S(\mathsf{ID}_i)} \left\|\phi(y_{1,j'_1}y_{2,j'_2}\cdots y_{d,j'_d})\right\|_\infty && (46)\\
&\leq \kappa(cn)^d && (47)
\end{aligned}$$

  where Eq.(44) follows from the definition, Eq.(45) holds because ϕ^{-1} is a homomorphism, Eq.(46) is from the triangle inequality, and Eq.(47) is from Lemma 7 and the fact that $\|y_{j,j'}\|_\infty \leq n$.

This completes the proof of Lemma 11. □

Lemma 12. *For any PPT adversary $\mathcal{A}$, there exists another PPT adversary $\mathcal{B}$ such that*

$$|\Pr[X_5] - \Pr[X_6]| \leq \mathsf{Adv}_{\mathcal{B}}^{\mathsf{RLWE}_{n,k+1,q,D^{\mathsf{coeff}}_{\mathbb{Z}^n,\alpha q}}}.$$

In particular, under the $\mathsf{RLWE}_{n,k+1,q,D^{\mathsf{coeff}}_{\mathbb{Z}^n,\alpha q}}$ assumption, we have $|\Pr[X_5] - \Pr[X_6]| = \mathsf{negl}(n)$.

Proof. Suppose an adversary $\mathcal{A}$ that has non-negligible advantage in distinguishing Game_5 and Game_6. We use $\mathcal{A}$ to construct an RLWE algorithm denoted $\mathcal{B}$, which proceeds as follows.

Instance. $\mathcal{B}$ is given the problem instance of RLWE $(\{a_i, v_i\}_{i=0}^k) \in (R_q \times R_q)^{k+1}$. We can assume without loss of generality that $v_i = v'_i + x_i$ for $x_i \xleftarrow{\$} D^{\mathsf{coeff}}_{\mathbb{Z}^n,\alpha q}$. Then $\mathcal{B}$'s task is to distinguish whether $v'_i = a_i s$ for some $s \in R_q$ or $v'_i \xleftarrow{\$} R_q$. We note this subtle change from the standard RLWE problem is done only for convenience of the proof.

Setup. To construct master public key mpk, $\mathcal{B}$ first sets

$$u := a_0, \quad \boldsymbol{a} := (a_1, \ldots, a_k), \quad v_0 := v_0, \quad \boldsymbol{v} := (v_1, \ldots, v_k)$$

It also picks $\boldsymbol{y}$ as in Game_1, $\boldsymbol{R}_0, \boldsymbol{R}_{i,j}$ as in Game_2 and sets $\boldsymbol{b}_0$ and $\boldsymbol{b}_{i,j}$ as in Eq.(34). Finally, it returns $\mathsf{mpk} = (\boldsymbol{a}, \boldsymbol{b}_0, \{\boldsymbol{b}_{i,j}\}_{(i,j)\in[d,\ell]}, u)$ to $\mathcal{A}$. $\mathcal{B}$ also picks a random bit $\mathsf{coin} \xleftarrow{\$} \{0,1\}$ and keeps it secret.

Phase 1 and Phase 2. The key extraction queries made by $\mathcal{A}$ are answered as in Game_4. This is done by using $\boldsymbol{R}_0$ and $\boldsymbol{R}_{i,j}$.

Challenge Query. When $\mathcal{A}$ makes the challenge query for the challenge identity $\mathsf{ID}^\star$ and message m, $\mathcal{B}$ first computes $\mathsf{F}_{\boldsymbol{y}}(\mathsf{ID}^\star)$. Then, it aborts and sets $\mathsf{coin}' \xleftarrow{\$} \{0,1\}$ if $\mathsf{F}_{\boldsymbol{y}}(\mathsf{ID}^\star) \neq 0$. Otherwise, it proceeds as follows. If $\mathsf{coin} = 0$, it computes $\boldsymbol{R}_{\mathsf{ID}^\star}$ and $\mathbf{c} \in \mathbb{Z}_q^{2k}$ as in Eq.(39). It then sets the challenge ciphertext $C^\star$ as in Eq. (38). In the case of $\mathsf{coin} = 1$, $\mathcal{B}$ picks $c_0 \xleftarrow{\$} R_q$, $\boldsymbol{c}_1 \xleftarrow{\$} R_q^{2k}$ and sets $C^\star = (c_0, \mathbf{c}_1)$. In both cases, $\mathcal{B}$ returns $C^\star$ to $\mathcal{A}$.

Guess. At last, $\mathcal{A}$ outputs its guess $\widehat{\mathsf{coin}}$ (if the abort condition has not been satisfied). Then, $\mathcal{B}$ sets $\mathsf{coin}' = \widehat{\mathsf{coin}}$. Finally, $\mathcal{B}$ outputs 1 if $\mathsf{coin}' = \mathsf{coin}$ and 0 otherwise.

Analysis. It can be seen that $\mathcal{B}$ perfectly simulates the view of $\mathcal{A}$ in Game_5 if $\{a_i, v_i' + x_i\}_{i=0}^k$ are valid RLWE samples (i.e., $v_i' = a_i s$) and Game_6 otherwise (i.e., $v_i' \xleftarrow{\$} R_q$). We therefore conclude that $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{RLWE}_{n,k+1,q,D_{\mathbb{Z}^n,\alpha q}^{\mathsf{coeff}}}} = |\Pr[X_5] - \Pr[X_6]|$ as desired.

□

5 Construction from Bilinear Maps

In the following, we present our IBE scheme from bilinear maps. Here, for simplicity, we present the scheme with only single-bit message space. A variant of our scheme that can deal with longer message space will appear in Appendix D.1. Let the identity space of the scheme be $\mathcal{ID} = \{0,1\}^\kappa$ for some $\kappa \in \mathbb{N}$. For our construction, we consider an efficiently computable injective map S that maps an identity $\mathsf{ID} \in \{0,1\}^\kappa$ to a subset $S(\mathsf{ID})$ of $[1,\ell] \times [1,\ell]$, where $\ell = \lceil\sqrt{\kappa}\rceil$. We would typically set $\kappa = O(\lambda)$, and thus $\ell = O(\sqrt{\lambda})$ in such a case. We also use $\mathsf{GL}(\mathsf{K}, \mathsf{rand})$ to denote the Goldreich-Levin hardcore bit [GL89] of K using randomness rand. Recall that $\mathsf{GL}(\mathsf{K}, \mathsf{rand})$ is the bitwise inner product between K and rand.

$\mathsf{Setup}(1^\lambda)$: On input 1^λ, it chooses an asymmetric bilinear group $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ with efficiently computable map $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ of prime order $p = p(\lambda)$. Let g and h be generators of $\mathbb{G}_1$ and $\mathbb{G}_2$ respectively. It then picks $w_0, w_{1,1}, \ldots, w_{1,\ell}, w_{2,1}, \ldots, w_{2,\ell}, \alpha, \beta \xleftarrow{\$} \mathbb{Z}_p$ and $\mathsf{rand} \xleftarrow{\$} \{0,1\}^{|\mathbb{G}_T|}$. It finally outputs

$$\begin{aligned} \mathsf{mpk} &= (g, W_0 = g^{w_0}, \{W_{1,i} = g^{w_{1,i}}\}_{i=1}^\ell, \{W_{2,i} = g^{w_{2,i}}\}_{i=1}^\ell, g^\alpha, h^\beta, \mathsf{rand}) \quad \text{and} \\ \mathsf{msk} &= (h, \alpha, \beta, w_0, w_{1,1}, \ldots, w_{1,\ell}, w_{2,1}, \ldots, w_{2,\ell}) \end{aligned}$$

In the following, we use a deterministic function $\mathsf{H} : \mathcal{ID} \to \mathbb{Z}_p$ that is defined as follows.

$$\mathsf{H}(\mathsf{ID}) = w_0 + \sum_{(i,j)\in S(\mathsf{ID})} w_{1,i} w_{2,j} \in \mathbb{Z}_p. \tag{48}$$

$\mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, \mathsf{ID})$: It first computes $\mathsf{H}(\mathsf{ID})$ using msk and picks $r \xleftarrow{\$} \mathbb{Z}_p$. It then returns

$$\mathsf{sk}_{\mathsf{ID}} = (\ A_1 = h^{\alpha\beta + r\cdot\mathsf{H}(\mathsf{ID})},\ A_2 = h^{-r},\ \{B_j = h^{r w_{2,j}}\}_{j=1}^\ell\ ). \tag{49}$$

$\mathsf{Encrypt}(\mathsf{mpk}, \mathsf{ID}, \mathsf{M})$: To encrypt a message $\mathsf{M} \in \{0,1\}$, it picks $s, t_1, \ldots, t_\ell \xleftarrow{\$} \mathbb{Z}_p$ and computes

$$\begin{aligned} C_0 &= \mathsf{M} \oplus \mathsf{GL}\big(e(g^\alpha, h^\beta)^s, \mathsf{rand}\big), \quad C_1 = g^s, \quad C_2 = W_0^s \cdot \prod_{j\in[1,\ell]} W_{2,j}^{t_j}, \\ D_j &= g^{t_j} \cdot \left(\prod_{i\in\{i\in[1,\ell]|(i,j)\in S(\mathsf{ID})\}} W_{1,i} \right)^{-s} \qquad \text{for} \quad j \in [1,\ell] \end{aligned} \tag{50}$$

Finally, it returns the ciphertext $C = (C_0, C_1, C_2, \{D_j\}_{j=1}^\ell)$.

$\mathsf{Decrypt}(\mathsf{mpk}, \mathsf{sk}_{\mathsf{ID}}, C)$: To decrypt a ciphertext $C = (C_0, C_1, C_2, \{D_j\}_{j=1}^\ell)$ using a private key $\mathsf{sk}_{\mathsf{ID}} = (A_1, A_2, \{B_j\}_{j=1}^\ell)$, it first computes

$$e(C_1, A_1) \cdot e(C_2, A_2) \cdot \prod_{j\in[1,\ell]} e(D_j, B_j) = e(g,h)^{s\alpha\beta}. \tag{51}$$

Then it retrieves the message by $C_0 \oplus \mathsf{GL}(e(g,h)^{s\alpha\beta}, \mathsf{rand})$.

5.1 Correctness of the Single-bit Variant

To verify the correctness of the scheme, it suffices to show Eq.(51). Let $g_T := e(g,h)$. We have

$$\begin{aligned} & \log_{g_T} \left(e(C_1, A_1) \cdot e(C_2, A_2) \cdot \prod_{j\in[1,\ell]} e(D_j, B_j) \right) \\ = \; & \log_{g_T} e(C_1, A_1) - r \left(s w_0 + \sum_{j\in[1,\ell]} t_j w_{2,j} \right) + \sum_{j\in[1,\ell]} r w_{2,j} \left(t_j - s \sum_{i\in\{i\in[1,\ell]|(i,j)\in S(\mathsf{ID})\}} w_{1,i} \right) \\ = \; & \log_{g_T} e(C_1, A_1) - rsw_0 - rs \sum_{j\in[1,\ell]} \left(\sum_{i\in\{i\in[1,\ell]|(i,j)\in S(\mathsf{ID})\}} w_{1,i} w_{2,j} \right) \\ = \; & s\alpha\beta + rs \left(w_0 + \sum_{(i,j)\in S(\mathsf{ID})} w_{1,i} w_{2,j} \right) - rs \left(w_0 + \sum_{(i,j)\in S(\mathsf{ID})} w_{1,i} w_{2,j} \right) \\ = \; & s\alpha\beta. \end{aligned}$$

Therefore, Eq.(51) follows.

5.2 Security Proof for the Single-bit Variant

The security of the scheme is proven under the 3-CBDHE assumption defined below.

Definition 2 (3-Computational Bilinear Diffie-Hellman Exponent (3-CBDHE) Assumption)**.** *We say that 3-CBDHE holds on* $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ *if*

$$\Pr[\mathcal{A}(g, g^s, g^a, g^{a^2}, h, h^a, h^{a^2}) \to e(g,h)^{sa^3}]$$

is negligible for any PPT adversary $\mathcal{A}$ *where* $g \xleftarrow{\$} \mathbb{G}_1$, $h \xleftarrow{\$} \mathbb{G}_2$, $s, a \xleftarrow{\$} \mathbb{Z}_p$.

We also introduce the following lemma concerning the Goldreich-Levin hardcore bit function which we use during our security proof.

Lemma 13 ([GL89]). *Let us assume that the 3-CBDHE assumption holds. Then, for any PPT adversary $\mathcal{A}$,*

$$\mathsf{Adv}^{\mathsf{3CBDHE}}_{\mathcal{A}} = \left|\Pr[\mathcal{A}\big(\Psi, \mathsf{rand}, \mathsf{GL}(e(g,h)^{sa^3}, \mathsf{rand})\big) \to 1] - \Pr[\mathcal{A}\big(\Psi, \mathsf{rand}, T\big) \to 1]\right|$$

is negligible where $\Psi = (g, g^s, g^a, g^{a^2}, h, h^a, h^{a^2})$, $a, s \xleftarrow{\$} \mathbb{Z}_p$, $T \xleftarrow{\$} \{0,1\}$ and $\mathsf{rand} \xleftarrow{\$} \{0,1\}^{|\mathbb{G}_T|}$.

The following theorem addresses the security of the scheme.

Theorem 3. *The above IBE scheme is adaptively secure assuming the 3-CBDHE assumption.*

Proof. Let $\mathcal{A}$ be a PPT adversary that breaks adaptive security of the scheme. In addition, let $\epsilon = \epsilon(\lambda)$ and $Q = Q(\lambda)$ be its advantage and the upper bound on the number of key extraction queries, respectively. Since $\mathcal{A}$ is PPT, there exists a constant number $c_1 \in \mathbb{N}$ such that $4(Q+1) \leq \lambda^{c_1}$ for all $\lambda \in \mathbb{N}$. Similarly, since $\mathcal{A}$ breaks the security of the scheme, there exists $c_2 \in \mathbb{N}$ such that $2\epsilon \geq \lambda^{-c_2}$ holds for infinitely many λ. By setting $c = c_1 + c_2$, we have that

$$4Q \leq \lambda^c \quad \text{for all } \lambda \in \mathbb{N} \quad \text{and} \quad \frac{\epsilon}{2(Q+1)} \geq \frac{1}{\lambda^c} \quad \text{for infinitely many } \lambda \in \mathbb{N}. \tag{52}$$

In the following, we assume that $p > \lambda^c$. Since the size of p is exponential in λ, this holds for sufficiently large λ.

We show the security of the scheme via the following games. In each game, a value $\mathsf{coin}' \in \{0,1\}$ is defined. While it is set $\mathsf{coin}' = \widehat{\mathsf{coin}}$ in the first game, these values might be different in the later games. In the following, we define X_i be the event that $\mathsf{coin}' = \mathsf{coin}$ in Game_i.

