Extended Nested Dual System Groups, Revisited

Junqing Gong[1], Jie Chen[2(✉)], Xiaolei Dong[3], Zhenfu Cao[3(✉)], and Shaohua Tang[4]

[1] Department of Computer Science and Engineering,
Shanghai Jiao Tong University, Shanghai, China
gongjunqing@126.com
[2] Shanghai Key Laboratory of Multidimensional Information Processing
and Shanghai Key Lab of Trustworthy Computing,
East China Normal University, Shanghai, China
S080001@e.ntu.edu.sg
[3] Shanghai Key Lab for Trustworthy Computing,
East China Normal University, Shanghai, China
{dongxiaolei, zfcao}@sei.ecnu.edu.cn
[4] School of Computer Science & Engineering,
South China University of Technology, Guangzhou, China
shtang@IEEE.org

Abstract. The notion of extended nested dual system groups (ENDSG) was recently proposed by Hofheinz *et al.* [PKC 2015] for constructing almost-tight identity based encryptions (IBE) in the multi-instance, multi-ciphertext (MIMC) setting. However only a composite-order instantiation was proposed and more efficient prime-order instantiations are absent. The paper fills the blank by presenting two constructions.

We revise the definition of ENDSG and realize it using prime-order bilinear groups based on Chen and Wee's prime-order instantiation of nested dual system groups [CRYPTO 2013]. This yields the first almost-tight IBE in the prime-order setting achieving *weak* adaptive security in MIMC scenario under the d-linear (d-Lin) assumption. We further enhanced the revised ENDSG to capture stronger security notions for IBE, including B-*weak* adaptive security and *full* adaptive security. We show that our prime-order instantiation is readily B-weak adaptive secure and full adaptive secure without introducing extra assumption.

We then try to find better solutions by fine-tuning ENDSG again and realizing it using the technique of Chen, Gay, and Wee [EUROCRYPT 2015]. This leads to an almost-tight secure IBE in the same setting with better performance than our first result, but the security relies on a non-standard assumption, d-linear assumption with auxiliary input (d-LinAI) for an even positive integer d. However we note that, the 2-LinAI assumption is implied by the external decisional linear (XDLIN) assumption. This concrete instantiation could also be realized using *symmetric* bilinear groups under standard decisional linear assumption.

Keywords: Identity based encryptions, Dual system groups, Tight security, Security model, Prime-order bilinear groups

1 Introduction

Dual System Encryption. Recently we have witnessed a breakthrough of proof technique in the field of functional encryptions. In 2009, Waters [36] proposed a new proof paradigm for identity based encryptions (IBE), called *dual system technique*, and obtained the first adaptively secure IBE with short public key in the standard model whose security relies on a static assumption and the security loss is $O(q)$ where q is the number of key extraction queries. From a high-level view, the dual system technique works with two copies of some target cryptographic primitive such as IBE. The first copy is put into the so-called *normal space* and acts as the real system, while the second copy is put into the so-called *semi-functional space* and only used in the proof. Furthermore, the independence of the two spaces (say, orthogonality under pairing operations) allows us to make some changes in the semi-functional space for proof but still maintain the correctness in the normal space. It is worth noting that the new technique permits the simulator to reply all queries made by the adversary and avoids the security loss caused by the classical partitioning technique [12, 10, 35].

The revolution was then spreading across the field of functional encryptions. In particular, the dual system technique has been applied for establishing adaptive security of various types of functional encryptions, ranging from simple functionality, such as IBE [9, 16, 22, 15, 25, 14, 32] to expressive and complicated functionality, like ABE and IPE [26, 27, 31, 5, 16, 37, 7, 13]. Some of them applied the dual system technique in a modular and abstract fashion such as Wee's predicate encoding [37] and Attrapadung's pairing encoding [5].

Almost-tight Reduction. The dual system technique also helped us to go further. Chen and Wee [15] combined the dual system technique with the proof idea underlying the Naor-Reingold pseudorandom function [28] and achieved the first almost-tight IBE from a standard assumption in the standard model. The security loss is $O(n)$ where n is the length of identities, and unrelated to the number of key extraction queries anymore. They established the real system in the normal space and a mirror one in the semi-functional space for proof as the original dual system technique [36]. However, instead of dealing with key extraction queries (in the semi-functional space) separately as Waters [36], they handled all (i.e., q) secret keys as a whole in the next step following the proof strategy of Naor and Reingold [28]. In detail, we may imagine the master secret key as a truly random function taking identities as input. Starting from the original master secret key whose domain is just $\{\epsilon\}$, the proof argues that one can double the domain size until it reaches the size of the identity space if identities are encoded in a bit-by-bit fashion [35]. For identity space $\{0,1\}^n$, only n steps are required. Finally, the property of the random function allows us to information-theoretically hide the challenge message.

Recent work by Hofheinz *et al.* [21] extended Chen and Wee's result [15] and achieved almost tightness in the *multi-instance, multi-ciphertext* (MIMC) setting where the adversary simultaneously attacks multiple challenge identities in multiple IBE instances. In Chen and Wee's paradigm [15], the ith step that

increases the domain size from 2^{i-1} to 2^i can only handle the situation where all challenge ciphertexts share the same ith bit, which no longer holds in the MIMC setting. The proposed solution [21] is to further split the semi-functional space into two independent (in some sense) subspaces, labelled by $\wedge$ and $\sim$ respectively. The ith step starts from ciphertexts with $\wedge$-semi-functional component. They then move the semi-functional components in all ciphertexts for identities whose ith bit is 1 to the $\sim$-semi-functional space. At this moment, (1) in the $\wedge$-semi-functional space, all ciphertexts share the same ith bit 0; (2) in the $\sim$-semi-functional space, all ciphertexts share the same ith bit 1, which means that one can now applied Chen and Wee's proof strategy [15] in both subspaces separately.

We emphasize that achieving tight reduction, especially in the MIMC setting, is of practical importance. Consider a scenario involving λ instances and Q ciphertexts per instance. A trivial but generic transformation arises multiplicative $\mathcal{O}(\lambda Q)$ security loss where both λ and Q may be quite huge quantities, say 2^{30}. Therefore a large group should be employed to compensate the loss. This always leads to longer ciphertexts and lower encryption/decryption procedures.

Problem and Goal. Hofheinz *et al.* only provided an instantiation of the above proof strategy using *composite-order* bilinear groups [21]. Our goal is to realize a fully and almost-tightly secure IBE in the MIMC setting using *prime-order* bilinear groups. We emphasize that it is not just a theoretical interest to pursue such a solution. Most schemes (including [21]) using composite-order bilinear groups base their security on the *Subgroup Decision Assumption* [8] which implies the hardness of factoring the group order. This forces us to work with elliptic curve groups with quite large, say 1024 bits, base field when implementing the scheme. In contrast, for constructions in the prime-order setting, we could employ smaller base field, say 160 bits, without sacrificing the security. Although the construction now becomes complex in general, this still brings us a considerable advantage in both computation and space efficiency.

1.1 Motivation and Observation

Hofheinz *et al.*'s work [21] roughly follows the style of [15]. In particular, they first extended the notion of *Nested Dual System Groups* (NDSG) proposed by Chen and Wee [15], then proposed a general IBE construction from the extended NDSG (ENDSG) in the MIMC setting, and finally presented an instantiation of ENDSG using composite-order bilinear groups. Therefore it is sufficient for our purpose to realize ENDSG using prime-order bilinear groups and apply the general transformation in [21]. However we observe that their definition of ENDSG sets too strong requirements on algebraic structure of underlying groups, which makes it hard to be instantiated using existing techniques for prime-order bilinear groups.