Game_0 : This is the real security game. Since the message space is $\{0,1\}$, without loss of generality, we assume that the adversary always chooses $\mathsf{M}_0 = 0$ and $\mathsf{M}_1 = 1$ as its target in the challenge phase. Then the challenger picks a random coin $\mathsf{coin} \xleftarrow{\$} \{0,1\}$ and returns an encryption of $\mathsf{M}_{\mathsf{coin}} = \mathsf{coin}$ as the challenge ciphertext. At the end of the game, $\mathcal{A}$ outputs a guess $\widehat{\mathsf{coin}}$ for coin. Finally, the challenger sets $\mathsf{coin}' = \widehat{\mathsf{coin}}$. By the definition, we have

$$\left|\Pr[X_0] - \frac{1}{2}\right| = \left|\Pr[\mathsf{coin}' = \mathsf{coin}] - \frac{1}{2}\right| = \left|\Pr[\widehat{\mathsf{coin}} = \mathsf{coin}] - \frac{1}{2}\right| = \epsilon.$$

Game_1 : In this game, we change Game_0 so that the challenger performs the following additional step at the end of the game. First, the challenger picks $\mathbf{y} = (y_0, \{y_{i,j}\}_{(i,j)\in[2]\times[\ell]})$ as

$$y_0 \xleftarrow{\$} [-\kappa\lambda^{2c}, -1] \quad \text{and} \quad y_{i,j} \xleftarrow{\$} [1, \lambda^c] \quad \text{for} \quad (i,j) \in [2] \times [\ell]. \tag{53}$$

We define a function $\mathsf{F}_{\mathbf{y}} : \mathcal{ID} \to \mathbb{Z}_p$ as follows:

$$\mathsf{F}_{\mathbf{y}}(\mathsf{ID}) = y_0 + \sum_{(j_1,j_2)\in S(\mathsf{ID})} y_{1,j_1} y_{2,j_2}.$$

Then the challenger checks whether the following condition holds:

$$\mathsf{F}_{\mathbf{y}}(\mathsf{ID}^\star) = 0 \ \wedge \ \mathsf{F}_{\mathbf{y}}(\mathsf{ID}_1) \neq 0 \ \wedge \ \cdots \ \wedge \ \mathsf{F}_{\mathbf{y}}(\mathsf{ID}_Q) \neq 0 \tag{54}$$

where $\mathsf{ID}^\star$ is the challenge identity, and $\mathsf{ID}_1, \ldots, \mathsf{ID}_Q$ are identities for which $\mathcal{A}$ has made key extraction queries. If it does not hold, the challenger ignores the output $\widehat{\mathsf{coin}}$ of $\mathcal{A}$, and sets $\mathsf{coin}' \xleftarrow{\$} \{0,1\}$. Otherwise, the challenger sets $\mathsf{coin}' = \widehat{\mathsf{coin}}$. In Lemma 14, we will show that

$$\left|\Pr[X_1] - \frac{1}{2}\right| \geq \frac{1}{\kappa\lambda^{2c}}\left(\frac{\epsilon}{2} - \frac{Q}{\lambda^c}\right).$$

Game_2 In this game, we change the way α, β, w_0, and $w_{i,j}$ are chosen. At the beginning of the game, the challenger picks $\mathbf{y}$ as in Game_1. It then picks $a, \tilde{w}_0, \tilde{w}_{1,1}, \ldots, \tilde{w}_{1,\ell}, \tilde{w}_{2,1}, \ldots, \tilde{w}_{2,\ell} \xleftarrow{\$} \mathbb{Z}_p$, $\tilde{\alpha}, \tilde{\beta} \xleftarrow{\$} \mathbb{Z}_p^*$ and sets

$$\alpha = a\tilde{\alpha},\ \beta = a^2\tilde{\beta},\ w_0 = a^2 y_0 + \tilde{w}_0,\ w_{i,j} = a y_{i,j} + \tilde{w}_{i,j} \text{ for } (i,j) \in [2] \times [\ell]. \tag{55}$$

This change does not alter the distribution of w_0, $w_{i,j}$, α, and β. Since this change is only conceptual, we have

$$\Pr[X_2] = \Pr[X_1].$$

Game_3 Recall that in the previous game, the challenger aborts at the end of the game, if the condition (54) is not satisfied. In this game, we change the game so that the challenger aborts as soon as the abort condition becomes true. Since this is only a conceptual change, we have

$$\Pr[X_3] = \Pr[X_2].$$

Before describing the next game, we observe that $\mathsf{H}(\mathsf{ID})$ can be written as an polynomial in a with degree 2 whose coefficients depend on ID and $\mathbf{y}$.

$$\begin{aligned}
&\mathsf{H}(\mathsf{ID}) \\
=\ & w_0 + \sum_{(i,j)\in S(\mathsf{ID})} w_{1,i} w_{2,j} \\
=\ & y_0 a^2 + \tilde{w}_0 + \sum_{(i,j)\in S(\mathsf{ID})} (y_{1,i} a + \tilde{w}_{1,i})(y_{2,j} a + \tilde{w}_{2,j}) \\
=\ & \underbrace{\left(y_0 + \sum_{(i,j)\in S(\mathsf{ID})} y_{1,i} y_{2,j}\right)}_{=\mathsf{F}_{\mathbf{y}}(\mathsf{ID})} a^2 + \underbrace{\left(\sum_{(i,j)\in S(\mathsf{ID})} \tilde{w}_{1,i} y_{2,j} + y_{1,i} \tilde{w}_{2,j}\right)}_{:=\mathsf{G}_{\mathbf{y}}(\mathsf{ID})} a + \underbrace{\left(\tilde{w}_0 + \sum_{(i,j)\in S(\mathsf{ID})} \tilde{w}_{1,i} \tilde{w}_{2,j}\right)}_{:=\mathsf{I}_{\mathbf{y}}(\mathsf{ID})} \\
=\ & \mathsf{F}_{\mathbf{y}}(\mathsf{ID}) a^2 + \mathsf{G}_{\mathbf{y}}(\mathsf{ID}) a + \mathsf{I}_{\mathbf{y}}(\mathsf{ID}).
\end{aligned}$$

Game_4 : In this game, we change the way the key extraction queries are answered. When $\mathcal{A}$ makes a key extraction query for an identity ID, the challenger aborts if $\mathsf{F}_{\mathbf{y}}(\mathsf{ID}) = 0$ as the previous game. Otherwise, it first picks $\tilde{r} \xleftarrow{\$} \mathbb{Z}_p$ and sets r as

$$r = \tilde{r} - \frac{\tilde{\alpha}\tilde{\beta}}{\mathsf{F}_{\mathbf{y}}(\mathsf{ID})} a. \tag{56}$$

Then the private key is generated as Eq.(49). Clearly, this is only a conceptual change and does not change the view of $\mathcal{A}$. Therefore, we have

$$\Pr[X_4] = \Pr[X_3].$$

Here, we observe that

$$\begin{aligned} & \alpha\beta + r\mathsf{H}(\mathsf{ID}) \\ = \; & a^3\tilde{\alpha}\tilde{\beta} + \left(\mathsf{F}_{\mathbf{y}}(\mathsf{ID})a^2 + \mathsf{G}_{\mathbf{y}}(\mathsf{ID})a + \mathsf{I}_{\mathbf{y}}(\mathsf{ID})\right)\left(\tilde{r} - \frac{\tilde{\alpha}\tilde{\beta}}{\mathsf{F}_{\mathbf{y}}(\mathsf{ID})}a\right) \\ = \; & \left(\tilde{r}\mathsf{F}_{\mathbf{y}}(\mathsf{ID}) - \frac{\tilde{\alpha}\tilde{\beta}\mathsf{G}_{\mathbf{y}}(\mathsf{ID})}{\mathsf{F}_{\mathbf{y}}(\mathsf{ID})}\right)a^2 + \left(\tilde{r}\mathsf{G}_{\mathbf{y}}(\mathsf{ID}) - \frac{\tilde{\alpha}\tilde{\beta}\mathsf{I}_{\mathbf{y}}(\mathsf{ID})}{\mathsf{F}_{\mathbf{y}}(\mathsf{ID})}\right)a + \tilde{r}\cdot\mathsf{I}_{\mathbf{y}}(\mathsf{ID}) \end{aligned} \tag{57}$$

and

$$\begin{aligned} rw_{2,j} &= \left(\tilde{r} - \frac{\tilde{\alpha}\tilde{\beta}}{\mathsf{F}_{\mathbf{y}}(\mathsf{ID})}a\right)(y_{2,j}a + \tilde{w}_{2,j}) \\ &= -\frac{\tilde{\alpha}\tilde{\beta}y_{2,j}}{\mathsf{F}_{\mathbf{y}}(\mathsf{ID})}a^2 + \left(\tilde{r}y_{2,j} - \frac{\tilde{\alpha}\tilde{\beta}\tilde{w}_{2,j}}{\mathsf{F}_{\mathbf{y}}(\mathsf{ID})}\right)a + r\tilde{w}_{2,j}. \end{aligned} \tag{58}$$

It can be seen that the term $a^3\tilde{\alpha}\tilde{\beta}$ cancels out in Eq.(57). Looking ahead, this is essential for the reduction from the 3-CBDHE assumption (Lemma 15) to be possible.

Game_5 In this game, we change the way the challenge ciphertext is created. When creating the challenge ciphertext, the challenger first picks $s', \tilde{t}_1, \ldots, \tilde{t}_\ell \xleftarrow{\$} \mathbb{Z}_p$ and sets

$$s = \frac{s'}{\tilde{\alpha}\tilde{\beta}}, \quad t_j = \begin{cases} \tilde{t}_1 + s\left(-\dfrac{\mathsf{G}_{\mathbf{y}}(\mathsf{ID}^\star)}{y_{2,1}} + \displaystyle\sum_{i\in\{i\in[1,\ell]\mid(i,1)\in S(\mathsf{ID}^\star)\}} w_{1,i}\right) & \text{for } j = 1 \\ \tilde{t}_j + s\left(\displaystyle\sum_{i\in\{i\in[1,\ell]\mid(i,j)\in S(\mathsf{ID}^\star)\}} w_{1,i}\right) & \text{for } j \in [2,\ell]. \end{cases} \tag{59}$$

Then, the challenge ciphertext is computed as Eq.(50). Note that since $1 \le y_{2,1} \le \lambda^c < p$ and thus $y \neq 0 \mod p$, the denominator in Eq.(59) is well-defined. Clearly, this is only a conceptual change and does not change the view of $\mathcal{A}$. Therefore, we have

$$\Pr[X_5] = \Pr[X_4].$$

Here, we observe that

$$C_0 = \mathsf{coin} \oplus \mathsf{GL}\left(e(g,h)^{s'a^3}, \mathsf{rand}\right), \; D_1 = g^{\tilde{t}_1}(g^{s'})^{-\mathsf{G}_{\mathbf{y}}(\mathsf{ID}^\star)/\tilde{\alpha}\tilde{\beta}y_{2,1}}, \; D_j = g^{\tilde{t}_j} \;\text{ for }\; j \in [2,\ell] \tag{60}$$

and

$$\begin{aligned} & \log_g C_2 \\ = \; & sw_0 + \sum_{j\in[1,\ell]} w_{2,j}t_j \end{aligned}$$

$$
\begin{aligned}
&= sw_0 - w_{2,1}s\left(\frac{\mathsf{G}_{\mathbf{y}}(\mathsf{ID}^\star)}{y_{2,1}}\right) + \sum_{j\in[1,\ell]} w_{2,j}\left(\tilde{t}_j + s\left(\sum_{i\in\{i\in[1,\ell]|(i,j)\in S(\mathsf{ID}^\star)\}} w_{1,i}\right)\right)\\
&= -w_{2,1}s\left(\frac{\mathsf{G}_{\mathbf{y}}(\mathsf{ID}^\star)}{y_{2,1}}\right) + \left(\sum_{j\in[1,\ell]} w_{2,j}\tilde{t}_j\right) + s\underbrace{\left(w_0 + \sum_{j\in[1,\ell]}\sum_{i\in\{i\in[1,\ell]|(i,j)\in S(\mathsf{ID}^\star)\}} w_{1,i}w_{2,j}\right)}_{=\mathsf{H}(\mathsf{ID}^\star)}\\
&= -s(y_{2,1}a + \tilde{w}_{2,1})\left(\frac{\mathsf{G}_{\mathbf{y}}(\mathsf{ID}^\star)}{y_{2,1}}\right) + \left(\sum_{j\in[1,\ell]} w_{2,j}\tilde{t}_j\right) + s\left(\underbrace{\mathsf{F}_{\mathbf{y}}(\mathsf{ID}^\star)}_{=0}a^2 + \mathsf{G}_{\mathbf{y}}(\mathsf{ID}^\star)a + \mathsf{I}_{\mathbf{y}}(\mathsf{ID}^\star)\right)\\
&= \cancel{-\mathsf{G}_{\mathbf{y}}(\mathsf{ID}^\star)sa} - s\left(\frac{\tilde{w}_{2,1}\mathsf{G}_{\mathbf{y}}(\mathsf{ID}^\star)}{y_{2,1}}\right) + \left(\sum_{j\in[1,\ell]} w_{2,j}\tilde{t}_j\right) + \cancel{\mathsf{G}_{\mathbf{y}}(\mathsf{ID}^\star)sa} + s\cdot\mathsf{I}_{\mathbf{y}}(\mathsf{ID}^\star)\\
&= s'\left(\frac{y_{2,1}\cdot\mathsf{I}_{\mathbf{y}}(\mathsf{ID}^\star) - \tilde{w}_{2,1}\cdot\mathsf{G}_{\mathbf{y}}(\mathsf{ID}^\star)}{\tilde{\alpha}\tilde{\beta}y_{2,1}}\right) + a\left(\sum_{j\in[1,\ell]} y_{2,j}\tilde{t}_j\right) + \left(\sum_{j\in[1,\ell]} \tilde{w}_{2,j}\tilde{t}_j\right). \qquad (61)
\end{aligned}
$$

It can be seen that the term $-\mathsf{G}_{\mathbf{y}}(\mathsf{ID}^\star)sa$ cancels out in Eq.(61). Looking ahead, this is essential for the reduction from the 3-CBDHE assumption (Lemma 15) to be possible.

Game_6 In this game, the component C_0 in the challenge ciphertext is changed to be a random bit. As we will show in Lemma 15, assuming the 3-CBDHE assumption is hard, we have

$$|\Pr[X_6] - \Pr[X_5]| = \mathsf{negl}(n). \qquad (62)$$

Analysis. From the above, we have

$$
\begin{aligned}
\left|\Pr[X_6] - \frac{1}{2}\right| &= \left|\Pr[X_1] - \frac{1}{2} + \sum_{i=1}^{5}\Pr[X_{i+1}] - \Pr[X_i]\right|\\
&\geq \left|\Pr[X_1] - \frac{1}{2}\right| - \sum_{i=1}^{5}|\Pr[X_{i+1}] - \Pr[X_i]|\\
&\geq \frac{1}{\kappa\lambda^{2c}}\left(\frac{\epsilon}{2} - \frac{Q}{\lambda^c}\right) - \mathsf{negl}(\lambda)\\
&= \frac{1}{\mathsf{poly}(\lambda)}\left(\frac{\epsilon}{2} - \frac{Q}{\lambda^c}\right) - \mathsf{negl}(\lambda). \qquad (63)
\end{aligned}
$$

Since the challenge ciphertext is independent from the value of coin in Game_6, we have $\Pr[X_6] = 1/2$ and thus $|\Pr[X_6] - 1/2| = 0$. Therefore, we have that $\epsilon/2 - Q/\lambda^c$ is negligible. However, by Eq.(52),

$$\frac{\epsilon}{2} - \frac{Q}{\lambda^c} \geq \frac{Q+1}{\lambda^c} - \frac{Q}{\lambda^c} = \frac{1}{\lambda^c}$$

holds for infinitely many λ, which is a contradiction. □

To complete the proof of Theorem 3, it remains to show Lemma 14 and Lemma 15.

Lemma 14. *For any PPT adversary $\mathcal{A}$, we have*

$$\left|\Pr[X_1] - \frac{1}{2}\right| \geq \frac{1}{\kappa\lambda^{2c}}\left(\frac{\epsilon}{2} - \frac{Q}{\lambda^c}\right).$$

Proof. For a sequence of identities $\mathbb{ID} = (\mathsf{ID}^\star, \mathsf{ID}_1, \ldots, \mathsf{ID}_Q) \in \mathcal{ID}^{Q+1}$, we define $\gamma(\mathbb{ID})$ as

$$\gamma(\mathbb{ID}) = \Pr_{\mathbf{y}}[\mathsf{F}_{\mathbf{y}}(\mathsf{ID}^\star) = 0 \wedge \mathsf{F}_{\mathbf{y}}(\mathsf{ID}_1) \neq 0 \wedge \mathsf{F}_{\mathbf{y}}(\mathsf{ID}_2) \neq 0 \wedge \cdots \wedge \mathsf{F}_{\mathbf{y}}(\mathsf{ID}_Q) \neq 0]$$

where the probability is taken over $\boldsymbol{y} = (y_0, \{y_{i,j}\}_{(i,j)\in[2,\ell]})$, which is chosen as specified in Game_1. It suffices to show

$$\frac{1}{\kappa\lambda^{2c}}\left(1 - \frac{2Q}{\lambda^c}\right) \leq \gamma(\mathbb{ID}) \leq \frac{1}{\kappa\lambda^{2c}} \tag{64}$$

since due to Lemma 8, this implies

$$\begin{aligned}\left|\Pr[X_1] - \frac{1}{2}\right| &\geq \frac{\epsilon}{\kappa\lambda^{2c}}\left(1 - \frac{2Q}{\lambda^c}\right) - \frac{1}{2\kappa\lambda^{2c}}\left(1 - \left(1 - \frac{2Q}{\lambda^c}\right)\right) \\ &\geq \frac{1}{\kappa\lambda^{2c}}\left(\frac{\epsilon}{2} - \frac{Q}{\lambda^c}\right)\end{aligned}$$

where we used Eq.(52) in the last inequality. In the following, we will prove Eq.(64) by applying Lemma 9. We would set

$$\begin{aligned} &d = 2, && \nu = 1, && \Phi = \Omega_1 = \mathbb{Z}_p, \\ &\Pi = \pi_1 = \mathrm{id}_{\mathbb{Z}_p}, && S_0 = [-\kappa\lambda^{2c}, -1], && S_1 = [1, \lambda^c] \end{aligned}$$

where $\mathrm{id}_{\mathbb{Z}_p}$ denotes the identity map on $\mathbb{Z}_p$. We set $\mu = 2\ell$ and define $f_i(\{Y_{j,j'}\}_{(j,j')\in[2]\times[\ell]})$ for $i \in [0, Q]$ as

$$f_i\left(\{Y_{j,j'}\}_{(j,j')\in[2]\times[\ell]}\right) = \sum_{(j'_1,j'_2)\in S(\mathsf{ID}_i)} Y_{1,j'_1}Y_{2,j'_2}$$

where we define $\mathsf{ID}_0 := \mathsf{ID}^\star$. Note that we have $\mathsf{F}_{\mathbf{y}}(\mathsf{ID}_i) = y_0 + f_i(\{y_{j,j'}\}_{(j,j')\in[2]\times[\ell]})$. We now check that the three conditions for Lemma 9 hold.