An ENDSG describes a set of abstract groups with a bunch of structural and computational requirements supporting Hofheinz *et al.*'s proof strategy. We roughly recall[5] that an ENDSG defined in [21] consists of five algorithms: SampP,

[5] The notation is slightly different from [21].

SampG, SampH, $\widehat{\mathsf{SampG}}$, and $\widetilde{\mathsf{SampG}}$. Informally, the first algorithm generates a set of groups $\mathbb{G}, \mathbb{H}, \mathbb{G}_T$ of order N (as well as other parameters) and the other four algorithms are used to sample random elements from some subgroup of $\mathbb{G}$ or $\mathbb{H}$ (which are associated with ciphertexts and secret keys, respectively, in the context of IBE). We emphasize that they required that

- Groups $\mathbb{G}$ and $\mathbb{H}$ are generated by some $g \in \mathbb{G}$ and $h \in \mathbb{H}$, respectively. (From the specification of group generator G.)
- " The outputs of SampG, $\widehat{\mathsf{SampG}}$, and $\widetilde{\mathsf{SampG}}$ are distributed uniformly over the generators of different nontrivial subgroups of $\mathbb{G}^{n+1}$ of coprime order, respectively. " (From the G-subgroups.)

However, nearly all techniques realizing dual system technique in the prime-order setting employs vector spaces over $\mathbb{F}_p$ (for a prime p) to simulate group $\mathbb{G}$ and $\mathbb{H}$ [25, 27, 31, 15, 16, 13]. Meanwhile subgroups of $\mathbb{G}$ and $\mathbb{H}$ are naturally simulated by its subspaces. Firstly, since a vector space is an additive group but not cyclic in general, neither $\mathbb{G}$ nor $\mathbb{H}$ is cyclic. Secondly, any d-dimensional subspace has p^d vectors, thus the orders of the outputs of SampG, $\widehat{\mathsf{SampG}}$, and $\widetilde{\mathsf{SampG}}$ must share a common factor p. In a word, techniques based on vector spaces by no means meets the requirements shown above.

Fortunately, we observe that both requirements are applied nowhere but to provide random self-reducibility of computational requirements (including LS1, LS2, NH) when they proved "ENDSG implies IBE". For example, the *Left Subgroup Indistinguishability 1* (LS1) said that, for any $(\mathrm{PP}, \mathrm{SP}) \leftarrow \mathsf{SampP}(k, n)$, the following two distributions are computationally indistinguishable.

$$\{\mathbf{g} : \mathbf{g} \leftarrow \mathsf{SampG}(\mathrm{PP})\} \quad \text{and} \quad \left\{\mathbf{g} \cdot \widehat{\mathbf{g}} : \mathbf{g} \leftarrow \mathsf{SampG}(\mathrm{PP}),\ \widehat{\mathbf{g}} \leftarrow \widehat{\mathsf{SampG}}(\mathrm{PP}, \mathrm{SP})\right\}.$$

Given $\mathbf{T}$ which is either $\mathbf{g}$ or $\mathbf{g} \cdot \widehat{\mathbf{g}}$, the simulator (in the proof) can sample $s \leftarrow \mathbb{Z}_N^*$ and generate another independent problem instance $\mathbf{T}^s$ following the two requirements we have reviewed. We note that this property is crucial for achieving almost-tight reduction in the MIMC setting where the adversary is able to enquire more than one challenge ciphertext. This suggests that, if we adapt the ENDSG to support such random self-reducibility *explicitly*, it will still imply an IBE in MIMC setting and the limitations on underlying groups may be removed. As this happens, many existing techniques in the prime-order setting can now be applied to realize ENDSG and finally derive an almost-tight IBE in the MIMC setting using *prime-order* bilinear groups.

1.2 Contributions and Techniques

In this paper, we revise the definition of ENDSG, and show that the revised ENDSG not only almost-tightly implies an IBE in the MIMC setting but also can be *tightly* instantiated using prime-order bilinear groups. Putting them together, we obtain a fully and almost-tightly secure IBE in the same setting from prime-order bilinear groups. In particular, we proposed two instantiations: the first one

is proven secure under the d-linear assumption (d-Lin), while the second one is proven secure under a stronger assumption, d-linear assumption with auxiliary input, d-LinAI for short, but achieves shorter keys and ciphertexts.

Revisiting Extended Nested Dual System Groups. Our ENDSG is defined mainly in the spirit of [21] but with the difference that we provide (in requirements like LS1) enough independently-sampled subgroup elements directly instead of assuming some special algebraic structure. As an example, we define LS1 as: for any $(\mathrm{PP}, \mathrm{SP}) \leftarrow \mathsf{SampP}(k, n)$, the following two distributions are computationally indistinguishable.

$$\left\{ \{\mathbf{g}_j\}_{j\in[q]} : \mathbf{g}_j \leftarrow \mathsf{SampG}(\mathrm{PP}) \right\} \quad \text{and}$$
$$\left\{ \{\mathbf{g}_j \cdot \widehat{\mathbf{g}}_j\}_{j\in[q]} : \mathbf{g}_j \leftarrow \mathsf{SampG}(\mathrm{PP}),\ \widehat{\mathbf{g}}_j \leftarrow \widehat{\mathsf{SampG}}(\mathrm{PP}, \mathrm{SP}) \right\}.$$

Here the parameter q depends on the number of challenge ciphertexts. This makes the definition more general and allows us to realize the notion using diverse algebra frameworks, especially prime-order bilinear groups. On the other hand, it still almost-tightly implies a fully secure IBE in the MIMC setting. The construction and the proof are nearly the same as [21].

To be fair, Hofheinz *et al.*'s definition is more convenient in the sense that any instantiation of ENDSG immediately results in an *almost-tight* IBE in the MIMC setting. In contrast, an instantiation of our definition with *loose* security reduction (say, with security loss $\mathcal{O}(q)$) clearly can not lead to *tightly secure* IBE. Hence, when working with our definition, we should not jump to the conclusion before checking the tightness. We also remark that we *do not* negate prime-order instantiations of Hofheinz *et al.*'s ENDSG.

Instantiation from d-Linear Assumption. We realize our revised ENDSG by extending the prime-order instantiation of NDSG by Chen and Wee [15]. The security only relies on the d-Lin assumption and the security loss is $\mathcal{O}(d)$ and independent of the number of samples, say q in the LS1 example, given to the adversary. By the generic construction [21], we obtain the first almost-tight IBE in the MIMC setting in the prime-order setting and fill the blank left in [21].

Technically, we extend the basis from $2d \times 2d$ matrix used in [15] to $3d \times 3d$ matrix in order to accommodate the additional semi-functional space. In detail, the first d-dimension subspace is the normal space, the next d-dimension subspace is the $\wedge$-semi-functional space, and the last d-dimension subspace is the $\sim$-semi-functional space.

The main challenge is to realize the *Left Subgroup Indistinguishability 2 (LS2)* property (c.f. Section 3). Roughly, we must prove that $\mathbf{g} \cdot \widehat{\mathbf{g}}$ (sampled from the normal space and $\wedge$-semi-functional space of $\mathbb{G}$) and $\mathbf{g} \cdot \widetilde{\mathbf{g}}$ (sampled from the normal space and $\sim$-semi-functional space of $\mathbb{G}$) are computationally indistinguishable even when the adversary can access to $\widehat{h}^* \cdot \widetilde{h}^* \in \mathbb{H}$ where $\widehat{h}^* \in \mathbb{H}$ is orthogonal to the normal and $\sim$-semi-functional space of $\mathbb{G}$ and $\widetilde{h}^* \in \mathbb{H}$ to the normal and $\wedge$-semi-functional space of $\mathbb{G}$. To simulate $\widehat{h}^* \cdot \widetilde{h}^*$, we further extend

the subspace of $\widehat{h}^*$ and $\widetilde{h}^*$ from 1-dimension in [15] to d-dimension which allows us to utilize the technique for proving *right subgroup indistinguishability* of Chen-Wee's prime-order instantiation of dual system groups [16]. So as to support this technical extension and conform to our revision, we model the process of sampling $\widehat{h}^*$ and $\widetilde{h}^*$ as two algorithms $\widehat{\mathsf{SampH}}^*$ and $\widetilde{\mathsf{SampH}}^*$ respectively, and give adversary adequate samples in related computational requirements. With such high-dimension $\widehat{h}^*$ and $\widetilde{h}^*$, the proof of *Nested-hiding Indistinguishability (NH)* (c.f. Section 3) will also be extended accordingly.

Achieving Stronger Security Guarantee. Hofheinz *et al.* [21] achieved weak security from their ENDSG where the adversary is allowed to make single challenge query for each identity in each instance. They introduced a variant of the BDDH assumption (s-BDDH) and proved the full security of their original construction where the above restriction on the adversary is removed. This additional computational requirement is realized under the *dual system bilinear DDH assumption* (DS-BDDH).

The revisions we have made do not involve the s-BDDH assumption, and the resulting ENDSG only leads to weak security. Motivated by and based on our prime-order instantiation, we investigate two flavors of stronger security: B*-weak* and *full adaptive security*. The former model allows adversary to make at most B challenge queries for each identity in each instance where B is a prior bound, while the latter one sets no limitation on the number of challenge queries on a single identity, i.e., polynomially many queries are allowed.

For each of them, we follow Hofheinz *et al.*'s workflow. Concretely, to achieve stronger security, we enhance the *non-degeneracy* property in our revised ENDSG and update the last step of Hofheinz *et al.*'s proof (decoupling challenge messages and ciphertexts) to make it sound in stronger models, where the non-degeneracy property is applied. We then prove that our instantiation of ENDSG under the d-Lin assumption (see Section 4) indeed satisfies the enhanced non-degeneracy property. The two results together imply an IBE with stronger security guarantee and almost-tight reduction in the MIMC setting. In particular,

1. We enhance the non-degenerate property to B*-bounded* version which states that the non-degeneracy property holds even when a single $\widehat{h}^*$ works with B $\widehat{g}_0$'s where B is a prior bound. It is easy to show that our instantiation under the d-Lin assumption is d-bounded non-degenerated unconditionally.
2. We enhance the non-degeneracy property to *computational* version which is essentially similar to the s-BDDH assumption [21] and states that the non-degeneracy property holds even when a single $\widehat{h}^*$ works with *polynomially many* $\widehat{g}_0$'s. Luckily, we can prove that our instantiation is computationally non-degenerated under the d-Lin assumption, and no additional assumption is required.

Towards More Efficient Instantiation. Having obtained the first construction, we continue to purse more efficient solutions. The main idea is to reduce the dimensions of two semi-functional spaces. However this forces us to base the

security on a non-standard assumption, d-LinAI assumption (c.f. Section 7) for *an even positive integer* d. We argue that the concrete assumption with $d = 2$ is implied by the classical external decision linear assumption (XDLIN) [1]. We give an overview of our method and the resulting IBE scheme in Section 7. All details are given in the full version of the paper.

1.3 Comparison and Discussion

We make a comparison among existing almost-tightly secure IBE schemes in the MIMC setting in terms of time and space efficiency. The details are shown in Table 1. Our comparison involves the composite-order construction by Hofheinz *et al.* [21], the prime-order construction in Section 5 based on the decisional linear (DLIN, 2-Lin) and symmetric external Diffie-Hellman (SXDH, 1-Lin) assumption, and the prime-order construction from Section 7 based on the XDLIN (2-LinAI) assumption. As a base line, we also consider the efficiency of prime-order construction by Chen and Wee [15] and Blazy *et al.* [9], which is *not* built for the MIMC setting.

Scheme	$\lvert G\rvert$	Assum.	$\lvert\mathrm{MPK}\rvert$		$\lvert\mathrm{SK}\rvert$	$\lvert\mathrm{CT}\rvert$		T_{Enc}		T_{Dec}	MIMC
			G_1/G	G_T	G_2/G	G_1/G	G_T	E_1/E	E_T	P	
[15]	P	d-Lin	$2d^2(2n+1)$	d	$4d$	$4d$	1	$4d^2$	d	$4d$	✗
		DLIN	$16n+8$	2	8	8	1	16	2	8	
		SXDH	$4n+2$	1	4	4	1	4	1	4	
[9]	P	d-Lin	$(2n+1)d^2+d$	d	$2d+1$	$2d+1$	1	$2d^2+1$	d	$2d+1$	✗
		DLIN	$8n+6$	2	5	5	1	9	2	5	
		SXDH	$2n+2$	1	3	3	1	3	1	3	
[21]	C	Static	$2n+1$	1	2	2	1	2	1	2	✔
Sec. 5	P	d-Lin	$3d^2(2n+1)$	d	$6d$	$6d$	1	$6d^2$	d	$6d$	✔
		DLIN	$24n+12$	2	12	12	1	24	2	12	
		SXDH	$6n+3$	1	6	6	1	6	1	6	
Sec. 7	P	d-LinAI	$2d^2(2n+1)$	d	$4d$	$4d$	1	$4d^2$	d	$4d$	✔
		XDLIN	$16n+8$	2	8	8	1	16	2	8	

Table 1. Comparing Efficiency among existing and proposed almost-tight IBE schemes. n is the length of identities. Column $\lvert\mathrm{MPK}\rvert$, $\lvert\mathrm{SK}\rvert$, and $\lvert\mathrm{CT}\rvert$ show the size of master public keys, user's secret keys and ciphertexts, respectively. Each sub-column contains the number of elements in G, G_1, G_2, and G_T. Column T_{Enc} and T_{Dec} show encryption and decryption cost, respectively. Each sub-column E, E_1, and E_T shows the number of exponentiations on group G, G_1, and G_T, respectively, and sub-column P shows the number of pairings. Column "Assum." shows the underlying assumption. "Static" means static assumptions in the composite-order bilinear group. Column "$\lvert G\rvert$" indicates the group order, "P" for prime and "C" for composite order, respectively.

Hofheinz *et al.*'s construction (see the third row) works with a symmetric bilinear group whose order is the product of four distinct primes, the sizes of group elements are much larger, and exponentiation and pairing operations are

much more expensive. Therefore the overall efficiency is not acceptable even though the numbers of group elements in MSK, SK and CT are smaller and Enc and Dec involve less exponentiation and pairing operations.

When instantiating our first proposal (see the fourth row) under the DLIN assumption, each group element in $\mathbb{G}$ and $\mathbb{H}$ is a 6-dimension vector over G_1 and G_2, respectively, where G_1 and G_2 are source groups of a prime-order bilinear group. When instantiating under the SXDH assumption, each group element in $\mathbb{G}$ and $\mathbb{H}$ is a 3-dimension vector over G_1 and G_2, respectively. Compared with Blazy *et al.*'s construction [9], both size of MPK, SK and CT and cost of Enc and Dec are (at least) doubled in our construction. On the other hand, in our second instantiation based on the XDLIN assumption (see the last row), each group element in $\mathbb{G}$ and $\mathbb{H}$ is a vector of 4-dimension over G. Although the resulting IBE is still less efficient than Blazy *et al.*'s construction [9] under the DLIN assumption, the stronger computational assumption (i.e., XDLIN) helps us to narrow the gap. We may view this as a tradeoff between strength of security and efficiency without changing the security model. We leave it as an open problem to find more efficient fully secure IBE with tight reduction in the MIMC setting, especially from standard d-Lin assumption.