- π_1 is injective on S_1 because it is the identity map on $\mathbb{Z}_p$ and $\lambda^c < p$.
- For $i \in [1, Q]$, we have

  $$f_0\left(\{Y_{j,j'}\}\right) - f_i\left(\{Y_{j,j'}\}\right) = \sum_{(j'_1,j'_2)\in S(\mathsf{ID}^\star)} Y_{1,j'_1}Y_{2,j'_2} - \sum_{(j'_1,j'_2)\in S(\mathsf{ID}_i)} Y_{1,j'_1}Y_{2,j'_2}.$$

  Since $\mathsf{ID}^\star \neq \mathsf{ID}_i$ and S is an injective map, we have $S(\mathsf{ID}^\star) \neq S(\mathsf{ID}_i)$. Therefore, there exists $(j_1^\star, j_2^\star) \in [\ell] \times [\ell]$ such that $(j_1^\star, j_2^\star) \in S(\mathsf{ID}^\star) \triangle S(\mathsf{ID}_i)$, where $S(\mathsf{ID}^\star) \triangle S(\mathsf{ID}_i)$ denotes the symmetric difference of $S(\mathsf{ID}^\star)$ and $S(\mathsf{ID}_i)$. Thus, the above polynomial is a non-zero polynomial with degree 2.
- Since $S_1 = [1, \lambda^c]$, we have

  $$1 \leq f_i(\{y_{j,j'}\}) = \sum_{(j'_1,j'_2)\in S(\mathsf{ID}_i)} y_{1,j'_1}y_{2,j'_2} \leq \sum_{(j'_1,j'_2)\in S(\mathsf{ID}_i)} \lambda^c \cdot \lambda^c \leq \kappa\lambda^{2c}$$

  for $i \in [Q]$. Therefore, we have $S_0 \supseteq \{-f_i(\{y_{j,j'}\}_{(j,j')\in[2]\times[\ell]}) | y_{j,j'} \in S_1\}$ for all $i \in [0, Q]$.

This completes the proof of Lemma 14. □

Lemma 15. *For any PPT adversary $\mathcal{A}$, there exists another PPT adversary $\mathcal{B}$ such that*

$$|\Pr[X_5] - \Pr[X_6]| \leq \mathsf{Adv}^{\mathsf{3CBDHE}}_{\mathcal{B}}.$$

In particular, under the 3CBDHE *assumption, we have* $|\Pr[X_5] - \Pr[X_6]| = \mathsf{negl}(n)$.

Proof. Suppose an adversary $\mathcal{A}$ that has non-negligible advantage in distinguishing Game_5 and Game_6. We use $\mathcal{A}$ to construct an 3CBDHE algorithm denoted $\mathcal{B}$, which proceeds as follows.

Instance. $\mathcal{B}$ is given the problem instance of 3CBDHE $\left(g, g^{s'}, g^{a}, g^{a^2}, h, h^{a}, h^{a^2}, \mathsf{rand}, T\right)$. The task of $\mathcal{B}$ is to distinguish whether $T = \mathsf{GL}(e(g,h)^{s'a^3}, \mathsf{rand})$ or $T \xleftarrow{\$} \{0,1\}$.

Setup. To construct master public key mpk, $\mathcal{B}$ first picks $\mathbf{y}$ as in Game_2. It also picks $\tilde{w}_0, \tilde{w}_{i,j}, \tilde{\alpha}, \tilde{\beta}$ and implicitly sets $w_0, w_{i,j}, \alpha, \beta$ as in Game_3. Then, $\mathcal{B}$ computes mpk as follows:

$$\mathsf{mpk} = \left(g, \begin{array}{ll} g^{\alpha} = (g^{a})^{\tilde{\alpha}}, & W_0 = (g^{a^2})^{y_0} \cdot g^{\tilde{w}_0}, \\ h^{\beta} = (h^{a^2})^{\tilde{\beta}}, & \{W_{i,j} = (g^{a})^{y_{i,j}} \cdot g^{\tilde{w}_{i,j}}\}_{(i,j)\in[2,\ell]}, \end{array} \mathsf{rand} \right). \tag{65}$$

Note that these values can be computed without explicitly knowing a. Finally, it returns mpk to $\mathcal{A}$. $\mathcal{B}$ also picks a random bit $\mathsf{coin} \xleftarrow{\$} \{0,1\}$ and keeps it secret.

Phase 1 and Phase 2. When $\mathcal{A}$ makes a key extraction query for ID, $\mathcal{B}$ proceeds as follows. We assume $\mathsf{F}(\mathsf{ID}) \neq 0$ since otherwise $\mathcal{B}$ aborts. By the change introduced in Game_4, we have that each component of $\mathsf{sk}_{\mathsf{ID}}$ can be written as a linear combination of (h, h^{a}, h^{a^2}) with the coefficients being known to $\mathcal{B}$ (See Eq.(56), (57), and (58)). Therefore, $\mathcal{B}$ can compute the secret key without explicitly knowing the value of a.

Challenge Query. When $\mathcal{A}$ makes the challenge query for the challenge identity $\mathsf{ID}^{\star}$, $\mathcal{B}$ proceeds as follows. We assume $\mathsf{F}_{\mathbf{y}}(\mathsf{ID}^{\star}) \neq 0$ since otherwise $\mathcal{B}$ aborts. By the change introduced in Game_6, C_1, C_2, $\{D_j\}_{j=1}^{\ell}$ in the challenge ciphertext can be written as a linear combination of $(g^{s'}, g, g^{a}, g^{a^2})$ (See Eq.(60) and (61)). $\mathcal{B}$ can therefore compute these components. Finally, $\mathcal{B}$ sets $C_0 = T \oplus \mathsf{coin}$ and gives the challenge ciphertext $C^{\star} = (C_0, C_1, C_2, \{D_j = g^{t_j}\}_{j\in[1,\ell]})$ to $\mathcal{A}$.

Guess. At last, $\mathcal{A}$ outputs its guess $\widehat{\mathsf{coin}}$ (if the abort condition has not been satisfied). Then, $\mathcal{B}$ sets $\mathsf{coin}' = \widehat{\mathsf{coin}}$. Finally, $\mathcal{B}$ outputs 1 if $\mathsf{coin}' = \mathsf{coin}$ and 0 otherwise.

Analysis. It can be seen that the view of $\mathcal{A}$ corresponds to that in Game_5 if $T = \mathsf{GL}(e(g,h)^{s'a^3}, \mathsf{rand})$ and Game_6 if $T \xleftarrow{\$} \{0,1\}$. Therefore, we have $|\Pr[X_5] - \Pr[X_6]| \leq \mathsf{Adv}^{\mathsf{3CBDHE}}_{\mathcal{B}}$. □

6 Comparisons and Discussions

In this section, we compare our IBE schemes obtained in Sec. 4 and 5 with previous schemes. Throughout this section, $|\mathsf{mpk}|$, $|C|$, and $|\mathsf{sk}_{\mathsf{ID}}|$ denote the sizes of the master public keys, ciphertexts, and private keys, respectively. We denote by κ the length of the identity, which corresponds to the output length of the collision resistant hash if we choose to hash the bit string representing an identity.

Ideal Lattice Based IBE. In Sec. 4. we proposed a new ideal lattice based IBE scheme. By changing the base b of the $\boldsymbol{g}_b$-trapdoor, we obtain two types of instantiation offering tradeoffs. Namely, by setting $b = 2$ we obtain the Type 1 IBE scheme presented in Appendix C, and by setting $b = n^{\frac{1}{4}}$ we obtain the Type 2 IBE scheme presented in Sec. 4.1. The Type 2 IBE allows for

a more compact size parameters compared to the Type 1 IBE, whereas the Type 1 IBE allows for a more efficient sampling procedure due to the smaller Gaussian width. Note that the technique of changing the base b is applicable for other existing IBE schemes as well, offering a similar tradeoff presented above. Both of our schemes achieve the best efficiency among existing adaptively secure IBE schemes assuming the fixed polynomial approximation of the RLWE problem. This is illustrated in Table 1. We point out that the largest improvement from the Yamada's IBE is that we greatly weakened the underlying hardness assumption while improving the overall efficiency of the scheme.

Table 1: Comparison of Lattice-Base IBEs in the Standard Model.

Schemes	$\lvert\mathsf{mpk}\rvert$	$\lvert C\rvert$, $\lvert\mathsf{sk}_{\mathsf{ID}}\rvert$	$1/\alpha$ for LWE Assumption	Anonymous?
[CHKP10]	$O(n\kappa \log^2 n)$	$O(n\kappa \log^2 n)$	Fixed $\mathsf{poly}(n)$	Yes
[ABB10]+[Boy10]*	$O(n\kappa \log^2 n)$	$O(n \log^2 n)$	Fixed $\mathsf{poly}(n)$	Yes
[Yam16]: Scheme 1	$O(n\kappa^{\frac{1}{d}} \log^4 n)$	$O(n \log^4 n)$	$n^{\omega(1)}$	Yes
[Yam16]: Scheme 2	$O(n\kappa^{\frac{1}{d}} \log^4 n)$	$O(n \log^4 n)$	All $\mathsf{poly}(n)$	No
Ours: Sec. 4. Type 1.	$O(n\kappa^{\frac{1}{d}} \log^2 n)$	$O(n \log^2 n)$	Fixed $\mathsf{poly}(n)$	Yes
Ours: Sec. 4. Type 2.	$O(n\kappa^{\frac{1}{d}} \log n)$	$O(n \log n)$	Fixed $\mathsf{poly}(n)$	Yes

All parameters presented in the table are obtained by instantiating the schemes in the ring setting. $d \in \mathbb{N}$ is a flexible constant, which can be set to be any value. "$1/\alpha$" for LWE assumption refers to the underlying LWE assumption used in the security reduction. "Fixed $\mathsf{poly}(n)$" means that the corresponding scheme is proven secure under the LWE assumption with $1/\alpha$ being some fixed polynomial (e.g., n^3). "All $\mathsf{poly}(n)$" mean that we have to assume the LWE assumption for all polynomial.

* In the security proof for the adaptively secure variant of IBE in [ABB10], we have a restriction that $q > Q$. Namely, only bounded form of the security is proven. This restriction is removed in the refined analysis due to Boyen [Boy10].

Bilinear Map Based IBE. Here, we compare our scheme in Sec. 5 with other adaptively secure IBE schemes based on the hardness of computational/search problems on bilinear maps in the standard model. To base the security of IBE schemes on such problems, we have to mask the message using the Goldreich-Levin hardcore bit [GL89]. To the best of our knowledge, there are only two IBE schemes that we can apply this modification: Waters IBE [Wat05] and Naccache IBE [Nac07]. As shown in Table 2, our scheme achieves asymptotically shorter master public key size than these schemes. We note that to compare the efficiency, we count the number of group elements. However our method comes at the cost of increasing the ciphertext and private key size and we further have to rely on a stronger assumption than theirs.

Table 2: Comparison of IBE from Bilinear Maps in the Standard Model.

Schemes	$\lvert\mathsf{mpk}\rvert$	$\lvert C\rvert$, $\lvert\mathsf{sk}_{\mathsf{ID}}\rvert$	Assumption
[Wat05] + Hardcore bit [GL89]	$O(\kappa)$	2	CBDH
[Nac07] + Hardcore bit [GL89]	$O(\kappa/\log(\lambda)) = O(\kappa/\log(\kappa))$	2	CBDH
Ours: Sec. 5	$O(\sqrt{\kappa})$	$O(\sqrt{\kappa})$	3-CBDHE

Acknowledgement. We would like to thank anonymous reviewers of Asiacrypt 2016 for helpful comments. We also thank the members of Shin-Akarui-Angou-Benkyoukai for their helpful

discussions and comments. This research was partially supported by CREST, JST. The second author is supported by JSPS KAKENHI Grant Number 16K16068.

References

[ABB10] S. Agrawal, D. Boneh, and X. Boyen. Efficient lattice (H) IBE in the standard model. In *EUROCRYPT*, pp. 553–572. 2010.

[ACPS09] B. Applebaum, D. Cash, C. Peikert, and A. Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In *CRYPTO*, pp. 595–618. 2009.

[Alp15] J. Alperin-Sheriff. Short signatures from homomorphic trapdoor functions. In *PKC*, pp. 236–255. 2015.

[AFL16] D. Apon, X. Fan, and F. Liu. Fully-secure lattice-based IBE as compact as PKE. In *IACR Cryptology ePrint Archive*. 2016:125, 2016.

[BB04a] D. Boneh and X. Boyen. Efficient selective-id secure identity-based encryption without random oracles. In *EUROCRYPT*, pp. 223–238. 2004.

[BB04b] D. Boneh and X. Boyen. Secure identity based encryption without random oracles. In *CRYPTO*, pp. 443–459. 2004.

[BBG05] D. Boneh, X. Boyen, and E. Goh. Hierarchical identity based encryption with constant size ciphertext. In *EUROCRYPT*, pp. 440–456. 2005.

[BF01] D. Boneh and M. Franklin. Identity-based encryption from the weil pairing. In *CRYPTO*, pp. 213–229. 2001.

[BGG+14] D. Boneh, C. Gentry, S. Gorbunov, S. Halevi, V. Nikolaenko, G. Segev, V. Vaikuntanathan, and D. Vinayagamurthy. Fully key-homomorphic encryption, arithmetic circuit ABE and compact garbled circuits. In *EUROCRYPT*, pp. 533–556. 2014.

[BGH07] D. Boneh, C. Gentry, and M. Hamburg. Space-efficient identity based encryption without pairings. In *FOCS*, pp. 647–657. 2007.

[BGW05] D. Boneh, C. Gentry, and B. Waters. Collusion resistant broadcast encryption with short ciphertexts and private keys. In *CRYPTO*, pp. 258–275. 2005.

[BH08] D. Boneh and M. Hamburg. Generalized identity based and broadcast encryption schemes. In *ASIACRYPT*, pp. 455–470. 2008.

[BLL+15] S. Bai, A. Langlois, T. Lepoint, D. Stehlé, and R. Steinfeld. Improved security proofs in lattice-based cryptography: using the rényi divergence rather than the statistical distance. In *ASIACRYPT*, pp. 3–24. 2015.

[Boy10] X. Boyen. Lattice mixing and vanishing trapdoors: A framework for fully secure short signatures and more. In *PKC*, pp. 499–517. 2010.

[BR09] M. Bellare and T. Ristenpart. Simulation without the artificial abort: Simplified proof and improved concrete security for Waters IBE scheme. In *EUROCRYPT*, pp. 407–424. 2009.

[CHKP10] D.=Cash, D. Hofheinz, E. Kiltz, and C. Peikert. Bonsai trees, or how to delegate a lattice basis. *EUROCRYPT*, pp. 523–552. 2010.

[CCZ11] Y. Chen, L. Chen, and Z. Zhang. CCA Secure IB-KEM from computational bilinear Diffie-Hellman in the standard model. In *ICISC*, pp. 275–301. 2011.

[Coc01] C. Cocks. An identity based encryption scheme based on quadratic residues. In *Cryptography and Coding*, pp. 360–363. 2001.

[CW13] J. Chen and H. Wee. Fully,(almost) tightly secure IBE and dual system groups. In *CRYPTO*, pp. 435–460. 2013.

[DM14] L. Ducas and D. Micciancio. Improved short lattice signatures in the standard model. In *CRYPTO*, pp. 335–352. 2014.

[DLP14] L. Ducas, V. Lyubashevsky, and T. Prest. Efficient identity-based encryption over NTRU lattices. In *ASIACRYPT*, pp. 22–41. 2014.

[DM15] L. Ducas and D. Micciancio. Fhew: Bootstrapping homomorphic encryption in less than a second. In *EUROCRYPT*, pp. 617–640. 2015.

[Gal10] D. Galindo. Chosen-ciphertext secure identity-based encryption from computational bilinear Diffie-Hellman. In *Pairing*, pp. 445–464. 2010.

[Gen06] C. Gentry. Practical identity-based encryption without random oracles. In *EUROCRYPT*, pp. 445–464. 2006.

[GL89] O. Goldreich and L. Levin. A hard-core predicate for all one-way functions. In *STOC*, pp. 25–32. 1989.

[GPV08] C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *STOC*, pp. 197–206. 2008.

[HJKS10] K. Haralambiev, T. Jager, E. Kiltz, and V. Shoup: Simple and efficient public-key encryption from computational Diffie-Hellman in the standard model. In *PKC*, pp. 1–18. 2010.

[JR13] C. Jutla and A. Roy. Shorter quasi-adaptive NIZK proofs for linear subspaces. In *ASIACRYPT*, pp. 1–20. 2013.

[LOS+10] A. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters. Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In *EUROCRYPT*, pp. 62–91. 2010.

[LPR10] V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. *EUROCRYPT*, pp. 1–23, 2010.

[LPR13] V. Lyubashevsky, C. Peikert, and O. Regev. A toolkit for ring-lwe cryptography. In *EUROCRYPT*, pp. 35–54. 2013.

[LS15] A. Langlois and D. Stehlé. Worst-case to average-case reductions for module lattices. *DES*, 75(3):565–599, 2015.

[LW10] A. Lewko and B. Waters. New techniques for dual system encryption and fully secure HIBE with short ciphertexts. In *TCC*, pp. 455–479. 2010.

[MP12] D. Micciancio and C. Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In *EUROCRYPT*, pp. 700–718. 2012.

[MR04] D. Micciancio and O. Regev. Worst-case to average-case reductions based on gaussian measures. In *FOCS*, pp. 372–381, 2004.

[Nac07] D. Naccache. Secure and *practical* identity-based encryption. In *IET Information Security*, volume 1(2): pp.. 59–64, 2007.