1.4 Related Work

Dual System Groups and Its Variants. Chen and Wee proposed the notion of dual system groups [16], which captures key algebraic structure supporting the dual system technique. They used this abstract primitive to obtain an HIBE scheme with constant-size ciphertexts using prime-order bilinear groups. The nested dual system group, an variant of dual system groups, was proposed by Chen and Wee [15] to reach almost-tight adaptively secure IBE in the standard model. Recently, the dual system group had been combined with the predicate/pairing encoding [13, 2] and led to a lot of functional encryptions in the prime-order setting. Very recent work by Gong *et al.* [20] extended the concept of dual system groups to build an unbounded HIBE [24, 25] with shorter ciphertexts in the prime-order setting.

Identity Based Encryption. The notion of identity based encryptions was introduced by Shamir [33] in 1984. The first practical realization was proposed by Boneh and Franklin [12] using bilinear groups and Cocks [17] using quadratic residue. Both of them rely on the heuristic random oracle model. Before Waters proposed his seminal work, there were several classical and practical solutions in the standard model, including Boneh-Boyen's IBE [11, 10], Waters' IBE [35], and Gentry's IBE [18]. IBE can also be realized using algebra frameworks other than bilinear groups, such as lattices [19, 3, 4].

1.5 Independent Work

The independent work by Attrapadung, Hanaoka, and Yamada [6] also involves several constructions of almost-tight IBE in the MIMC setting. They

developed an elegant framework for building almost-tight IBE in the MIMC setting from the so-called *broadcast encoding*, which is a special form of Attrapadung's pairing encoding [5], and obtained a series of concrete schemes with various properties (including sub-linear size master public key and anonymous version) using both composite-order and prime-order bilinear groups. Their results and ours partially overlap. Their scheme with constant-size ciphertext in prime-order group (i.e., $\Phi_{\mathsf{cc}}^{\mathsf{prime}}$) is similar to our second construction based on the XDLIN assumption shown in Section 7. In fact, they share the same performance in terms of the size of ciphertexts and secret keys and running time of Enc and Dec. However we note that we also provide an generalization of this construction but proven secure under the non-standard d-LinAI assumption. Furthermore, our first construction in Section 5 is full-adaptively secure under the *standard* d-Lin assumption, and derives a SXDH-based concrete scheme, which has the best (space and time) performance among all proposed solutions so far.

Outline. Section 2 presents necessary background. Section 3 gives our revised definition of ENDSG. We realize our revised ENDSG in the prime-order setting in Section 4 and investigate how to update our ENDSG and its prime-order instantiation to achieve higher security level in Section 6. At last, Section 7 is an overview of obtaining a more efficient solution.

2 Preliminaries

2.1 Notations

For a finite set S, we use $s \leftarrow S$ to denote the process of picking s from S at random. For any $n \in \mathbb{Z}^+$, we take $[n]$ as the brief representation of set $\{1, \ldots, n\}$. For a probabilistic algorithm Alg and an fixed input x, we use $[\mathsf{Alg}(x)]$ to indicate the set of all possible outputs of algorithm Alg on input x. "p.p.t." stands for "probabilistic polynomial time". We let $\mathbf{e}_i$ denote the vector with 1 on the ith position and 0 elsewhere. For a group G and $g \in G$, let $h^{\mathbf{e}_i}$ be a vector over G with h on the ith position and 1 elsewhere. For two vectors $\mathbf{g} := (g_1, \ldots, g_n) \in G^n$ and $\mathbf{g}' := (g'_1, \ldots, g'_n) \in G^n$, we define $\mathbf{g} \cdot \mathbf{g}' = (g_1 \cdot g'_1, \ldots, g_n \cdot g'_n) \in G^n$ where "$\cdot$" on the right-hand side is the group operation of G. For any vector $\mathbf{x} = (x_1, \ldots, x_n)$ and $i \in [n]$, we define $\mathbf{x}_{-i}$ as a vector $(x_1, \ldots, x_{i-1}, \perp, x_{i+1}, \ldots, x_n)$ whose ith position is unknown (we take $\perp$ as a placeholder).

2.2 Identity Based Encryptions

Algorithms. An IBE scheme in the multi-instance setting consists of five p.p.t. algorithms defined as follows[6]. (1) The *parameter generation algorithm* $\mathsf{Param}(1^k, \mathrm{SYS})$ takes as input a security parameter $k \in \mathbb{Z}^+$ in its unary form and a system-level parameter SYS, and outputs a global parameter GP. (2) The

[6] The definition shown here is slightly different from that in [21]. The adaptation is purely conceptual and made for clarity. The security model is tuned accordingly.

setup algorithm $\mathsf{Setup}(\text{GP})$ takes as input a global parameter GP, and outputs a master public/secret key pair (MPK, MSK). (3) The *key generation algorithm* $\mathsf{KeyGen}(\text{MPK}, \text{MSK}, \mathbf{y})$ takes as input a master public key MPK, a master secret key MSK and an identity $\mathbf{y}$, and outputs a secret key $\text{SK}_{\mathbf{y}}$ for the identity. (4) The *encryption algorithm* $\mathsf{Enc}(\text{MPK}, \mathbf{x}, \text{M})$ takes as input a master public key MPK, an identity $\mathbf{x}$ and a message M, outputs a ciphertext $\text{CT}_{\mathbf{x}}$ for the message under the identity. (5) The *decryption algorithm* $\mathsf{Dec}(\text{MPK}, \text{SK}, \text{CT})$ takes as input a master public key MPK, a secret key SK and a ciphertext CT, outputs a message M or a failure symbol $\perp$.

The so-called "multi-instance setting" indicates that we are considering a collection of IBE instances established under the same global parameter GP. We leave the system-level parameter SYS undefined for generality. It may depend on concrete constructions or application scenarios.

Correctness. For any parameter $k \in \mathbb{Z}^+$, any SYS, any $\text{GP} \in [\mathsf{Param}(1^k, \text{SYS})]$, any $(\text{MPK}, \text{MSK}) \in [\mathsf{Setup}(\text{GP})]$, any identity $\mathbf{x}$, and any message M, it holds that

$$\Pr\left[\mathsf{Dec}(\text{MPK}, \mathsf{KeyGen}(\text{MPK}, \text{MSK}, \mathbf{x}), \mathsf{Enc}(\text{MPK}, \mathbf{x}, \text{M})) = \text{M}\right] \geqslant 1 - 2^{-\Omega(k)}.$$

The probability space is defined by the random coins consumed by algorithm KeyGen and Enc.

Adaptive Security in the Multi-instance, Multi-ciphertext Setting. Roughly, the adaptive security in the multi-instance, multi-ciphertext setting extends the traditional adaptive security model for IBE [12] in the sense that the adversary can access to multiple IBE instances (obtaining master public key and users' keys) and attack multiple ciphertexts (i.e., challenge ciphertexts), which is formalized by Hofheinz *et al.* [21]. Ideally, the adversary is free to choose the challenge instance, the challenge identity and the challenge message pair. Hofheinz *et al.* [21] also identified a weaker variant in which only one challenge ciphertext is allowed for each challenge identity in each challenge instance, and called the ideal one *full security*.

We review the experiment $\mathbf{Exp}^{\text{IBE}}_{\mathcal{A}}(k, \lambda, q_K, q_C, q_R)$ between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$ [21], which captures both the weaker and full security notion.