[Pei10] C. Peikert. An efficient and parallel gaussian sampler for lattices. In *CRYPTO*, pp. 80–97. 2010.

[Reg05] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. In *STOC*, pp. 84–93. ACM Press, 2005.

[Sha85] A. Shamir. Identity-based cryptosystems and signature schemes. In *CRYPTO*, pp. 47–53. 1985.

[SOK00] R. Sakai, K. Ohgishi, and M. Kasahara. Cryptosystems based on pairings. In *SCIS*, 2000. (In Japanese).

[SRB12] K. Singh, C. Pandu Rangan, and A. K. Banerjee. Adaptively secure efficient Lattice (H)IBE in standard model with short public parameters. In *SPACE*, pp.. 153–172, 2012.

[SS11] D. Stehlé and R. Steinfeld. Making NTRU as secure as worst-case problems over ideal lattices. In *EUROCRYPT*, pp. 27–47. 2011.

[SSTX09] D. Stehlé, R. Steinfeld, K. Tanaka, and K. Xagawa. Efficient public key encryption based on ideal lattices. In *ASIACRYPT*, pp. 617–635. 2009.

[Wat05] B. Waters. Efficient identity-based encryption without random oracles. In *EUROCRYPT*, pp. 114–127. 2005.

[Wat09] B. Waters. Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In *CRYPTO*, pp. 619–636. 2009.

[Xag13] K. Xagawa. Improved (hierarchical) inner-product encryption from lattices. In *PKC*, pp. 235–252. 2013.

[Yam16] S. Yamada. Adaptively secure identity-based encryption from lattices with asymptotically shorter public parameters. In *EUROCRYPT*, pp. 32–62. 2016.

[YKHK10] S. Yamada, Y. Kawai, G. Hanaoka, and N. Kunihiro. Public key encryption schemes from the (B)CDH assumption with better efficiency. IEICE Transactions 93-A(11), pp. 1984–1993, 2010.

[ZCZ16] J. Zhang, Y. Chen, and Z. Zhang. Programmable hash functions from lattices: Short signatures and IBEs with small key sizes. In *CRYPTO*. 2016. To appear.

A Supplementary Note on Ring Elements

Useful Formulas. In hope of making the paper more accessible, we provide some formulas on ring elements when viewed as vectors/matrices over $\mathbb{Z}$. Let R denote the polynomial ring $\mathbb{Z}[X]/(\Phi_m(X))$ for m a power of 2 and recall that we can view elements of R as $\mathbb{Z}^n$ through the coefficient embedding $\phi(\cdot)$ and as the subring of anti-circulant marices in $\mathbb{Z}^{n\times n}$ through the ring homomorphism $\mathrm{rot}(\cdot)$. In addition, vectors are viewed in their row forms. All of the following statement holds when we view the polynomial ring $R_q = \mathbb{Z}[X]/(q, \Phi_m(X))$ as $\mathbb{Z}_q$.

First of all, for any element $s \in R$, vectors $\boldsymbol{a}, \boldsymbol{e} \in R^k$ and matrix $\boldsymbol{R} \in R^{k\times\ell}$ recall that we have the following:

$$\begin{aligned} &\phi(s) \in \mathbb{Z}^n, && \phi(\boldsymbol{a}) \in \mathbb{Z}^{nk}, && \\ &\mathrm{rot}(s) \in \mathbb{Z}^{n\times n}, && \mathrm{rot}(\boldsymbol{a}) \in \mathbb{Z}^{n\times nk}, && \mathrm{rot}(\boldsymbol{R}) \in \mathbb{Z}^{nk\times n\ell}. \end{aligned}$$

Then, we obtain the following formulas through simple calculation:

1. $\phi(s\boldsymbol{a}) = \phi(s)\mathrm{rot}(\boldsymbol{a}) \in \mathbb{Z}^{nk}$
2. $\phi(\boldsymbol{a}\boldsymbol{e}^T) = \phi(\boldsymbol{a})\mathrm{rot}(\boldsymbol{e}^T) \in \mathbb{Z}^n$
3. $\phi(\boldsymbol{a}\boldsymbol{R}) = \phi(\boldsymbol{a})\mathrm{rot}(\boldsymbol{R}) \in \mathbb{Z}^{n\ell}$
4. $\mathrm{rot}(\boldsymbol{a}\boldsymbol{R}) = \mathrm{rot}(\boldsymbol{a})\mathrm{rot}(\boldsymbol{R}) \in \mathbb{Z}^{n\times n\ell}$

Gaussian Sampling. The second formula above is mainly used to bridge the gap between the Gaussian sampling algorithms for normal lattices and for ideal lattices (see Sec. 3.3 Lem. 5). Suppose we wish to sample a short vector $\boldsymbol{e} \in R^k$ (from a certain distribution we discuss later) such that $\boldsymbol{a}\boldsymbol{e}^T = u$, where $\boldsymbol{a} \in R^k$ and $u \in R$. Note that this comes up during the KeyGen procedure in our lattice-based construction. Applying the second formula in slightly a different order, we obtain the following:

$$\begin{aligned} & \phi(u) = \phi(\boldsymbol{a}\boldsymbol{e}^T) = \phi(\boldsymbol{e})\mathrm{rot}(\boldsymbol{a}^T) = \left(\mathrm{rot}(\boldsymbol{a}^T)^T\phi(\boldsymbol{e})^T\right)^T \\ \Leftrightarrow \quad & \mathrm{rot}(\boldsymbol{a}^T)^T\phi(\boldsymbol{e})^T = \phi(u)^T \in \mathbb{Z}_q^n. \end{aligned}$$

Note that in general $\mathrm{rot}(\boldsymbol{a}) \neq \mathrm{rot}(\boldsymbol{a}^T)^T$. Therefore, we only have to sample a vector $\mathbf{e} \in \mathbb{Z}^{nk}$ from the coset $\Lambda^{\perp}_{\phi(\boldsymbol{u})}(\mathrm{rot}(\boldsymbol{a}^T)^T)$ and map it back to its ring representation $\boldsymbol{e} = \phi^{-1}(\mathbf{e}) \in R^k$ to obtain a short sample $\boldsymbol{e}$ such that $\boldsymbol{a}\boldsymbol{e}^T = u$. This can be done easily by using a basis $\mathrm{rot}(\boldsymbol{T_a})$ for the lattice $\Lambda^{\perp}(\mathrm{rot}(\boldsymbol{a}^T)^T)$.

B Omitted Details/Proofs from Section 3

B.1 Proof of Lemma 1

Before proving Lemma 1 on noise rerandomization, we recall the following two lemmas. Note that Lemma 17, the special case of the claim from [Reg05], is restated in order to make the comparison between Lemma 16 more clear.

Lemma 16 ([Pei10], Special Case of Theorem 3.1)**.** *Let n be a positive integer and r be a positive real satisfying $r \geq \omega(\sqrt{\log n})$. Then, if we choose $\mathbf{x}_1$ from the continuous Gaussian D_r^n and then choose $\mathbf{x}_2$ from the discrete Gaussian $D_{\mathbb{Z}^n-\mathbf{x}_1,r}$, then $\mathbf{x}_1+\mathbf{x}_2$ is distributed statistically close to the discrete Gaussian $D_{\mathbb{Z}^n,\sqrt{2}r}$.*

Lemma 17 ([Reg05], Special Case of Claim 3.9)**.** *Let n be a positive integer and let r a positive real satisfying $r \geq \omega(\sqrt{\log n})$. Then, if we choose $\mathbf{x}_1$ from the continuous Gaussian D_r^n and choose $\mathbf{x}_2$ from the discrete Gaussian $D_{\mathbb{Z}^n,r}$, $\mathbf{x}_1+\mathbf{x}_2$ is distributed statistically close to the continuous Gaussian $D_{\sqrt{2}r}^n$.*

Then, the proof of Lemma 1 is given as follows.

Proof. The algorithm samples $\mathbf{c}, \mathbf{d}$ and $\mathbf{f}$ as follows:

1. sample $\mathbf{c}$ from the continuous Gaussian distribution D_r^m,
2. sample $\mathbf{d}$ from the continuous Gaussian distribution $D_{\sqrt{2}r(\sigma^2\mathbf{I}_\ell-\mathbf{V}^T\mathbf{V})^{1/2}}$,
3. sample $\mathbf{f}$ from the discrete Gaussian $D_{\mathbb{Z}^\ell-(\mathbf{cV}+\mathbf{d}),\sqrt{2}r\sigma}$.

Observe the distribution of $\mathbf{d}$ is well-defined. The algorithm outputs the following,

$$\mathbf{b}' = ((\mathbf{b}+\mathbf{x})+\mathbf{c})\,\mathbf{V}+\mathbf{d}+\mathbf{f} = \mathbf{bV} + \underbrace{(\mathbf{x}+\mathbf{c})\mathbf{V}+\mathbf{d}+\mathbf{f}}_{\mathbf{x}':=\text{“noise term”}} \in \mathbb{Z}_q^\ell.$$

We analyse the noise term and show that it is distributed as in the statement. Let $\mathbf{x}' = (\mathbf{x}+\mathbf{c})\mathbf{V}+\mathbf{d}+\mathbf{f}$. Observe that by Lemma 17, $\mathbf{x}+\mathbf{c}$ is distributed as the continuous Gaussian distribution $D_{\sqrt{2}r}^m$. Therefore, $(\mathbf{x}+\mathbf{c})\mathbf{V}$ is distributed as the distribution $D_{\sqrt{2}r\mathbf{V}^T}$. Since $\mathbf{d}$ is sampled from the continuous Gaussian distribution $D_{\sqrt{2}r(\sigma^2\mathbf{I}_\ell-\mathbf{V}^T\mathbf{V})^{1/2}}$, it follows that $\mathbf{y} = (\mathbf{x}+\mathbf{c})\mathbf{V}+\mathbf{d}$ is distributed as a spherical continuous Gaussian $D_{\sqrt{2}r\sigma}^\ell$. Next, observe that since $\mathbf{xV} \in \mathbb{Z}^\ell$, the two distributions $D_{\mathbb{Z}^\ell-\mathbf{y},\sqrt{2}r\sigma}$ and $D_{\mathbb{Z}^\ell-(\mathbf{cV}+\mathbf{d}),\sqrt{2}r\sigma}$ are equivalent by definition. Therefore, by Lemma 16, adding $\mathbf{f}$ chosen from the discrete Gaussian $D_{\mathbb{Z}^\ell-(\mathbf{cV}+\mathbf{d}),\sqrt{2}r\sigma}$ to $\mathbf{y}$, which we can do without knowledge of the unknown value $\mathbf{x}$, we can discretize $\mathbf{y}$. Hence $\mathbf{x}' = \mathbf{y}+\mathbf{f}$ is distributed according to the discrete Gaussian $D_{\mathbb{Z}^\ell,2r\sigma}$ as in the above statement. □

B.2 Proof of Lemma 3

Proof. The first part of the lemma is taken from Lemma 2.3 of [SSTX09]. Therefore, we only prove the latter part of the lemma, which is implicit in [SS11]. If $x \notin R_q^*$, $x \in \langle t_1 \rangle$ or $x \in \langle t_2 \rangle$ holds over R_q. We assume that the former holds without loss of generality. Then, $t \in \langle t_1, q \rangle$ holds over R. Thus, $\mathcal{N}(x) = \mathcal{N}(\langle x \rangle) \geq \mathcal{N}(\langle t_1, q \rangle) = q^{n/2}$, where $\mathcal{N}$ is the (field) norm. (See [SS11] for the definition.) Then, by using the additive geometric mean it can be seen that $\|\sigma(x)\|_2 = \sqrt{\sum_{i=1}^n |\sigma_i(x)|^2} \geq \sqrt{n} \cdot \sqrt[2n]{\prod_{i=1}^n |\sigma_i(x)|^2} = \sqrt{n} \cdot \sqrt[n]{\mathcal{N}(x)} \geq \sqrt{nq}$ holds. Since $\|\sigma(x)\|_2 = \sqrt{n}\|\phi(x)\|_2$, the statement follows. □

B.3 Proof of Lemma 4

Proof. We first show the former part of the lemma. Let $\boldsymbol{x}_1 \neq \boldsymbol{x}_2 \in R_q^{k\times 1}$ be arbitrary elements in $[-\rho,\rho]_R^k$ and set $\boldsymbol{z} = \boldsymbol{x}_1 - \boldsymbol{x}_2 \in R_q^{k\times 1}$. Then we have $\boldsymbol{z} \in [-2\rho, 2\rho]_R^k$. Assume for some $\boldsymbol{A} \in R_q^{k'\times k}$, we have $h_{\boldsymbol{A}}(\boldsymbol{x}_1) = h_{\boldsymbol{A}}(\boldsymbol{x}_2)$, i.e., $h_{\boldsymbol{A}}(\boldsymbol{z}) = 0$. Since, $\boldsymbol{x}_1 \neq \boldsymbol{x}_2$, there exists $j \in [k]$ such that the

jth coefficient of $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are different. Then, by Lemma 3, since $\|\phi(z_j)\|_2 \le 2\rho\sqrt{n} < \sqrt{q}$, z_j must be invertible. Therefore, $\boldsymbol{a}_j = z_j^{-1}\sum_{i\neq j} z_i\boldsymbol{a}_i$ where $\boldsymbol{a}_i \in R_q^{k'\times 1}$ is the ith column of $\boldsymbol{A}$. The probability of a random $\boldsymbol{A} \in R_q^{k'\times k}$ satisfying this condition is exactly $1/q^{nk'} = 1/|R_q|^{k'}$. Hence $\mathcal{H}$ is universal. We then show the latter part of the lemma. We observe that the case of $\ell = 1$ follows from the leftover hash lemma since the min-entropy of $\boldsymbol{X}$ is $(1/(2\rho+1))^{kn}$ in this case. The case of $\ell \ge 2$ immediately follows from a standard hybrid argument. □

B.4 Correctness of TrapGen in Lemma 5

Proof. The proof follows by combining several Lemmas from [MP12] and our Lemma 2 and Lemma 4. First for simplicity asssume k is even, i.e., $k = 2k'$ for some $k' \in \mathbb{N}$, and assume that $k' = \lceil \log_b q \rceil$ for some positive integer b. We first show that $\boldsymbol{a} = [\boldsymbol{a}'|\boldsymbol{g}_b - \boldsymbol{a}'\boldsymbol{R}]$ is distributed uniformly at random over R^k when $\boldsymbol{a}' \xleftarrow{\$} R^{k'}$ and $\boldsymbol{R} \xleftarrow{\$} [-\rho,\rho]_R^{k'\times k'}$. This follows from Lemma 4, since we have

$$\frac{k'}{2}\sqrt{\left(\frac{q}{(2\rho+1)^{k'}}\right)^n} \le \frac{k'}{2}\left(\frac{q}{(2\rho)^{k'}}\right)^{\frac{n}{2}} \le \frac{k'}{2}\left(\frac{1}{2^{k'}}\right)^{\frac{n}{2}} \le \frac{k'}{2^{n+1}} = \mathsf{negl}(n),$$

when $1 < \rho < \frac{1}{2}\sqrt{q/n}$, $k' \ge \log_\rho q$ and k' is polynomial in n. Similar result holds for the case $\rho = 1$. Note that in the case of $\rho = 1$, we define $\log_1 q := \log_2 q$. Next, by the property of $\boldsymbol{g}_b$ there exists a publicly known basis $\boldsymbol{T}_{\boldsymbol{g}_b} \in R^{k'\times k'}$ such that $\mathrm{rot}(\boldsymbol{T}_{\boldsymbol{g}_b})$ is a basis for $\Lambda^\perp(\mathrm{rot}(\boldsymbol{g}_b^T)^T)$ (or equivallently for $\Lambda^\perp(\mathrm{rot}(\boldsymbol{g}_b))$) such that $\|\mathrm{rot}(\boldsymbol{T}_{\boldsymbol{g}_b})\|_{\mathsf{GS}} \le \sqrt{b^2+1}$. We also have $s_1(\boldsymbol{R}) \le O(\rho\cdot\sqrt{nk'})$ with all but negligible probability from Lemma 2. Then using the fact that $\mathrm{rot}(\boldsymbol{R}^T)^T$ (resp. $\mathrm{rot}(\boldsymbol{R})$) is a $\boldsymbol{g}_b$-trapdoor for $\mathrm{rot}(\boldsymbol{a}^T)^T$ (resp. $\mathrm{rot}(\boldsymbol{a})$) and by combining the ring version of Theorem 4.1 and Lemma 5.3 from [MP12], we obtain a basis $\boldsymbol{T}_{\boldsymbol{a}}$ such that $\|\mathrm{rot}(\boldsymbol{T}_{\boldsymbol{a}})\|_{\mathsf{GS}} = O(b\rho\cdot\sqrt{n\log_\rho q})$. Note that we obtain bases for both $\Lambda^\perp(\mathrm{rot}(\boldsymbol{a}^T)^T)$ and $\Lambda^\perp(\mathrm{rot}(\boldsymbol{a}))$ from $\boldsymbol{T}_{\boldsymbol{a}}$ by properly rearanging $\boldsymbol{T}_{\boldsymbol{a}}$. □

B.5 Proof of Lemma 6

Before proving the lemma, we state the following lemma that provides us with a useful bound for the singular value of a single element in R.