Setup. $\mathcal{C}$ gets $\text{GP} \leftarrow \mathsf{Param}(1^k, \text{SYS})$ and creates $(\text{MPK}_\iota, \text{MSK}_\iota) \leftarrow \mathsf{Setup}(\text{GP})$ for $\iota \in [\lambda]$. All master public keys $\{\text{MPK}_\iota\}_{\iota \in [\lambda]}$ are sent to $\mathcal{A}$. $\mathcal{C}$ also chooses a secret random bit $\beta \in \{0, 1\}$ and initializes Q_K and Q_C as empty sets.

Query. $\mathcal{A}$ is allowed to make two types of queries: key extraction queries and challenge queries. $\mathcal{C}$ answers every queries as follows: (1) For each key extraction query $(\iota, \mathbf{y})$, $\mathcal{C}$ returns $\text{SK} \leftarrow \mathsf{KeyGen}(\text{MPK}_\iota, \text{MSK}_\iota, \mathbf{y})$ and updates $Q_K := Q_K \cup \{(\iota, \mathbf{y})\}$. (2) For each challenge query $(\iota^*, \mathbf{x}^*, \text{M}_0^*, \text{M}_1^*)$, $\mathcal{C}$ returns $\text{CT}^* \leftarrow \mathsf{Enc}(\text{MPK}_{\iota^*}, \mathbf{x}^*, \text{M}_\beta^*)$ and updates $Q_C := Q_C \cup \{(\iota^*, \mathbf{x}^*)\}$.

Guess. $\mathcal{A}$ outputs its guess $\beta' \in \{0, 1\}$.

We say an adversary $\mathcal{A}$ wins experiment $\mathbf{Exp}^{\text{IBE}}_{\mathcal{A}}(k, \lambda, q_K, q_C, q_R)$, denoted by $\mathbf{Exp}^{\text{IBE}}_{\mathcal{A}}(k, \lambda, q_K, q_C, q_R) = 1$, if and only if (1) $\beta = \beta'$, (2) $Q_K \cap Q_C = \emptyset$, (3)

$\mathcal{A}$ made at most q_K key extraction queries, (4) there are at most q_C challenge identities, and (5) for each of them, there exist at most q_R challenge ciphertexts. We define the advantage of $\mathcal{A}$ as

$$\mathsf{Adv}^{\mathrm{IBE}}_{\mathcal{A}}(k,\lambda,q_K,q_C,q_R) = \left|\Pr[\mathbf{Exp}^{\mathrm{IBE}}_{\mathcal{A}}(k,\lambda,q_K,q_C,q_R)=1] - 1/2\right|.$$

The probability space is defined by random coins consumed by both $\mathcal{C}$ and $\mathcal{A}$. An IBE is (λ, q_K, q_C, q_R)-*adaptively-secure* if, for any p.p.t. adversary $\mathcal{A}$ the advantage $\mathsf{Adv}^{\mathrm{IBE}}_{\mathcal{A}}(k,\lambda,q_K,q_C,q_K)$ is bounded by $2^{-\Omega(k)}$. Clearly, the (λ, q_k, q_C, q_R)-adaptive security with unbounded q_R is consistent with the full security, while the $(\lambda, q_k, q_C, 1)$-adaptive security is exactly the weak security. Furthermore, we define B-*weak adaptive security*, an intermediate security notion between them, as (λ, q_K, q_C, B)-adaptive security for a priori bound $B \geqslant 1$.

3 Revisiting Extended Nested Dual System Groups

This section revises the ENDSG proposed by Hofheinz *et al.* [21]. Following the intuitive discussion in Section 1, the key points are: we (1) remove special group requirements, (2) explicitly provide samples in each computational assumption, (3) generalize subgroup of $\widehat{h}^*$ and $\widetilde{h}^*$. We show our definition followed by a series of remarks clarifying motivations behind several technical decisions.

Syntax. Our revised ENDSG consists of eight p.p.t. algorithms as follows:

- $\mathsf{SampP}(1^k, n)$: Output: (1) PP containing (a) group description $(\mathbb{G}, \mathbb{H}, \mathbb{G}_T)$ and an admissible bilinear map $e : \mathbb{G} \times \mathbb{H} \to \mathbb{G}_T$; (b) an efficient linear map μ defined on $\mathbb{H}$; (c) an efficient sampler for $\mathbb{H}$ and $\mathbb{Z}_{\mathrm{ord}(\mathbb{H})}$, respectively; (d) public parameters for SampG and SampH. (2) SP containing secret parameters for $\widehat{\mathsf{SampG}}$, $\widetilde{\mathsf{SampG}}$, $\widehat{\mathsf{SampH}}^*$ and $\widetilde{\mathsf{SampH}}^*$.
- SampGT: $\mathrm{Im}(\mu) \to \mathbb{G}_T$.
- $\mathsf{SampG}(\mathrm{PP})$: Output $\mathbf{g} = (g_0, g_1, \ldots, g_n) \in \mathbb{G}^{n+1}$.
- $\mathsf{SampH}(\mathrm{PP})$: Output $\mathbf{h} = (h_0, h_1, \ldots, h_n) \in \mathbb{H}^{n+1}$.
- $\widehat{\mathsf{SampG}}(\mathrm{PP}, \mathrm{SP})$: Output $\widehat{\mathbf{g}} = (\widehat{g}_0, \widehat{g}_1, \ldots, \widehat{g}_n) \in \mathbb{G}^{n+1}$.
- $\widetilde{\mathsf{SampG}}(\mathrm{PP}, \mathrm{SP})$: Output $\widetilde{\mathbf{g}} = (\widetilde{g}_0, \widetilde{g}_1, \ldots, \widetilde{g}_n) \in \mathbb{G}^{n+1}$.
- $\widehat{\mathsf{SampH}}^*(\mathrm{PP}, \mathrm{SP})$: Output $\widehat{h}^* \in \mathbb{H}$.
- $\widetilde{\mathsf{SampH}}^*(\mathrm{PP}, \mathrm{SP})$: Output $\widetilde{h}^* \in \mathbb{H}$.

The first four algorithms are used in the real system, while the remaining ones are defined for the proof. We let SampG_0 refer to the first element in the output of SampG, i.e., g_0. The notation also applies to SampH, $\widehat{\mathsf{SampG}}$, and $\widetilde{\mathsf{SampG}}$.

Correctness. For all $k, n \in \mathbb{Z}^+$ and all $(\mathrm{PP}, \mathrm{SP}) \in [\mathsf{SampP}(1^k, n)]$, we require

(Projective.) For all $h \in \mathbb{H}$ and all possible random coins s, $\mathsf{SampGT}(\mu(h); s) = e(\mathsf{SampG}_0(\mathrm{PP}; s), h)$.

(Associative.) For all $(g_0, g_1, \ldots, g_n) \in [\mathsf{SampG}(\mathrm{PP})]$ and all $(h_0, h_1, \ldots, h_n) \in [\mathsf{SampH}(\mathrm{PP})]$, $e(g_0, h_i) = e(g_i, h_0)$ for $i \in [n]$.

Security. For all $k, n \in \mathbb{Z}^+$ and all $(\mathrm{PP}, \mathrm{SP}) \in [\mathsf{SampP}(1^k, n)]$, we require

(Orthogonality.) For all $\widehat{h}^* \in [\widehat{\mathsf{SampH}}^*(\mathrm{PP}, \mathrm{SP})]$ and all $\widetilde{h}^* \in [\widetilde{\mathsf{SampH}}^*(\mathrm{PP}, \mathrm{SP})]$,
1. $\mu(\widehat{h}^*) = \mu(\widetilde{h}^*) = 1$;
2. $e(\widehat{g}_0, \widetilde{h}^*) = 1$ for all $\widehat{g}_0 \in [\widehat{\mathsf{SampG}}_0(\mathrm{PP}, \mathrm{SP})]$;
3. $e(\widetilde{g}_0, \widehat{h}^*) = 1$ for all $\widetilde{g}_0 \in [\widetilde{\mathsf{SampG}}_0(\mathrm{PP}, \mathrm{SP})]$;

The first requirement implies that $e(g_0, \widetilde{h}^*) = e(g_0, \widehat{h}^*) = 1$ for all $g_0 \in [\mathsf{SampG}_0(\mathrm{PP})]$ by the *projective* property (c.f. Section 3.2 in [15]).

(Non-degeneracy.) Over the probability space defined by $\widehat{g}_0 \leftarrow \widehat{\mathsf{SampG}}_0(\mathrm{PP}, \mathrm{SP})$, with overwhelming probability $1 - 2^{-\Omega(k)}$, $e(\widehat{g}_0, \widehat{h}^*)$ is distributed uniformly over $\mathbb{G}_T$ when sampling $\widehat{h}^* \leftarrow \widehat{\mathsf{SampH}}^*(\mathrm{PP}, \mathrm{SP})$.

($\mathbb{H}$-subgroup.) The output of $\mathsf{SampH}(\mathrm{PP})$ is distributed uniformly over some subgroup of $\mathbb{H}^{n+1}$, while those of $\widehat{\mathsf{SampH}}^*(\mathrm{PP}, \mathrm{SP})$ and $\widetilde{\mathsf{SampH}}^*(\mathrm{PP}, \mathrm{SP})$ are distributed uniformly over some subgroup of $\mathbb{H}$, respectively.

(Left subgroup indistinguishability 1 (LS1).) For any p.p.t. adversary $\mathcal{A}$, the following advantage function is negligible in k,

$$\mathsf{Adv}^{\mathrm{LS1}}_{\mathcal{A}}(k, q) := \left|\Pr[\mathcal{A}(D, T_0) = 1] - \Pr[\mathcal{A}(D, T_1) = 1]\right|,$$

where

$$D := (\mathrm{PP}),\ T_0 := \{\mathbf{g}_j\}_{j \in [q]},\ T_1 := \left\{\mathbf{g}_j \cdot \boxed{\widehat{\mathbf{g}}_j}\right\}_{j \in [q]}$$

and $\mathbf{g}_j \leftarrow \mathsf{SampG}(\mathrm{PP})$ and $\widehat{\mathbf{g}}_j \leftarrow \widehat{\mathsf{SampG}}(\mathrm{PP}, \mathrm{SP})$.

(Left subgroup indistinguishability 2 (LS2).) For any p.p.t. adversary $\mathcal{A}$, the following advantage function is negligible in k,

$$\mathsf{Adv}^{\mathrm{LS2}}_{\mathcal{A}}(k, q, q') := \left|\Pr[\mathcal{A}(D, T_0) = 1] - \Pr[\mathcal{A}(D, T_1) = 1]\right|,$$

where

$$D := \left(\mathrm{PP}, \left\{\widehat{h}^*_j \cdot \widetilde{h}^*_j\right\}_{j \in [q+q']}, \left\{\mathbf{g}'_j \cdot \widehat{\mathbf{g}}'_j\right\}_{j \in [q]}\right),$$

$$T_0 := \{\mathbf{g}_j \cdot \widehat{\mathbf{g}}_j\}_{j \in [q]},\ T_1 := \left\{\mathbf{g}_j \cdot \boxed{\widetilde{\mathbf{g}}_j}\right\}_{j \in [q]}$$

and $\widehat{h}^*_j \leftarrow \widehat{\mathsf{SampH}}^*(\mathrm{PP}, \mathrm{SP})$, $\widetilde{h}^*_j \leftarrow \widetilde{\mathsf{SampH}}^*(\mathrm{PP}, \mathrm{SP})$, $\mathbf{g}'_j \leftarrow \mathsf{SampG}(\mathrm{PP})$, $\widehat{\mathbf{g}}'_j \leftarrow \widehat{\mathsf{SampG}}(\mathrm{PP}, \mathrm{SP})$, $\mathbf{g}_j \leftarrow \mathsf{SampG}(\mathrm{PP})$, $\widehat{\mathbf{g}}_j \leftarrow \widehat{\mathsf{SampG}}(\mathrm{PP}, \mathrm{SP})$, $\widetilde{\mathbf{g}}_j \leftarrow \widetilde{\mathsf{SampG}}(\mathrm{PP}, \mathrm{SP})$.

(Nested-hiding indistinguishability (NH).) For any $\eta \in [\lfloor n/2 \rfloor]$ and any p.p.t. adversary $\mathcal{A}$, the following advantage function is negligible in k,

$$\mathsf{Adv}^{\mathrm{NH}(\eta)}_{\mathcal{A}}(k, q, q') := \left|\Pr[\mathcal{A}(D, T_0) = 1] - \Pr[\mathcal{A}(D, T_1) = 1]\right|,$$

where

$$D := \left(\mathrm{PP}, \left\{\widehat{h}^*_j\right\}_{j \in [q+q']}, \left\{\widetilde{h}^*_j\right\}_{j \in [q+q']}, \left\{(\widehat{\mathbf{g}}_j)_{-(2\eta-1)}\right\}_{j \in [q]}, \left\{(\widetilde{\mathbf{g}}_j)_{-2\eta}\right\}_{j \in [q]}\right),$$

$$T_0 := \{\mathbf{h}_j\}_{j\in[q']},\ T_1 := \left\{\mathbf{h}_j \cdot \boxed{(\widehat{h}_j^{**})^{\mathbf{e}_{2\eta-1}} \cdot (\widetilde{h}_j^{**})^{\mathbf{e}_{2\eta}}}\right\}_{j\in[q']}$$

and $\widehat{h}_j^* \leftarrow \widehat{\mathsf{SampH}}^*(\text{PP}, \text{SP})$, $\widetilde{h}_j^* \leftarrow \widetilde{\mathsf{SampH}}^*(\text{PP}, \text{SP})$, $\widehat{\mathbf{g}}_j \leftarrow \widehat{\mathsf{SampG}}(\text{PP}, \text{SP})$, $\widetilde{\mathbf{g}}_j \leftarrow \widetilde{\mathsf{SampG}}(\text{PP}, \text{SP})$, $\mathbf{h}_j \leftarrow \mathsf{SampH}(\text{PP})$, $\widehat{h}_j^{**} \leftarrow \widehat{\mathsf{SampH}}^*(\text{PP}, \text{SP})$, $\widetilde{h}_j^{**} \leftarrow \widetilde{\mathsf{SampH}}^*(\text{PP}, \text{SP})$. We let $\mathsf{Adv}^{\text{NH}}_{\mathcal{A}}(k, q, q') := \max_{\eta\in[\lfloor n/2\rfloor]}\left\{\mathsf{Adv}^{\text{NH}(\eta)}_{\mathcal{A}}(k, q, q')\right\}$.

Remark 1 (notations). ENDSG is mainly defined for building IBE. We remark that, in the description of LS1, LS2, and NH, the parameter q and q' roughly correspond to the maximum number of challenge queries and key extraction queries, respectively.

Remark 2 (sampling $\widehat{h}^*$ and $\widetilde{h}^*$, and $\mathbb{H}$-subgroup). We model the process of sampling over subgroup generated by $\widehat{h}^*$ and $\widetilde{h}^*$ (in [21]) as algorithm $\widehat{\mathsf{SampH}}^*$ and $\widetilde{\mathsf{SampH}}^*$, respectively. This allows us to employ more complex algebraic structure (say, subspaces of higher dimensions), which is crucial for our prime-order instantiation in Section 4. Accordingly, we extend $\mathbb{H}$-subgroup property to take $\widehat{\mathsf{SampH}}^*$ and $\widetilde{\mathsf{SampH}}^*$ into account.