Lemma 18 ([DM14], Lemma 5)**.** *For any ring element $a \in R$, we have $s_1(a) \le \|\phi(a)\|_1$.*

Then, the proof of Lemma 6 is given as follows.

Proof. We prove it by induction. The base case (the case of $d = 1$) is trivial. Therefore, let us assume the hypothesis for $d-1$ where $d \ge 2$. Then, we have

$$s_1(\boldsymbol{R}_0) \le B\delta^{d-2} + Bbnk\left(\frac{\delta^{d-2}-1}{\delta-1}\right) \quad \text{and} \quad \mathsf{PubEval}_{d-1}(\boldsymbol{b}_2,\ldots,\boldsymbol{b}_d) = \boldsymbol{a}\boldsymbol{R}_0 + y_2\cdots y_d\boldsymbol{g}_b$$

for efficiently computable $\boldsymbol{R}_0$. Therefore, by the definition of $\mathsf{PubEval}_d$, we have

$$\begin{aligned}
& \mathsf{PubEval}_d(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_d) \\
= \;& (\boldsymbol{a}\boldsymbol{R}_1 + y_1\boldsymbol{g}_b)\cdot\boldsymbol{g}_b^{-1}\big(\mathsf{PubEval}_{d-1}(\boldsymbol{b}_2,\ldots,\boldsymbol{b}_d)\big) \\
= \;& \boldsymbol{a}\boldsymbol{R}_1\cdot\boldsymbol{g}_b^{-1}\big(\mathsf{PubEval}_{d-1}(\boldsymbol{b}_2,\ldots,\boldsymbol{b}_d)\big) + y_1\cdot\mathsf{PubEval}_{d-1}(\boldsymbol{b}_2,\ldots,\boldsymbol{b}_d) \\
= \;& \boldsymbol{a}\boldsymbol{R}_1\cdot\boldsymbol{g}_b^{-1}\big(\mathsf{PubEval}_{d-1}(\boldsymbol{b}_2,\ldots,\boldsymbol{b}_d)\big) + y_1(\boldsymbol{a}\boldsymbol{R}_0 + y_2\cdots y_d\boldsymbol{g}_b)
\end{aligned}$$

$$= \boldsymbol{a}(\boldsymbol{R}_1 \cdot \boldsymbol{g}_b^{-1}(\mathsf{PubEval}_{d-1}(\boldsymbol{b}_2, \ldots, \boldsymbol{b}_d)) + y_1 \boldsymbol{R}_0) + y_1 y_2 \cdots y_d \boldsymbol{g}_b.$$

It can be seen that Eq.(12) holds by setting

$$\boldsymbol{R}' = \boldsymbol{R}_1 \cdot \boldsymbol{g}_b^{-1}(\mathsf{PubEval}_{d-1}(\boldsymbol{b}_2, \ldots, \boldsymbol{b}_d)) + y_1 \boldsymbol{R}_0.$$

It is clear that it can be efficiently computable. Furthermore, we have

$$\begin{aligned} s_1(\boldsymbol{R}') &\leq s_1(\boldsymbol{R}_1) \cdot s_1\Big(\boldsymbol{g}_b^{-1}(\mathsf{PubEval}_{d-1}(\boldsymbol{b}_2, \ldots, \boldsymbol{b}_d))\Big) + s_1(y_1) \cdot s_1(\boldsymbol{R}_0) \\ &\leq B \cdot bnk + \|\phi(a)\|_1 \cdot s_1(\boldsymbol{R}_0) \\ &\leq Bbnk + \delta\left(B\delta^{d-2} + Bbnk\left(\frac{\delta^{d-2}-1}{\delta-1}\right)\right) \\ &= B\delta^{d-1} + Bbnk\left(\frac{\delta^{d-1}-1}{\delta-1}\right). \end{aligned}$$

The second inequality follows from Lemma 18 and the fact that $s_1(\boldsymbol{g}_b^{-1}(\boldsymbol{u})) \leq bnk$ holds for any $\boldsymbol{u} \in R_q^k$. □

B.6 Proof of Lemma 7

Proof. We have

$$\begin{aligned} \|\phi(uv)\|_\infty &= \left\| \phi\left(\sum_{j=0}^{c_1+c_2-2} \left(\sum_{i=\max\{0,j+1-c_2\}}^{\min\{c_1-1,j\}} u_i v_{j-i} \right) X^j \right) \right\|_\infty \\ &= \max_{j \in [0, c_1+c_2-2]} \left\{ \sum_{i=\max\{0,j+1-c_2\}}^{\min\{c_1-1,j\}} u_i v_{j-i} \right\} \\ &\leq \min\{c_1, c_2\} B_1 B_2 \end{aligned}$$

where the last equation follows from $\|\phi(u)\|_\infty \leq B_1$, $\|\phi(v)\|_\infty \leq B_2$, and $\min\{c_1-1, j\} + 1 - \max\{0, j+1-c_2\} \leq \min\{(c_1-1)+1-0, j+1-(j+1-c_2)\} = \min\{c_1, c_2\}$.

□

B.7 Proof of Lemma 8

Proof. For $\mathbb{ID} = (\mathsf{ID}^\star, \mathsf{ID}_1, \ldots, \mathsf{ID}_Q)$, we define $\mathsf{Q}(\mathbb{ID})$ as the event that $\mathcal{A}$ chooses $\mathsf{ID}^\star$ as its challenge identity and it makes key extraction queries for $\mathsf{ID}_1, \ldots, \mathsf{ID}_Q$. We also define $\mathsf{Replace}$ as the event that coin' is set as $\mathsf{coin}' \xleftarrow{\$} \{0,1\}$. Then, we have

$$\begin{aligned} & \left| \Pr[\mathsf{coin}' = \mathsf{coin}] - \frac{1}{2} \right| \\ = & \left| \sum_{\mathbb{ID}} \Pr[\mathsf{Q}(\mathbb{ID})] \cdot \Pr[\mathsf{coin}' = \mathsf{coin} | \mathsf{Q}(\mathbb{ID})] - \frac{1}{2} \right| \qquad (66) \\ = & \left| \sum_{\mathbb{ID}} \Pr[\mathsf{Q}(\mathbb{ID})] \cdot \Big(\Pr[\mathsf{coin}' = \mathsf{coin} \wedge \neg \mathsf{Replace} | \mathsf{Q}(\mathbb{ID})] \right. \end{aligned}$$

$$+\Pr[\mathsf{coin}' = \mathsf{coin} \wedge \mathsf{Replace} | \mathsf{Q}(\mathbb{ID})] - \frac{1}{2}\Big)\Big| \tag{67}$$

$$= \left|\sum_{\mathbb{ID}} \Pr[\mathsf{Q}(\mathbb{ID})] \cdot \left(\Pr[\widehat{\mathsf{coin}} = \mathsf{coin} | \mathsf{Q}(\mathbb{ID})] \cdot \gamma(\mathbb{ID}) + \frac{1}{2} \cdot \big(1 - \gamma(\mathbb{ID})\big) - \frac{1}{2}\right)\right| \tag{68}$$

$$= \left|\sum_{\mathbb{ID}} \gamma(\mathbb{ID}) \cdot \Pr[\mathsf{Q}(\mathbb{ID})] \cdot \left(\Pr[\widehat{\mathsf{coin}} = \mathsf{coin} | \mathsf{Q}(\mathbb{ID})] - \frac{1}{2}\right)\right| \tag{69}$$

$$\geq \left|\sum_{\mathbb{ID}} \gamma_{\min} \cdot \Pr[\mathsf{Q}(\mathbb{ID})] \cdot \left(\Pr[\widehat{\mathsf{coin}} = \mathsf{coin} | \mathsf{Q}(\mathbb{ID})] - \frac{1}{2}\right)\right| - \left|\sum_{\mathbb{ID}} (\gamma(\mathbb{ID}) - \gamma_{\min}) \cdot \Pr[\mathsf{Q}(\mathbb{ID})] \cdot \left(\Pr[\widehat{\mathsf{coin}} = \mathsf{coin} | \mathsf{Q}(\mathbb{ID})] - \frac{1}{2}\right)\right| \tag{70}$$

$$\geq \gamma_{\min} \left|\sum_{\mathbb{ID}} \Pr[\mathsf{Q}(\mathbb{ID})] \cdot \left(\Pr[\widehat{\mathsf{coin}} = \mathsf{coin} | \mathsf{Q}(\mathbb{ID})] - \frac{1}{2}\right)\right| - \frac{\gamma_{\max} - \gamma_{\min}}{2} \left|\sum_{\mathbb{ID}} \Pr[\mathsf{Q}(\mathbb{ID})]\right| \tag{71}$$

$$= \gamma_{\min} \left|\Pr[\widehat{\mathsf{coin}} = \mathsf{coin}] - \frac{1}{2}\right| - \frac{\gamma_{\max} - \gamma_{\min}}{2} \tag{72}$$

$$= \gamma_{\min} \cdot \epsilon - \frac{\gamma_{\max} - \gamma_{\min}}{2} \tag{73}$$

where the sum is taken over all possible $\mathbb{ID}$ (i.e., $\mathbb{ID}$ with $\mathsf{Q}(\mathbb{ID}) > 0$). In the above, Eq.(66) follows by the law of total probability, Eq.(67) follows from the law of total probability and $\sum_{\mathbb{ID}} \Pr[\mathsf{Q}(\mathbb{ID})] = 1$, Eq.(68) follows from the fact that the probability of $\mathsf{Replace}$ is $\gamma(\mathbb{ID})$, when conditioned on $\mathsf{Q}(\mathbb{ID})$ (regardless of the value of $\widehat{\mathsf{coin}}$), Eq.(69) is trivial, Eq.(70) follows from the triangle inequality, Eq.(71) holds since $\gamma(\mathbb{ID}) \leq \gamma_{\max}$ and $|\Pr[\widehat{\mathsf{coin}} = \mathsf{coin} | \mathsf{Q}(\mathsf{ID})] - 1/2| \leq 1/2$, Eq.(72) follows again from $\sum_{\mathbb{ID}} \Pr[\mathsf{Q}(\mathbb{ID})] = 1$, and Eq.(73) is by the definition of ϵ. □

C Correctness of the Decryption Algorithm in Sec 4

Here, we prove Lemma 10, which gives a sufficient condition for the correctness of the decryption algorithm in our scheme in Sec. 4. Before proving Lemma 10, we prepare the following two lemmas.

Lemma 19 ([MR04], Lemma 4.4)**.** *For any n-dimensional lattice Λ, real $\epsilon \in (0,1)$ and $s \geq \eta_\epsilon(\Lambda)$, we have $\Pr[\|\mathbf{x}\| > s\sqrt{n} | \ \mathbf{x} \leftarrow D_{\Lambda, s\omega(\sqrt{\log n})}] \leq \frac{1+\epsilon}{1-\epsilon} \cdot 2^{-n}$.*

The following is an analogue of [ABB10], Lemma 12 where the error is instead chosen from the discrete Gaussian.

Lemma 20 (Discrete Gaussian Error Bound)**.** *Let $\mathbf{e}$ be some vector in $\mathbb{Z}^n$ and let $\mathbf{x} \leftarrow D_{\mathbb{Z}^n, \alpha q}$ for some $\alpha q > \omega(\sqrt{\log n})$. Then the quantity $|\mathbf{e}\mathbf{x}^T|$ treated as an integer in $[0, \ldots, q-1]$ satisfies $|\mathbf{e}\mathbf{x}^T| \leq \|\mathbf{e}\|_2 \alpha q \omega(\sqrt{\log n})$ with all but negligible probability in n.*

Proof. (of Lemma 20.) By [MP12], Lemma 2.8, each element of x_i are δ-subgaussian of parameter αq, where $\delta > 0$ is negligible in n. Then the random variable $\mathbf{e}\mathbf{x}^T$ is $n\delta$-subgaussian with parameter $\|\mathbf{e}\|_2 \alpha q$. Hence by the subgaussian distribution tail bound, we have $\Pr[|\mathbf{e}\mathbf{x}^T| > \|\mathbf{e}\|_2 \alpha q \omega(\sqrt{\log n})] \leq \mathsf{negl}(n)$, which proves the lemma. □

Then, the proof of Lemma 10 is given as follows.

Proof. When the Decrypt algorithm operates as specified for a valid encryption of message $\mathsf{M} \in \{0,1\}^n \subset R$, we have

$$\phi(c_0 - \boldsymbol{c}_1 \boldsymbol{e}^T) = \lfloor \frac{q}{2} \rceil \phi(\mathsf{M}) + \underbrace{\phi(x_0) - \phi([\boldsymbol{x}_1|\boldsymbol{x}_2])\mathrm{rot}(\boldsymbol{e}^T)}_{\text{error term}},$$

Hence, for the Decrypt algorithm to output M, we need to show that the error term does not exceed, say $q/5$. Since $x_0 \xleftarrow{\$} D^{\mathsf{coeff}}_{\mathbb{Z}^n,\alpha q}$, the vector $\phi(x_0)$ is a subgaussian with parameter αq, i.e., $D_{\mathbb{Z}^n,\alpha q}$. Therefore, by the standard subgaussian tail bound argument, $|\phi(x_0)_j| \leq \alpha q \omega(\sqrt{\log n})$ with all but negligible probability, where $\phi(x_0)_j$ denotes the jth entry. Furthermore, since $\boldsymbol{x}_1, \boldsymbol{x}_2 \xleftarrow{\$} (D^{\mathsf{coeff}}_{\mathbb{Z}^n,\alpha'})^k$, we have that $\phi([\boldsymbol{x}_1|\boldsymbol{x}_2]) \xleftarrow{\$} D_{\mathbb{Z}^{2nk},\alpha'}$. From the definition of the map rot, we have that each column of $\mathrm{rot}(\boldsymbol{e}^T) \in \mathbb{Z}^{2nk \times n}$ is of norm $\|\phi(\boldsymbol{e})\|_2$. Hence, by Lemma 19, Lemma 20 and from the fact that $\phi(\boldsymbol{e}) \xleftarrow{\$} D_{\Lambda^{\perp}_{\phi(u)}([\mathrm{rot}(a^T)^T|\mathrm{rot}(\mathsf{H}(\mathsf{ID})^T)^T]),\sigma}$, we have $|\phi([\boldsymbol{x}_1|\boldsymbol{x}_2])\mathrm{rot}(\boldsymbol{e}^T)_j| \leq \|\phi(\boldsymbol{e})\|_2 \cdot \alpha' \omega(\sqrt{\log nk}) \leq \sqrt{nk}\alpha'\sigma\omega(\sqrt{\log nk})$ with all but negligible probability, where $\mathrm{rot}(\boldsymbol{e}^T)_j$ denotes the jth column.

Putting all the pieces together, we conclude that the jth entry of the error term is bounded as

$$\left| \left(\phi(x_0) - \phi([\boldsymbol{x}_1|\boldsymbol{x}_2])\mathrm{rot}(\boldsymbol{e}^T)\right)_j \right| \leq \alpha q \omega(\sqrt{\log n}) + \sqrt{nk}\alpha'\sigma\omega(\sqrt{\log nk}),$$

with all but negligible probability. By assumption this is smaller than $q/5$ with overwhelming probability. Hence, the error probability for the Decrypt algorithm is negligible. □

D Further Details on IBEs from Bilinear Maps

D.1 Multi-bit Variant

Let us try to extend our single-bit scheme in Sec. 5 to be a multi-bit scheme with message space $\{0,1\}^{\ell_M}$ for some $\ell_M \in \mathbb{N}$. The most obvious way to achieve this is to just run the encryption algorithm ℓ_M times. However, this naive method will make the ciphertext ℓ_M times longer. Another way would be to prepare ℓ_M copies of g^α and h^β and put them into the master public key. However, this approach will result in a scheme with master public key containing extra $O(\ell_M)$ group elements. In this section, we show that it is possible to obtain a multi-bit scheme with the same ciphertext-size as the single-bit scheme, by adding only $O(\sqrt{\ell_M})$ group elements to the master public key. This can be accomplished by incorporating our single bit scheme in Sec. 5 with the technique from [HJKS10, YKHK10].

For simplicity, we assume that $\ell_M = (\ell')^2$ for some $\ell' \in \mathbb{N}$ in the following.