Remark 3 ($\mathbb{G}$-subgroup and $\mathbb{H}$-subgroup). Since we provide adequate samples of $\mathbb{G}^{n+1}$ directly in the last three computational security requirements and further re-randomization is not necessary in the proof, the $\mathbb{G}$-subgroup in the original definition could be safely removed. However this won't let the revised ENDSG free from $\mathbb{H}$-subgroup property. The simulator still need the property to re-randomize T_0 or T_1 in NH(η) using $\mathsf{SampH}(\text{PP})$ to maintain the consistency of truly random functions on two identities sharing the same η-bit prefix.

On one hand, our revised definition for ENDSG is essentially consistent with Hofheinz *et al.*'s definition [21]. In particular, it is not hard to see that one may use Hofheinz *et al.*'s ENDSG [21] to realize this revised version. Therefore their instantiation using composite-order bilinear groups can also be taken as an instantiation of the revised version above. On the other hand, our revised definition still almost-tightly implies an IBE in the MIMC setting. In fact, the construction, the security result and its proof are nearly the same as those presented in [21]. One may consider them as rewriting Hofheinz *et al.*'s results [21] in the language of our revised ENDSG. We present the construction and sketch of the proof in the full version of the paper. It is worth noting that the construction only achieves weak adaptive security. We will show how to enhance *non-degeneracy* to reach full adaptive security in Section 6.

4 Instantiating ENDSG from *d*-Linear Assumption

This section gives an instantiation of our revised ENDSG (defined in Section 3) using prime-order bilinear groups. See Section 1 for more motivation.

4.1 Prime-order Bilinear Groups and Computational Assumptions

A prime-order bilinear group generator $\mathsf{GrpGen}(1^k)$ takes security parameter 1^k as input and outputs $\mathcal{G} := (p, G_1, G_2, G_T, e)$, where G_1, G_2 and G_T are finite cyclic groups of prime order p, and $e : G_1 \times G_2 \to G_T$ is a non-degenerated and efficiently computable bilinear map. We let g_1, g_2 and $g_T := e(g_1, g_2)$ be a generator of G_1, G_2 and G_T, respectively. We state the (standard) d-linear assumption (d-Lin) in G_1 (see Assumption 1), the analogous assumption in G_2 can be defined by exchanging the role of G_1 and G_2.

Assumption 1 (d-Linear Assumption in G_1). For any p.p.t. adversary $\mathcal{A}$, the following advantage function is negligible in k,

$$\mathsf{Adv}^{d\text{-Lin}}_{\mathcal{A}}(k) := \left|\Pr[\mathcal{A}(D, T_0) = 1] - \Pr[\mathcal{A}(D, T_1) = 1]\right|,$$

where

$$D := \left(\mathcal{G}, g_1, g_2, g_1^{a_1}, \ldots, g_1^{a_d}, g_1^{a_{d+1}}, g_1^{a_1 s_1}, \ldots, g_1^{a_d s_d}\right),$$

$$T_0 := g_1^{a_{d+1}(s_1+\cdots+s_d)},\ T_1 := g_1^{a_{d+1}(s_1+\cdots+s_d)+\boxed{s_{d+1}}}$$

and $\mathcal{G} \leftarrow \mathsf{GrpGen}(1^k)$, $a_1, \ldots, a_d, a_{d+1}, s_{d+1} \leftarrow \mathbb{Z}_p^*$ and $s_1, \ldots, s_d \leftarrow \mathbb{Z}_p$.

"Matrix-in-the-exponent" Notation. For an $m \times n$ matrix $\mathbf{X} = (x_{i,j})$ over $\mathbb{Z}_p$ and a group element g of G, we define $g^{\mathbf{X}} := (g^{x_{i,j}})$, an $m \times n$ matrix over G. We extend pairing e as: given two matrices $\mathbf{A} \in \mathbb{Z}_p^{t\times m}$ and $\mathbf{B} \in \mathbb{Z}_p^{t\times n}$, we define $e(g_1^{\mathbf{A}}, g_2^{\mathbf{B}}) := e(g_1, g_2)^{\mathbf{A}^\top \mathbf{B}} \in G_T^{m\times n}$. For vectors $\mathbf{x}$ and $\mathbf{y}$ over $\mathbb{Z}_p$ of the same length, we have $e(g_1^{\mathbf{x}}, g_2^{\mathbf{y}}) := e(g_1, g_2)^{\mathbf{x}^\top \mathbf{y}} \in G_T$, the standard inner product $\langle \mathbf{x}, \mathbf{y} \rangle$ in the exponent. We will use $\mathbf{0}$ to denote both vectors and matrices with only zero entries, and give out its dimension or size in the subscript if necessary.

An extended version of d-Lifted Linear Assumption. We describe an extension of the d-Lifted Linear (d-LLin) assumption [23] for improving the readability of our proofs, which is called (d, ℓ, q)-Lifted Linear ((d, ℓ, q)-LLin) Assumption. We present the assumption in G_1 and the counterpart in G_2 is readily derived. We then give Lemma 1 showing that the (d, ℓ, q)-LLin assumption is tightly implied by the d-Lin assumption following [23, 15]. The proof could be found in the full version of the paper. We remark that, since ℓ corresponds to a relatively small parameter, say 2, in our construction and q corresponds to the amount of adversary's queries which may be 2^{30}, we prove the Lemma under the assumption that $\ell < q$ for simplicity.

Assumption 2 ((d, ℓ, q)-Lifted Linear Assumption in G_1). For any p.p.t. adversary $\mathcal{A}$, the following advantage function is negligible in k,

$$\mathsf{Adv}^{(d,\ell,q)\text{-LLin}}_{\mathcal{A}}(k) := \left|\Pr[\mathcal{A}(D, T_0) = 1] - \Pr[\mathcal{A}(D, T_1) = 1]\right|,$$

where

$$D := \left(\mathcal{G}, g_1, g_2, g_1^{a_1}, \ldots, g_1^{a_d}, \left\{g_1^{b_{i,j}}\right\}_{i\in[\ell], j\in[d]}, \left\{g_1^{a_1 s_{1,j}}, \ldots, g_1^{a_d s_{d,j}}\right\}_{j\in[q]}\right),$$

$T_0 := \left\{ g_1^{b_{i,1}s_{1,j}+\cdots+b_{i,d}s_{d,j}} \right\}_{i\in[\ell],j\in[q]}$, $T_1 := \left\{ g_1^{b_{i,1}s_{1,j}+\cdots+b_{i,d}s_{d,j}+\boxed{s_{d+i,j}}} \right\}_{i\in[\ell],j\in[q]}$

and $\mathcal{G} \leftarrow \mathsf{GrpGen}(1^k)$, $a_1, \ldots, a_d, b_{i,j}, s_{d+i,j} \leftarrow \mathbb{Z}_p^*$, $s_{1,j}, \ldots, s_{d,j} \leftarrow \mathbb{Z}_p$.

Lemma 1 (d-Lin $\Rightarrow$ (d,ℓ,q)-LLin). *For any p.p.t. adversary $\mathcal{A}$, there exists an adversary $\mathcal{B}$ such that*

$$\mathsf{Adv}_{\mathcal{A}}^{(d,\ell,q)\text{-LLin}}(k) \leqslant \ell \cdot \mathsf{Adv}_{\mathcal{B}}^{d\text{-Lin}}(k) + 1/(p-1),$$

and $\mathsf{Time}(\mathcal{B}) \approx \mathsf{Time}(\mathcal{A}) + \ell^2 d \cdot \mathsf{poly}(k)$ *where* $\mathsf{poly}(k)$ *is independent of* $\mathsf{Time}(\mathcal{A})$.