$\mathsf{Setup}(1^\lambda)$: On input 1^λ, it chooses an asymmetric bilinear group $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ with efficiently computable map $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ of prime order $p = p(\lambda)$. Let g and h be generator of $\mathbb{G}_1$ and $\mathbb{G}_2$ respectively. It then picks $w_0, w_{1,1}, \ldots, w_{1,\ell}, w_{2,1}, \ldots, w_{2,\ell}, \alpha_1, \ldots \alpha_{\ell'}, \beta_1, \ldots, \beta_{\ell'} \xleftarrow{\$} \mathbb{Z}_p$ and $\mathsf{rand} \xleftarrow{\$} \{0,1\}^{|\mathbb{G}_T|}$. It finally outputs

$$\begin{aligned} \mathsf{mpk} &= (g, W_0 = g^{w_0}, \{W_{i,j} = g^{w_{i,j}}\}_{(i,j)\in[2]\times[\ell]}, \{g^{\alpha_i}\}_{i=1}^{\ell'}, \{g^{\beta_i}\}_{i=1}^{\ell'}, \mathsf{rand}) \quad \text{and} \\ \mathsf{msk} &= (h, \{\alpha_i\}_{i\in[\ell']}, \{\beta_i\}_{i\in[\ell']}, w_0, w_{1,1}, \ldots, w_{1,\ell}, w_{2,1}, \ldots, w_{2,\ell}) \end{aligned}$$

$\mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, \mathsf{ID})$: It first computes $\mathsf{H}(\mathsf{ID})$ (defined as Eq.(48)) using msk and picks $r^{(i,j)} \xleftarrow{\$} \mathbb{Z}_p$ for $(i,j) \in [\ell'] \times [\ell']$. It then computes

$$\mathsf{sk}_{\mathsf{ID}}^{(i,j)} = \left(A_1^{(i,j)} = h^{\alpha_i \beta_j + r^{(i,j)} \cdot \mathsf{H}(\mathsf{ID})},\ A_2^{(i,j)} = h^{-r^{(i,j)}},\ \{B_k^{(i,j)} = h^{r^{(i,j)} w_{2,k}}\}_{k=1}^{\ell} \right)$$

for $(i,j) \in [\ell'] \times [\ell']$. It then outputs $\mathsf{sk}_{\mathsf{ID}} = \{\mathsf{sk}_{\mathsf{ID}}^{(i,j)}\}_{(i,j) \in [\ell'] \times [\ell']}$.

$\mathsf{Encrypt}(\mathsf{mpk}, \mathsf{ID}, \mathsf{M})$: To encrypt a message $\mathsf{M} = \{0,1\}^{\ell_M}$, it picks $s, t_1, \ldots, t_\ell \xleftarrow{\$} \mathbb{Z}_p$ and computes

$$C_1 = g^s,\ C_2 = W_0^s \cdot \prod_{j \in [1,\ell]} W_{2,j}^{t_j},\ D_j = g^{t_j} \cdot \left(\prod_{i \in \{i \in [1,\ell] | (i,j) \in S(\mathsf{ID})\}} W_{1,i} \right)^{-s} \text{ for } j \in [1,\ell]$$

It also computes $e((g^{\alpha_i})^s, h^{\beta_j}) = e(g,h)^{s\alpha_i\beta_j}$ and sets

$$\mathsf{K}^{(i,j)} = \mathsf{GL}(e(g,h)^{s\alpha_i\beta_j}, \mathsf{rand})$$

for all $(i,j) \in [\ell', \ell']$. It then sets $\mathsf{K} = \mathsf{K}^{(1,1)} \| \mathsf{K}^{(1,2)} \| \cdots \| \mathsf{K}^{(\ell',\ell')}$ and $C_0 = \mathsf{K} \oplus \mathsf{M}$. Finally, it returns the ciphertext $C = (C_0, C_1, C_2, \{D_j\}_{j=1}^{\ell})$.

$\mathsf{Decrypt}(\mathsf{mpk}, \mathsf{sk}_{\mathsf{ID}}, C)$: To decrypt a ciphertext $C = (C_0, C_1, C_2, \{D_j\}_{j=1}^{\ell})$ using a private key $\mathsf{sk}_{\mathsf{ID}} = (\{A_1^{(i,j)}, A_2^{(i,j)}, \{B_k^{(i,j)}\}_{k=1}^{\ell}\}_{(i,j) \in [\ell'] \times [\ell']})$, it first computes

$$e(C_1, A_1^{(i,j)}) \cdot e(C_2, A_2^{(i,j)}) \cdot \prod_{k \in [1,\ell]} e(D_j, B_k^{(i,j)}) = e(g,h)^{s\alpha_i\beta_j}.$$

for $(i,j) \in [\ell'] \times [\ell']$. Then it sets $\mathsf{K}^{(i,j)} = \mathsf{GL}(e(g,h)^{s\alpha_i\beta_j}, \mathsf{rand})$ and $\mathsf{K} = \mathsf{K}^{(1,1)} \| \mathsf{K}^{(1,2)} \| \cdots \| \mathsf{K}^{(\ell',\ell')}$. Finally, it retrieves the message by $C_0 \oplus \mathsf{K} = \mathsf{M}$.

Correctness of the scheme can be checked similarly to the single-bit version in Sec. 5.

D.2 Security of the Multi-bit Variant

Security of the multi-bit scheme is reduced to the security of a certain variant of the single-bit scheme. Concretely, we consider a variant of our single-bit scheme with the master public key being changed to

$$\mathsf{mpk} = (g, W_0 = g^{w_0}, \boxed{h^{w_0}}, \{W_{i,j} = g^{w_{i,j}}\}_{(i,j) \in [2] \times [\ell]}, \boxed{\{h^{w_{2,i}}\}_{i=1}^{\ell}}, g^{\alpha}, h^{\beta}, \boxed{\{h^{w_{1,i} w_{2,j}}\}_{(i,j) \in [\ell] \times [\ell]}}, \mathsf{rand})$$

Namely, we add h^{w_0}, $\{h^{w_{2,i}}\}_{i \in [\ell]}$, and $\{h^{w_{1,i}w_{2,j}}\}_{(i,j) \in [\ell] \times [\ell]}$ to mpk. The rest of the scheme is unchanged. We call the scheme "single bit scheme with redundant key". We claim that the security of this scheme can also be proven under the 3-CBDHE assumption with almost an identical proof to that of Theorem 3. The only place where we need to change is Lemma 15. Here, we have to simulate the above additional terms. In fact, this can easily be done using the problem instance of the 3-CBDHE assumption, since we have

$$h^{w_0} = (h^{a^2})^{y_0} h^{\tilde{w}_0}, \quad h^{w_{2,i}} = (h^a)^{y_{2,i}} h^{\tilde{w}_{2,i}}, \quad h^{w_{1,i}w_{2,j}} = (h^{a^2})^{y_{1,i}y_{2,j}} \cdot (h^a)^{y_{1,i}\tilde{w}_{2,j} + y_{2,j}\tilde{w}_{1,i}} \cdot h^{\tilde{w}_{1,i}\tilde{w}_{2,j}}.$$

Summing up the above discussion, we have the following theorem.

Theorem 4. *The single-bit scheme with redundant key is adaptively secure under the 3-CBDHE assumption.*

Therefore, to prove the security of our multi-bit variant, it suffices to show the following.

Theorem 5. *Assuming the single-bit scheme with redundant key is adaptively secure, so is the multi-bit scheme.*

Proof. Let $\mathcal{A}$ be a PPT adversary that breaks the adaptive security of the scheme. To prove the theorem, we consider the following hybrid games for $(i,j) \in \{(1,0)\} \cup ([\ell'] \times [\ell'])$. For convenience, we will denote $(i, \ell'+1) := (i+1, 1)$ and $(i, 0) := (i-1, \ell')$.

$\mathsf{Game}^{(i,j)}$: This is the real game except that the challenger encrypts a message

$$\mathsf{M}_1^{(1,1)} \| \mathsf{M}_1^{(1,2)} \| \cdots \| \mathsf{M}_1^{(i,j)} \| \mathsf{M}_0^{(i,j+1)} \| \cdots \| \mathsf{M}_0^{(\ell',\ell')}$$

where $\mathsf{M}_b^{(i,j)}$ denotes the $(i-1)\ell' + j$th bit of M_b for $b \in \{0,1\}$.

It can be seen that $\mathsf{Game}^{(1,0)}$ corresponds to the case of $\mathsf{coin} = 0$ (M_0 is always encrypted) and $\mathsf{Game}^{(\ell',\ell')}$ corresponds to the case of $\mathsf{coin} = 1$ (M_1 is always encrypted). We denote the event that $\mathcal{A}$ outputs 1 in $\mathsf{Game}^{(i,j)}$ be $X^{(i,j)}$. We have

$$\begin{aligned}
\left| \Pr[\widehat{\mathsf{coin}} = \mathsf{coin}] - \frac{1}{2} \right| &= \left| \frac{1}{2} \Pr[\widehat{\mathsf{coin}} = 1 | \mathsf{coin} = 1] + \frac{1}{2} \Pr[\widehat{\mathsf{coin}} = 0 | \mathsf{coin} = 0] - \frac{1}{2} \right| \\
&= \left| \frac{1}{2} \Pr[\widehat{\mathsf{coin}} = 1 | \mathsf{coin} = 1] - \frac{1}{2} \Pr[\widehat{\mathsf{coin}} = 1 | \mathsf{coin} = 0] \right| \\
&= \frac{1}{2} \left| \Pr[X^{(1,0)}] - \Pr[X^{(\ell',\ell')}] \right| \\
&= \frac{1}{2} \left| \sum_{(i,j) \in [\ell'] \times [\ell']} \Pr[X^{(i,j-1)}] - \Pr[X^{(i,j)}] \right| \\
&\leq \frac{1}{2} \sum_{(i,j) \in [\ell'] \times [\ell']} \left| \Pr[X^{(i,j-1)}] - \Pr[X^{(i,j)}] \right|.
\end{aligned}$$

where the third equality follows from the definition of $X^{(i,j)}$ and the fourth equation follows from our definition $\mathsf{Game}^{(i,0)} = \mathsf{Game}^{(i-1,\ell')}$. Therefore, to prove the theorem, it suffices to show that $|\Pr[X^{(i,j-1)}] - \Pr[X^{(i,j)}]|$ is negligible for all $(i,j) \in [\ell'] \times [\ell']$.

Lemma 21. *For any $i^\star, j^\star \in [\ell']$, there exists PPT adversary $\mathcal{B}$ whose advantage against the adaptive security of the single-bit scheme with redundant key is at least $|\Pr[X^{(i^\star,j^\star-1)}] - \Pr[X^{(i^\star,j^\star)}]|/2$.*

Proof. Suppose an adversary $\mathcal{A}$ that has non-negligible advantage in distinguishing $\mathsf{Game}^{(i^\star,j^\star-1)}$ and $\mathsf{Game}^{(i^\star,j^\star)}$. We use $\mathcal{A}$ to construct an adversary $\mathcal{B}$ against the variant of the single-bit scheme, which proceeds as follows.

Setup. At the beginning of the game, $\mathcal{B}$ is given the master public key $\mathsf{mpk}' = (g, W_0, \{W_{i,j}\}_{[i,j] \in [2] \times [\ell]}, g^\alpha, h^\beta, \{h^{w_{2,i}}\}_{i \in [\ell]}, \{h^{w_{1,i} w_{2,j}}\}_{(i,j) \in [\ell] \times [\ell]}, \mathsf{rand})$ for the single bit scheme. Then, $\mathcal{B}$ picks $\tilde{\alpha}_i \xleftarrow{\$} \mathbb{Z}_p$ for $i \in [\ell'] \backslash \{i^\star\}$ and $\tilde{\beta}_j \xleftarrow{\$} \mathbb{Z}_p$ for $j \in [\ell'] \backslash \{j^\star\}$ and sets

$$g^{\alpha_i} = \begin{cases} g^{\tilde{\alpha_i}} & \text{for } i \in [\ell'] \backslash \{i^\star\} \\ g^\alpha & \text{for } i = i^\star \end{cases}, \qquad h^{\beta_j} = \begin{cases} h^{\tilde{\beta}_j} & \text{for } j \in [\ell'] \backslash \{i^\star\} \\ h^\beta & \text{for } j = j^\star \end{cases}.$$

Note that $\mathcal{B}$ implicitly sets $\alpha_{i^\star} = \alpha$ and $\beta_{j^\star} = \beta$ here. Finally, it gives the master public key of the multi-bit scheme $\mathsf{mpk} = (g, W_0, \{W_{1,i}\}_{i=1}^{\ell}, \{W_{2,i}\}_{i=1}^{\ell}, \{g^{\alpha_i}\}_{i=1}^{\ell'}, \{h^{\beta_i}\}_{i=1}^{\ell'}, \mathsf{rand})$ to $\mathcal{A}$. $\mathcal{B}$ does not give h^{w_0}, $\{h^{w_{2,i}}\}_{i\in[\ell]}$, and $\{h^{w_{1,i}w_{2,j}}\}_{(i,j)\in[\ell]\times[\ell]}$ to $\mathcal{A}$ and keeps them secret.

Phase 1 and Phase 2. When $\mathcal{A}$ makes a key extraction query for ID, $\mathcal{B}$ proceeds as follows. We first observe that $\mathcal{B}$ can compute $h^{\alpha_i\beta_j}$ for all $(i,j) \in ([\ell'] \times [\ell'])\backslash\{(i^\star, j^\star)\}$ as follows:

$$h^{\alpha_i\beta_j} = \begin{cases} h^{\tilde{\alpha}_i\tilde{\beta}_j} & \text{for } i \neq i^\star,\ j \neq j^\star \\ (h^{\alpha})^{\tilde{\beta}_j} & \text{for } i = i^\star,\ j \neq j^\star \\ (h^{\beta})^{\tilde{\alpha}_i} & \text{for } i \neq i^\star,\ j = j^\star \end{cases}. \tag{74}$$

For $(i,j) \in ([\ell']\times[\ell'])\backslash\{(i^\star, j^\star)\}$, $\mathcal{B}$ picks $r^{(i,j)} \xleftarrow{\$} \mathbb{Z}_p$ and computes $\mathsf{sk}^{(i,j)} = (A_1^{(i,j)}, A_2^{(i,j)}, \{B_k^{(i,j)}\}_{k=1}^{\ell})$ as

$$A_1^{(i,j)} = h^{\alpha_i\beta_j} \cdot \left(h^{w_0} \prod_{(i',j')\in S(\mathsf{ID})} h^{w_{1,i'}w_{2,j'}} \right)^{r^{(i,j)}},\ A_2^{(i,j)} = h^{-r^{(i,j)}},\ \left\{ B_k^{(i,j)} = (h^{w_{2,k}})^{r^{(i,j)}} \right\}_{k=1}^{\ell}.$$

These can be computed using h^{w_0}, $h^{w_{2,i'}}$, and $h^{w_{1,i'}w_{2,j'}}$. To generate other parts of the private key (i.e., $\mathsf{sk}_{\mathsf{ID}}^{(i^\star,j^\star)}$), $\mathcal{B}$ resort to its challenger. Namely, $\mathcal{B}$ makes key extraction query for ID and obtains $\mathsf{sk}'_{\mathsf{ID}} = (A_1 = h^{\alpha\beta + r\cdot\mathsf{H}(\mathsf{ID})} = h^{\alpha_{i^\star}\beta_{j^\star} + r\cdot\mathsf{H}(\mathsf{ID})}, A_2 = h^{-r}, \{B_k = h^{rw_{2,k}}\}_{k=1}^{\ell})$. Then, it sets

$$\mathsf{sk}_{\mathsf{ID}}^{(i^\star,j^\star)} = \left(A_1^{(i^\star,j^\star)} = A_1,\quad A_2^{(i^\star,j^\star)} = A_2,\quad \{B_k^{(i^\star,j^\star)} = B_k\}_{k=1}^{\ell} \right).$$

Finally, it returns the secret key $\mathsf{sk}_{\mathsf{ID}} = \{\mathsf{sk}_{\mathsf{ID}}^{(i,j)}\}_{(i,j)\in[\ell']\times[\ell']}$.

Challenge Query. When $\mathcal{A}$ makes the challenge query for the challenge identity $\mathsf{ID}^\star$ and messages $\mathsf{M}_0, \mathsf{M}_1 \in \{0,1\}^{\ell_M}$, $\mathcal{B}$ proceeds as follows. It makes a challenge query for its challenger for the identity $\mathsf{ID}^\star$ and messages $(\mathsf{M}_0^{(i^\star,j^\star)}, \mathsf{M}_1^{(i^\star,j^\star)})$, where $\mathsf{M}_b^{(i^\star,j^\star)}$ is the $(i^\star-1)\ell' + j^\star$th bit of M_b. Then, the challenge ciphertext

$$\left(C'_0 = \mathsf{M}_{\mathsf{coin}}^{(i^\star,j^\star)} \oplus \mathsf{GL}\big(e(g,h)^{s\alpha\beta}, \mathsf{rand}\big),\quad C'_1 = g^s,\quad C'_2,\quad \{D'_j\}_{j=1}^{\ell} \right)$$

is given to $\mathcal{B}$. $\mathcal{B}$ then computes $\mathsf{K}^{(i,j)} = \mathsf{GL}\big(e(C_1, h^{\alpha_i\beta_j}), \mathsf{rand}\big) = \mathsf{GL}\big(e(g,h)^{s\alpha_i\beta_j}, \mathsf{rand}\big)$ for $(i,j) \in ([\ell']\times[\ell'])\backslash\{(i^\star,j^\star)\}$. This is possible because $h^{\alpha_i\beta_j}$ for $(i,j) \neq (i^\star,j^\star)$ can be efficiently computable as we observed in Eq.(74). Finally, $\mathcal{B}$ sets $C_0 \in \{0,1\}^{\ell_M}$ as follows. In the following, $C_0^{(i,j)}$ denotes $(i-1)\ell' + j$th bit of C_0.

$$C_0^{(i,j)} = \begin{cases} \mathsf{K}^{(i,j)} \oplus \mathsf{M}_1^{(i,j)} & \text{for } (i < i^\star) \vee (i = i^\star \wedge j < j^\star) \\ C'_0 & \text{for } i = i^\star,\ j = j^\star \\ \mathsf{K}^{(i,j)} \oplus \mathsf{M}_0^{(i,j)} & \text{for } (i > i^\star) \vee (i = i^\star \wedge j > j^\star) \end{cases}.$$

Finally, $\mathcal{B}$ returns the challenge ciphertext $(C_0, C_1, C_2, \{D_j\}_{j=1}^{\ell})$ to $\mathcal{B}$.

Guess. At last, $\mathcal{A}$ outputs $\widehat{\mathsf{coin}}$. Then, $\mathcal{B}$ outputs $\mathsf{coin}' = \widehat{\mathsf{coin}}$.

Analysis. It can be seen that the view of $\mathcal{A}$ corresponds to that in $\mathsf{Game}^{(i^\star,j^\star-1)}$ if $\mathsf{coin} = 0$ and $\mathsf{Game}^{(i^\star,j^\star)}$ if $\mathsf{coin} = 1$. Therefore, $\mathcal{B}$'s advantage is

$$\left| \Pr[\mathsf{coin}' = \mathsf{coin}] - \frac{1}{2} \right| \quad = \quad \left| \Pr[\widehat{\mathsf{coin}} = \mathsf{coin}] - \frac{1}{2} \right|$$

$$
\begin{aligned}
&= \left|\frac{1}{2}\Pr[\widehat{\mathsf{coin}} = 1|\mathsf{coin} = 1] + \frac{1}{2}\Pr[\widehat{\mathsf{coin}} = 0|\mathsf{coin} = 0] - \frac{1}{2}\right| \\
&= \frac{1}{2}\left|\Pr[\widehat{\mathsf{coin}} = 1|\mathsf{coin} = 1] - \Pr[\widehat{\mathsf{coin}} = 1|\mathsf{coin} = 0]\right| \\
&= \frac{1}{2}\left|\Pr[X^{(i^\star,j^\star-1)}] - \Pr[X^{(i^\star,j^\star)}]\right|
\end{aligned}
$$

as desired. This completes the proof of Lemma 21. □

This completes the proof of Theorem 5. □

E Proof of Theorem 1

Here, we prove Theorem 1. Note that the proof is obtained by the straightforward combination of previous results (in particular, those of [LPR10] and [LS15]). However, to the best of our knowledge, there are no papers explicitly proving the theorem. This section is included for the purpose of completeness.

E.1 Gaussians over Ideal Lattices

We give a brief overview of Gaussians over ideal lattices and introduce the notations we will be using. We refer the general definitions of rings and ideal lattices to the works of [LPR10, LPR13]. In what follows, ζ_m is the primitive mth root of unity for $m > 2$, $\Phi_m(X)$ is the mth cyclotomic polynomial, $K = \mathbb{Q}(\zeta_m)$ is the mth cyclotomic number field of degree $n = \varphi(m)$, $R = \mathbb{Z}[\zeta_m] \cong \mathbb{Z}[X]/(\Phi_m(X))$ is the ring of integers of K[7] , $R^\vee \subseteq K$ is the dual ring and $K_\mathbb{R} = K \otimes_\mathbb{Q} \mathbb{R}$ is the field tensor product. Furthermore, the number field K has exactly n ring embeddings $\sigma_i : K \to \mathbb{C}$ that maps ζ_m to each of the complex roots of the cyclotomic polynomial $\Phi_m(X)$. The canonical embedding $\sigma : K \to \mathbb{C}^n$ is then defined as $\sigma(a) \to (\sigma_i(a))_{i\in\mathbb{Z}_m^*}$.

The Space H. Recall that when working with K (or $K_\mathbb{R}$) under the canonical embedding σ, it is convenient to use the following subspace $H \subseteq \mathbb{C}^n$,

$$H = \{(x_j)_{j\in\mathbb{Z}_m^*} \mid \forall j \in \mathbb{Z}_m^*,\ x_j = \overline{x_{m-j}} \in \mathbb{C}\}.$$

The space H is isomorphic as a real vector space to $K_\mathbb{R}$ via σ. Furthermore, the space H is a $\mathbb{R}$ vector space generated by the columns of the following basis matrix $\mathbf{T}$,

$$\mathbf{T} = \frac{1}{\sqrt{2}}\begin{bmatrix} \mathbf{I}_{n/2} & i\mathbf{J}_{n/2} \\ \mathbf{J}_{n/2} & -i\mathbf{I}_{n/2} \end{bmatrix} \in \mathbb{C}^{n\times n},$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{J}$ is the matrix with ones on the anti-diagonal. Let $\mathbf{h}_j$ denote the jth column of $\mathbf{T}$. Then, for any $a \in K$, there is a unique $\mathbf{v} = (v_1, \ldots, v_n) \in \mathbb{R}^n$ such that $\sigma(a) = \mathbf{T}\mathbf{v}^T = \sum_{j\in[n]} v_j\mathbf{h}_j$, where σ denotes the canonical embedding.

Gaussians over H. For $r > 0$, the Gaussian function $\rho_r : H \to (0, 1]$ over H is defined as, $\rho_r(\mathbf{x}) = \exp(-\pi\|\mathbf{x}\|_2^2/r^2)$ for all $\mathbf{x} \in H$. By appropriately normalizing the Gaussian function ρ_r, we obtain the *continuous spherical Gaussian distribution D_r over H*. We use the basis $\{\mathbf{h}_j\}_{j\in n}$ to define the *continuous elliptical* Gaussian distribution as in [LPR10]. Let $\mathbf{r} = [r_1, \ldots, r_n] \in \mathbb{R}^n_{>0}$ be a vector of positive real numbers such that $r_j = r_{n+1-j}$ for all $j \in [n]$. Then a sample $\mathbf{x}$ from

[7] Note that in our main body, we view R as $\mathbb{Z}[X]/(X^n + 1)$ w.l.o.g .

the elliptical Gaussian distribution $D_{\mathbf{r}}$ over H is given by $\sum_{j\in[n]} v_j \mathbf{h}_j$, where each v_j are chosen independently from the one-dimensional Gaussian distribution D_{r_j} over $\mathbb{R}$. One can check that in case all r_j are the same, this distribution coincides with the above spherical Gaussian distribution, since we have $x_j = \overline{x_{n+1-j}}$ for $\mathbf{x} \in H$. In case we want to explicitly express the domain in which the Gaussian distribution is defined over, we use a superscript to denote it, e.g., $D^H_{\mathbf{r}}$.

The *discrete* (*spherical*) Gaussian is defined similarly to the standard lattices in $\mathbb{R}^n$. Namely, for a lattice in $\Lambda \subset H$, a vector $\mathbf{u} \in H$ and a real $r > 0$, the discrete Gaussian distribution over the coset $\Lambda + \mathbf{u}$ is defined as $D_{\Lambda+\mathbf{u},r}(\mathbf{x}) = \rho_r(\mathbf{x})/\rho_r(\Lambda + \mathbf{u})$ for all $\mathbf{x} \in \Lambda + \mathbf{u}$.

Gaussians over $K_{\mathbb{R}}$. Using the canonical embedding $\sigma : K_{\mathbb{R}} \to H$ (which is an isomorphism), we can consider a *continuous Gaussian distribution* $D^{K_{\mathbb{R}}}_{\mathbf{r}}$ *over* $K_{\mathbb{R}}$ induced by $D^H_{\mathbf{r}}$. Recall we can uniquely express any $a \in K_{\mathbb{R}}$ as a $\mathbb{R}$-linear combination of the power basis $\{\zeta_m^i\}_{i=0}^{n-1}$. Namely, if we denote $\boldsymbol{\zeta}$ as the ordered power basis, then $a = \phi(a)\boldsymbol{\zeta}^T$ for all $a \in K_{\mathbb{R}}$, where ϕ denotes the coefficient embedding. Next, let Δ_m (or CRT_m) denote the matrix corresponding with evaluating a polynomial at all the primitive mth root of unity, i.e., $\sigma(a) = \Delta_m \phi(a)^T \in H$ for all $a \in K_{\mathbb{R}}$. Then, using this expression a sample $a \in K_{\mathbb{R}}$ from $D^{K_{\mathbb{R}}}_{\mathbf{r}}$ is given by $\phi(a)\boldsymbol{\zeta}^T$, where $\mathbf{x} \in H$ is sampled from the continuous Gaussian distribution $D^H_{\mathbf{r}}$ and $\phi(a)$ is set as $\Delta_m^{-1}\mathbf{x}^T$. By definition, we have $D^{K_{\mathbb{R}}}_{\mathbf{r}}(a) = D^H_{\mathbf{r}}(\sigma(a))$. Furthermore, recalling the definition of $D^H_{\mathbf{r}}$, we can also view $D^{K_{\mathbb{R}}}_{\mathbf{r}}$ being induced by $D^{\mathbb{R}^n}_{\mathbf{r}} = D_{r_1} \times \cdots \times D_{r_n}$. Concretely, a sample $a \in K_{\mathbb{R}}$ of $D^{K_{\mathbb{R}}}_{\mathbf{r}}$ can also be obtained by first sampling $\mathbf{v} \in \mathbb{R}^n$ from $D^{\mathbb{R}^n}_{\mathbf{r}}$ and then setting a to satisfy $\phi(a) = \Delta_m^{-1}\mathbf{T}\mathbf{v}^T$.

Gaussians over Fractional Ideals I in K. Recall that a fractional ideal I in K is a set such that $dI \subseteq R$ is an integral ideal for some $d \in R$ and that has a $\mathbb{Z}$-basis $U = \{u_1, \ldots, u_n\} \subseteq K$. Therefore, under the canonical embedding σ, the ideal yields a rank n lattice $\sigma(\mathcal{I})$ in H having basis $\{\sigma(u_1), \ldots, \sigma(u_n)\} \subset H$. We call this lattice $\sigma(\mathcal{I})$ created by the fractional ideal $\mathcal{I}$ as an *ideal lattice*. As in the case of $K_{\mathbb{R}}$, we can consider a *discrete Gaussian distribution over the ideal* $\mathcal{I}$. For a fractional ideal $\mathcal{I} \subset K$, element $t \in K$, and real $r > 0$, the discrete Gaussian distribution over $\mathcal{I} + t$ is defined as $D_{\mathcal{I}+t,r}(a) = D_{\sigma(\mathcal{I})+\sigma(t),r}(\sigma(a))$ for all $a \in \mathcal{I} + t$.

Discretization over Ideal Lattices. Theorem 3.1 of [Pei10] holds for lattices in H. Therefore, we can use it to discretize the contiunous Gaussian distribution $D^{K_{\mathbb{R}}}_r$ to the discrete Gaussian distribution $D_{I+t,r'}$ as follows. Note that $\eta_\epsilon(I)$ denotes the smoothing parameter for the ideal lattice $\sigma(I)$.

Lemma 22. *Let s, s_1, s_2 be positive reals such that $s^2 \geq s_1^2 + s_2^2$. Let I be a fractional ideal in K and t an element in $K_{\mathbb{R}}$. Further assume that $s_1 \geq \eta_\epsilon(I)$ for some positive $\epsilon \leq 1/2$. Then, if we choose a_2 from the continuous Gaussian $D^{K_{\mathbb{R}}}_{s_2}$ over $K_{\mathbb{R}}$ and then choose a_1 from the discrete Gaussian D_{I+t-a_2,s_1}, then $a_1 + a_2$ is within statistical distance 8ϵ of the discrete Gaussian $D_{I+t,s}$.*

Proof. The statement is a direct result of [Pei10], Theorem 3.1 by noticing the following facts: $D^{K_{\mathbb{R}}}_{s_2}(a) = D^H_{s_2}(\sigma(a))$ for all $a \in K_{\mathbb{R}}$, $D_{I+t,s_1}(a) = D_{\sigma(I)+\sigma(t),s_1}(\sigma(a))$ for all $a \in I + t, t \in K_{\mathbb{R}}$, and that $\sigma(I)$ embeds as a lattice in H. □

E.2 Power of 2 Polynomial Rings

Here, we discuss the power of 2 polynomial rings and its properties. For the special case when m is a power of 2, the mth cyclotomic polynomial is given as $\Phi_m(X) = X^n + 1$ where $n = \varphi(m) = m/2$. Therefore, $R \cong \mathbb{Z}[X]/(X^n + 1)$. For this special case, all the columns of Δ_m are orthogonal to each other and we have $\Delta_m^{-1} = \frac{1}{n}\Delta_m^*$, where Δ_m^* is the conjugate transpose. In other words, $\frac{1}{\sqrt{n}}\Delta_m$ is a unitary matrix. Using the properties $\sigma(a) = \Delta_m \phi(a)^T$ and $\phi(\boldsymbol{b}\boldsymbol{R}) = \phi(\boldsymbol{b})\mathrm{rot}(\boldsymbol{R})$ for any element $a \in K_{\mathbb{R}}$, vector $\boldsymbol{b} \in K^s_{\mathbb{R}}$ and matrix $\boldsymbol{R} \in K^{s\times t}_{\mathbb{R}}$, we obtain the following facts:

- $\|\sigma(a)\|_2 = \sqrt{n}\|\phi(a)\|_2$,
- $s_1(\boldsymbol{R}) = \max\limits_{\boldsymbol{x}\in R^t\setminus\{\mathbf{0}\}} \dfrac{\|\sigma(\boldsymbol{xR})\|_2}{\|\sigma(\boldsymbol{x})\|_2} = \max\limits_{\boldsymbol{x}\in R^t\setminus\{\mathbf{0}\}} \dfrac{\|\phi(\boldsymbol{xR})\|_2}{\|\phi(\boldsymbol{x})\|_2} = \max\limits_{\mathbf{z}\in\mathbb{R}^{tn}\setminus\{\mathbf{0}\}} \dfrac{\|\mathbf{z}\cdot \mathrm{rot}(\boldsymbol{R})\|_2}{\|\mathbf{z}\|_2}$.

Recalling the definition of the continuous Gaussian distribution $D_r^{K_\mathbb{R}}$ and the fact that the space H has matrix $\mathbf{T}$ as its basis, $D_r^{K_\mathbb{R}}$ can be described by the procedure of first sampling $\mathbf{v} \xleftarrow{\$} D_r^m$, then outputting $a = \phi(a)\boldsymbol{\zeta}^T$ where $\phi(a)$ is set as $\frac{1}{\sqrt{n}}(\frac{1}{\sqrt{n}}\Delta_m^*)\mathbf{T}\mathbf{v}^T$. Therefore, since $\frac{1}{\sqrt{n}}\Delta_m^*$ and $\mathbf{T}$ are both unitary matrices, a sample from $D_r^{K_\mathbb{R}}$ is simply an element with its coefficients sampled from $D^m_{\sqrt{n}r}$. Finally, for the special power of 2 polynomial ring, we have $R^\vee = \frac{1}{n}R$.

E.3 Ring LWE on Number Fields

We start with recalling the definition of RLWE assumption on number fields (more precisely, on $K_\mathbb{R}$), whose hardness is shown directly in previous works.

Definition 3 (RLWE on $K_\mathbb{R}$)**.** *For integers $n = n(\lambda)$, $k = k(n)$, a prime integer $q = q(n) > 2$, a family of error distribution $\Psi = \Psi(n)$ over $K_\mathbb{R}$, and an PPT algorithm $\mathcal{A}$, an advantage for the RLWE problem $\mathsf{RLWE}^{K_\mathbb{R}}_{n,k,q,\Psi}$ of $\mathcal{A}$ is defined as follows:*

$$\mathsf{Adv}^{\mathsf{RLWE}^{K_\mathbb{R}}_{n,k,q,\Psi}}_{\mathcal{A}} = |\Pr[\mathcal{A}^{\mathcal{O}_{s,\chi}}(1^\lambda, n, k, q) \to 1] - \Pr[\mathcal{A}^{\mathcal{O}_\$}(1^\lambda, n, k, q) \to 1]|$$

where $s \xleftarrow{\$} R_q^\vee, \chi \xleftarrow{\$} \Psi$. The oracles $\mathcal{O}_\$$ and $\mathcal{O}_{s,\chi}$ are specified as follows.

$\mathcal{O}_{s,\chi}$: *When called, it picks $a \xleftarrow{\$} R_q$, $e \xleftarrow{\$} \chi$ and returns $(a, as/q + e)$.*

$\mathcal{O}_\$$: *When called, it returns $(a, v) \xleftarrow{\$} R_q \times K_\mathbb{R}/R^\vee$.*

Both oracles can be called at most k times. If there is no bound on the number of calls, we denote $k = \infty$. In case Ψ consists of a single distribution χ we simply treat the set Ψ as a distribution and write $\mathsf{RLWE}^{K_\mathbb{R}}_{n,k,q,\chi}$, and for this particular case $\mathcal{A}$ further receives as input the distribution χ used by the oracle. We say that $\mathsf{RLWE}^{K_\mathbb{R}}_{n,k,q,\Psi}$ assumption holds if $\mathsf{Adv}^{\mathsf{RLWE}^{K_\mathbb{R}}_{n,k,q,\Psi}}_{\mathcal{A}}$ is negligible for all PPT $\mathcal{A}$.

In [LPR10], it is shown that solving $\mathsf{RLWE}^{K_\mathbb{R}}_{n,\infty,p,\Psi}$ with prime p such that $p \equiv 1 \mod m$ and certain Ψ is as hard as quantumly approximating SIVP (or SVP) on ideal lattices in the worst case. In the subsequent work [LS15], it is shown that the former can be further reduced to $\mathsf{RLWE}^{K_\mathbb{R}}_{n,\infty,q,\Psi'}$ with any q and a certain Ψ'. In what follows, $\Psi_{\le\alpha}$ denotes the family of all elliptical Gaussian distributions $D^{K_\mathbb{R}}_{\mathbf{r}}$ where each parameter $r_i \le \alpha$. Furthermore, Υ_β is a certain family of distribution that is parametrized by $\beta \in \mathbb{R}$. Since the precise definition is not necessary for our purpose, we omit this and refer to [LPR10, LS15]. Then, we have the following results.

Lemma 23 ([LPR10], Theorem 3.6)**.** *Let $\beta > 0$ and let $p \ge 2$, $p \equiv 1 \mod m$ be a polynomially bounded prime such that $\beta p \ge \omega(\sqrt{\log n})$. Then there is a probabilistic polynomial-time quantum reduction from $\tilde{O}(\sqrt{n}/\beta)$-approximate SIVP (or SVP) to $\mathsf{RLWE}^{K_\mathbb{R}}_{n,\infty,p,\Upsilon_\beta}$.*

Lemma 24 ([LS15], From Lemma 4.22, 4.24, and 4.26)**.** *Let $p, q \ge 2$ be polynomially bounded primes and $\alpha, \beta \in (0,1)$ such that $\alpha \ge \beta \cdot \max\{1, p/q\} \cdot n^{3/4}\omega(\log^2 n)$ and $\beta p \ge \omega(\sqrt{\log n/n})$. There exists a polynomial reduction from $\mathsf{RLWE}^{K_\mathbb{R}}_{n,\infty,p,\Upsilon_\beta}$ to $\mathsf{RLWE}^{K_\mathbb{R}}_{n,\infty,q,\Psi_{\le\alpha}}$.*

By combining the above Lemmas, we obtain the hardness of the RLWE with arbitrary modulus q for a skewed Gaussian. In the next step (Lemma 27), we further reduce it to the RLWE with spherical Gaussian. To prepare for the proof, we define Rényi Divergence (of order 2) and review its properties following [LPR10, BLL+15].

Definition 4 (Rényi Divergence)**.** *Let us consider two density functions $P, Q : \mathbb{R}^n \to \mathbb{R}^{\geq 0}$ where $P(\mathbf{x}) = 0$ whenever $Q(\mathbf{x}) = 0$. We define the Rényi divergence $RD(P\|Q)$ as*

$$RD(P\|Q) = \int_{\mathbb{R}^n} \frac{P(\mathbf{x})^2}{Q(\mathbf{x})} \mathrm{d}\mathbf{x}.$$

For Rényi Divergence, the following properties hold. For any distribution P and Q, we have $RD(P\|P) = 1$ and $RD(P\|Q) \geq 1$. Let us assume that P (resp. Q) is a direct product of independent distributions P_1 and P_2 (resp. Q_1 and Q_2). Then, we have $RD(P\|Q) = RD(P_1 \times P_2\|Q_1 \times Q_2) = RD(P_1\|Q_1) \cdot RD(P_2\|Q_2)$.

Lemma 25 ([LPR10], Claim 5.15)**.** *Let $r_1, \ldots r_n \in \mathbb{R}^+$ and $s_1, \ldots, s_n \in \mathbb{R}^+$ be such that for all i, $|s_i/r_i - 1| < \sqrt{\log n/n}$. Then, there exists an polynomial $f_{RD} : \mathbb{N} \to \mathbb{R}$ such that $RD(D_{r_1} \times \cdots \times D_{r_n}\|D_{s_1} \times \cdots \times D_{s_n}) = f_{RD}(n)$.*

Lemma 26 (Implicit in [LPR10])**.** *Let P and Q denote distributions with $\mathrm{Supp}(P) \subseteq \mathrm{Supp}(Q)$. Let $A \subseteq \mathrm{Supp}(Q)$ be any set. Then, we have $Q(A) \geq P(A)^2/RD(P\|Q)$ where $P(A)$ and $Q(A)$ are measure of A under P and Q, respectively.*

Here, we review the proof of [LPR10] that converts the error distribution from the skewed Gaussian to the spherical Gaussian.

Lemma 27 (Adapted from [LPR10], Lemma 5.16)**.** *Let q be a polynomially bounded prime, k a positive integer and $\alpha, \beta \in (0, 1)$. There exists a polynomial time reduction from $\mathsf{RLWE}^{K_\mathbb{R}}_{n,\infty,q,\Psi_{\leq\alpha}}$ to $\mathsf{RLWE}^{K_\mathbb{R}}_{n,k,q,\chi}$ with $\chi = D^{K_\mathbb{R}}_\xi$ where $\xi = \alpha(nk/\log(nk))^{1/4}$.*

Proof. We construct an adversary $\mathcal{B}$ against $\mathsf{RLWE}^{K_\mathbb{R}}_{n,\infty,q,\Psi_{\leq\alpha}}$ from adversary $\mathcal{A}$ that solves $\mathsf{RLWE}^{K_\mathbb{R}}_{n,k,q,\chi}$ with non-negligible advantage $\epsilon(\lambda)$. By assumption, there exists a constant $c \in \mathbb{N}$ such that $\epsilon(\lambda) > 1/\lambda^c$ for infinitely many $\lambda \in \mathbb{N}$.

Reduction. $\mathcal{B}$ is equipped with an oracle $\mathcal{O}$ and its task is to distinguish whether $\mathcal{O} = \mathcal{O}_{s,\chi'}$ or $\mathcal{O} = \mathcal{O}_\$$, where $\chi' = D^{K_\mathbb{R}}_\mathbf{r} \xleftarrow{\$} \Psi_{\leq\alpha}$. $\mathcal{B}$ proceeds as follows. It first obtains estimate $\hat{p}_0$ for the probability

$$p_0 := \Pr[\mathcal{A}(\{(a_i, v_i)\}_{i=1}^k) \to 1 | (a_1, v_1), \ldots, (a_k, v_k) \xleftarrow{\$} R_q \times K_\mathbb{R}/R^\vee]$$

by running $\mathcal{A}$ on $N := 100\lambda^{2c+1}$ fresh inputs. It then repeats the following $M := 4\lambda^{2c+1} f_{RD}(nk)$ times, where f_{RD} is the polynomial specified in Lemma 25.

- It picks random $s' \xleftarrow{\$} R_q^\vee$ and $e'_1, \ldots, e'_k \xleftarrow{\$} D^{K_\mathbb{R}}_\xi$. Then it obtains estimate $\hat{p}_1(s', e'_1, \ldots, e'_k)$ for the probability

  $$p_1(s', e'_1, \ldots, e'_k) := \Pr[\mathcal{A}(\{(a_i, v_i + a_i s'/q + e'_i)\}_{i=1}^k) \to 1 | (a_1, v_1), \ldots, (a_k, v_k) \xleftarrow{\$} \mathcal{O}]$$

  by running $\mathcal{A}$ on N fresh inputs. This can be done by calling the oracle Nk times.

If it happens that $|\hat{p}_1(s', e'_1, \ldots, e'_k) - \hat{p}_0| > 1/4\lambda^c$ at any point during the loop, it outputs 1. Otherwise it outputs 0.

Analysis. It is clear that $\mathcal{B}$ is a (probabilistic) polynomial time algorithm. It suffices to show that $\mathcal{B}$ has overwhelming advantage when $\epsilon > 1/\lambda^c$. We note that by the Hoeffding bound, $|p_0 - \hat{p}_0| < 1/10\lambda^c$ and $|p_1(s', e'_1, \ldots, e'_k) - \hat{p}_1(s', e'_1, \ldots, e'_k)| < 1/10\lambda^c$ for any $(s', e'_1, \ldots, e'_k)$ hold except for probability $e^{-N\cdot(1/10\lambda^c)^2} < 2^{-\lambda}$. In the following, we assume that these always hold.

We first observe that if the oracle $\mathcal{O} = \mathcal{O}_{\$}$, it is clear that both inputs to $\mathcal{A}$ follow the the uniform distribution over $R_q \times K_{\mathbb{R}}/R^\vee$. Therefore, $p_0 = p_1(s', e'_1, \ldots, e'_k)$ holds for any $(s', e'_1, \ldots, e'_k)$. Thus,

$$\begin{aligned} |\hat{p}_0 - \hat{p}_1(s', e'_1, \ldots, e'_k)| &\leq |\hat{p}_0 - p_0| + |p_0 - p_1(s', e'_1, \ldots, e'_k)| + |p_1(s', e'_1, \ldots, e'_k) - \hat{p}_1(s', e'_1, \ldots, e'_k)| \\ &\leq 1/10\lambda^c + 1/10\lambda^c < 1/4\lambda^c. \end{aligned}$$

Hence, $\mathcal{B}$ outputs 0 with all but negligible probability.

Next, let us consider the case where $\mathcal{O} = \mathcal{O}_{s,\chi'}$. In this case, during the loop, an input to $\mathcal{A}$ is of the form $\{(a_i, a_i(s+s')/q + e_i + e'_i)\}_{i=1}^k$ where $e_i \xleftarrow{\$} D_{\mathbf{r}}^{K_{\mathbb{R}}}$ and $e'_i \xleftarrow{\$} D_{\xi}^{K_{\mathbb{R}}}$ for $i \in [k]$. Let us define the vector $\mathbf{r}'$ with coordinates $r'^2_j = \xi^2 - r_j^2$. We claim that the average of $p_1(s', e'_1, \ldots, e'_k)$ over $e'_1, \ldots, e'_k$ chosen independently from $D_{\mathbf{r}'}^{K_{\mathbb{R}}}$ (rather than $D_{\xi}^{K_{\mathbb{R}}}$, which is the actual distribution) is at least $1/\lambda^c$ far from p_0. This can be seen by observing that the error terms $e_i + e'_i$ are distributed as $D_{\mathbf{r}}^{K_{\mathbb{R}}} + D_{\mathbf{r}'}^{K_{\mathbb{R}}} = D_{\xi}^{K_{\mathbb{R}}}$ and by our assumption on $\mathcal{A}$. Let us define S as the set of all tuples $(s', e'_1, \ldots, e'_k)$ such that $|p_1(s', e'_1, \ldots, e'_k) - p_0| > 1/2\lambda^c$. By the averaging argument, we have that the measure of S over $U(R_q) \times (D_{\mathbf{r}'}^{K_{\mathbb{R}}})^k$ is at least $1/2\lambda^c$. Now, let us consider the measure of S over $U(R_q) \times (D_{\xi}^{K_{\mathbb{R}}})^k$, which is the actual distribution. By the definition of $D_{\mathbf{r}}^{K_{\mathbb{R}}}$ and since $1 \leq \xi/\sqrt{\xi^2 - r'^2_i} \leq \xi/\sqrt{\xi^2 - \alpha^2} \leq 1 + \sqrt{\log(nk)/nk}$, we have

$$RD(U(R_q) \times (D_{\mathbf{r}'}^{K_{\mathbb{R}}})^k \| U(R_q) \times (D_{\xi}^{K_{\mathbb{R}}})^k) = RD((D_{\mathbf{r}'}^{K_{\mathbb{R}}})^k \| (D_{\xi}^{K_{\mathbb{R}}})^k) = f_{RD}(nk)$$

by Lemma 25. Hence, by Lemma 26, we have that the measure of S over $U(R_q) \times (D_{\xi}^{K_{\mathbb{R}}})^k$ is at least $1/4\lambda^{2c} f_{RD}(nk)$. Therefore, B picks $(s', e'_1, \ldots, e'_k)$ in S at least once during the loop except for probability $(1 - 1/4\lambda^{2c} f_{RD}(nk))^M < 2^{-\lambda}$. Furthermore, for $(s', e'_1, \ldots, e'_k) \in S$, we have that

$$\begin{aligned} |\hat{p}_1(s', e'_1, \ldots, e'_k) - \hat{p}_0| &\geq |p_1(s', e'_1, \ldots, e'_k) - p_0| - |\hat{p}_1(s', e'_1, \ldots, e'_k) - p_1(s', e'_1, \ldots, e'_k)| - |\hat{p}_0 - p_0| \\ &> 1/2\lambda^c - 1/10\lambda^c - 1/10\lambda^c > 1/4\lambda^c \end{aligned}$$

Therefore, $\mathcal{B}$ outputs 1 with all but negligible probability in this case. □

Finally, we discretize the error distribution and get rid of $R^\vee$ by scaling it appropriately. The following $\mathsf{RLWE}_{n,k,q,\chi}$ is the problem we considered in the main body of our work (cf. Definition 1).

Lemma 28. *Let m be a power of 2, $n = \varphi(m) = m/2$, k be an integer, $q \equiv 3 \mod 8$ be a prime number, and ξ a positive real satisfying $\xi \geq \omega(\sqrt{\log n/n})/q$. There exists a polynomial time reduction from $\mathsf{RLWE}_{n,k,q,\chi}^{K_{\mathbb{R}}}$ with $\chi = D_{\xi}^{K_{\mathbb{R}}}$ to $\mathsf{RLWE}_{n,k,q,\chi}$ with $\chi = D_{\mathbb{Z}^n,\sqrt{2n}q\xi}^{\mathsf{coeff}}$.*

Proof. To show the theorem, it suffices to show an efficient transformation T that takes $\{(a_i, v_i)\}_{i=1}^k \in (R_q \times K_{\mathbb{R}}/R^\vee)^k$ chosen from either $\mathcal{O}_{\$}$ or $\mathcal{O}_s$ as input and has the following properties.

- If $(a_i, v_i) \xleftarrow{\$} \mathcal{O}_{\$}$ for $i \in [k]$, the output of T is uniform over $(R_q \times R_q)^k$.

- If $(a_i, v_i) \xleftarrow{\$} \mathcal{O}_s$ for $i \in [k]$, the output of T is of the form $\{(a_i, a_i s' + e'_i)\}_{i=1}^k$ where $s' \xleftarrow{\$} R_q$ and $e'_1 \ldots, e'_k \xleftarrow{\$} D^{\mathsf{coeff}}_{\mathbb{Z}^n, \sqrt{2n}q\xi}$.

Given $\{(a_i, v_i)\}_{i=1}^k$, T first discretizes $v_i \in K_{\mathbb{R}}/R^\vee$ to $\bar{v}_i \in \frac{1}{q}R^\vee/R^\vee$ while preserving the correct error distribution by adding samples d_i chosen from $D_{\frac{1}{q}R^\vee - v'_i, \xi}$ to each v_i where $v'_i = v_i \mod \frac{1}{q}R^\vee$. We show the validity of this procedure. The case when the input to T is from $\mathcal{O}_\$$ is trivial. Hence, we assume the input was from $\mathcal{O}_s$, i.e., $v_i = a_i s + e_i$ for $e_i \xleftarrow{\$} D^{K_{\mathbb{R}}}_\xi$. For the special case when m is a power of 2, we have $\eta_\epsilon(\frac{1}{q}R^\vee) = \omega(\sqrt{\log n/n})/q$ for some negligible $\epsilon > 0$. Therefore, by the condition on ξ and from Lemma 22, $\bar{e_i} = e_i + d_i$ is distributed negligibly close to the discrete Gaussian distribution $D_{\frac{1}{q}R^\vee, \sqrt{2}\xi}$ when $e_i \xleftarrow{\$} D^{K_{\mathbb{R}}}_\xi$ and $d_i \xleftarrow{\$} D_{\frac{1}{q}R^\vee - e_i, \xi}$. Since $e_i = v'_i \mod \frac{1}{q}R^\vee$, this d_i has the same distribution as the d_i sampled in the above procedure. Therefore, T outputs $\bar{v}_i = a_i s + \bar{e}_i$ where $\bar{e}_i \xleftarrow{\$} D_{\frac{1}{q}R^\vee, \sqrt{2}\xi}$ if the input is from $\mathcal{O}_s$.

Then, T sets $v'_i = qn\bar{v}_i$ in order to move into R. We can see that $\{v'_i\}_{i=1}^k$ are uniformly distributed over R_q when the oracle is $\mathcal{O}_\$$. This is because $R^\vee = \frac{1}{n}R$, which holds whenever m is a power of 2. When $\mathcal{O} = \mathcal{O}_s$, we have $v'_i = a_i ns + qne_i$. We can see that $s' := ns$ is uniformly random over R_q. We can also see that the distribution of qne_i follows $D_{R, \sqrt{2}qn\xi}$. We complete the proof by observing that for m a power of 2, we have $D_{R, \sqrt{2}qn\xi} = D^{\mathsf{coeff}}_{\mathbb{Z}^n, \sqrt{2n}q\xi}$, which follows from the fact that $\phi(R) = \mathbb{Z}^n$ and $\|\sigma(a)\| = \sqrt{n}\|\phi(a)\|$ for any $a \in K_{\mathbb{R}}$. Recall that $D^{\mathsf{coeff}}_{\mathbb{Z}^n, \sqrt{2n}q\xi}$ is the distribution of $a \in R$ where the coefficient vector of a is sampled from $D_{\mathbb{Z}^n, \sqrt{2n}q\xi}$

□

Contents