4.2 Construction

We let $\pi_{\mathrm{L}}(\cdot)$, $\pi_{\mathrm{M}}(\cdot)$, and $\pi_{\mathrm{R}}(\cdot)$ be functions mapping from a $3d \times 3d$ matrix to its left-most d columns, its middle d columns, and its right-most d columns, respectively. Algorithms of our revised ENDSG are shown as follows.

- $\mathsf{SampP}(1^k, n)$: Run $(p, G_1, G_2, G_T, e) \leftarrow \mathsf{GrpGen}(1^k)$ and set $(\mathbb{G}, \mathbb{H}, \mathbb{G}_T, e) := (G_1^{3d}, G_2^{3d}, G_T, e)$. Sample $\mathbf{B}, \mathbf{R} \leftarrow \mathrm{GL}_{3d}(\mathbb{Z}_p)$ and $\mathbf{A}_1, \ldots, \mathbf{A}_n \leftarrow \mathbb{Z}_p^{3d\times 3d}$. Set $\mathbf{B}^* := (\mathbf{B}^{-1})^\top$. Define

$$\begin{array}{l} \mathbf{D} := \pi_{\mathrm{L}}(\mathbf{B}),\ \mathbf{D}_i = \pi_{\mathrm{L}}(\mathbf{B}\mathbf{A}_i);\ \ \mathbf{E} := \pi_{\mathrm{M}}(\mathbf{B}),\ \mathbf{E}_i = \pi_{\mathrm{M}}(\mathbf{B}\mathbf{A}_i); \\ \mathbf{D}^* := \mathbf{B}^*\mathbf{R},\ \mathbf{D}_i^* = \mathbf{B}^*\mathbf{A}_i^\top\mathbf{R};\ \ \mathbf{F} := \pi_{\mathrm{R}}(\mathbf{B}),\ \mathbf{F}_i = \pi_{\mathrm{R}}(\mathbf{B}\mathbf{A}_i); \end{array}$$

  for $i \in [n]$. Define $\mu(g_2^{\mathbf{k}}) := e(g_1^{\mathbf{D}}, g_2^{\mathbf{k}}) = e(g_1, g_2)^{\mathbf{D}^\top\mathbf{k}}$ for all $\mathbf{k} \in \mathbb{Z}_p^{3d}$. Output

$$\text{PP} := \begin{pmatrix} g_1^{\mathbf{D}},\ g_1^{\mathbf{D}_1}, \ldots, g_1^{\mathbf{D}_n} \\ g_2^{\mathbf{D}^*}, g_2^{\mathbf{D}_1^*}, \ldots, g_2^{\mathbf{D}_n^*} \end{pmatrix} \text{ and } \text{SP} := \begin{pmatrix} g_2^{\pi_{\mathrm{M}}(\mathbf{B}^*)}, g_1^{\mathbf{E}}, g_1^{\mathbf{E}_1}, \ldots, g_1^{\mathbf{E}_n} \\ g_2^{\pi_{\mathrm{R}}(\mathbf{B}^*)}, g_1^{\mathbf{F}}, g_1^{\mathbf{F}_1}, \ldots, g_1^{\mathbf{F}_n} \end{pmatrix}.$$

  We assume PP always contains $\mathbb{G}, \mathbb{H}, \mathbb{G}_T, e, \mu$ and group order p.
- $\mathsf{SampGT}(g_T^{\mathbf{p}})$: Sample $\mathbf{s} \leftarrow \mathbb{Z}_p^d$ and output $g_T^{\mathbf{s}^\top\mathbf{p}} \in G_T$.
- $\mathsf{SampG}(\text{PP})$: Sample $\mathbf{s} \leftarrow \mathbb{Z}_p^d$ and output $\left(g_1^{\mathbf{D}\mathbf{s}}, g_1^{\mathbf{D}_1\mathbf{s}}, \ldots, g_1^{\mathbf{D}_n\mathbf{s}}\right) \in (G_1^{3d})^{n+1}$.
- $\mathsf{SampH}(\text{PP})$: Sample $\mathbf{r} \leftarrow \mathbb{Z}_p^{3d}$ and output $\left(g_2^{\mathbf{D}^*\mathbf{r}}, g_2^{\mathbf{D}_1^*\mathbf{r}}, \ldots, g_2^{\mathbf{D}_n^*\mathbf{r}}\right) \in (G_2^{3d})^{n+1}$.
- $\widehat{\mathsf{SampG}}(\text{PP}, \text{SP})$: Sample $\widehat{\mathbf{s}} \leftarrow \mathbb{Z}_p^d$ and output $\left(g_1^{\mathbf{E}\widehat{\mathbf{s}}}, g_1^{\mathbf{E}_1\widehat{\mathbf{s}}}, \ldots, g_1^{\mathbf{E}_n\widehat{\mathbf{s}}}\right) \in (G_1^{3d})^{n+1}$.
- $\widetilde{\mathsf{SampG}}(\text{PP}, \text{SP})$: Sample $\widetilde{\mathbf{s}} \leftarrow \mathbb{Z}_p^d$ and output $\left(g_1^{\mathbf{F}\widetilde{\mathbf{s}}}, g_1^{\mathbf{F}_1\widetilde{\mathbf{s}}}, \ldots, g_1^{\mathbf{F}_n\widetilde{\mathbf{s}}}\right) \in (G_1^{3d})^{n+1}$.
- $\widehat{\mathsf{SampH}}^*(\text{PP}, \text{SP})$: Sample $\widehat{\mathbf{r}} \leftarrow \mathbb{Z}_p^d$ and output $g_2^{\pi_{\mathrm{M}}(\mathbf{B}^*)\widehat{\mathbf{r}}} \in G_2^{3d}$.
- $\widetilde{\mathsf{SampH}}^*(\text{PP}, \text{SP})$: Sample $\widetilde{\mathbf{r}} \leftarrow \mathbb{Z}_p^d$ and output $g_2^{\pi_{\mathrm{R}}(\mathbf{B}^*)\widetilde{\mathbf{r}}} \in G_2^{3d}$.

4.3 Security Analysis

One can easily check the *projective*, *associative*, *orthogonality*, *non-degeneracy*, $\mathbb{H}$-*subgroup*, and *LS1* properties following [15]. Due to lack of space, we just give the proof of *left subgroup indistinguishability 2 (LS2)* and sketch the proof of *nested-hiding indistinguishability (NH)*, and leave detailed proofs in the full version of the paper. We emphasize that all three computational properties are *tightly* reduced to the d-Lin assumption.

Left subgroup indistinguishability 2. We first rewrite entries involved in the LS2 advantage function $\mathsf{Adv}^{\mathrm{LS2}}_{\mathcal{A}}(k,q,q')$ in terms of $\mathbf{B},\mathbf{B}^*,\mathbf{A}_i,\mathbf{R}$ as follows

$$\mathrm{PP} := \begin{pmatrix} g_1^{\pi_{\mathrm{L}}(\mathbf{B})}, g_1^{\pi_{\mathrm{L}}(\mathbf{B}\mathbf{A}_1)}, \ldots, g_1^{\pi_{\mathrm{L}}(\mathbf{B}\mathbf{A}_n)} \\ g_2^{\mathbf{B}^*\mathbf{R}}, \; g_2^{\mathbf{B}^*\mathbf{A}_1^\top\mathbf{R}}, \; \ldots, g_2^{\mathbf{B}^*\mathbf{A}_n^\top\mathbf{R}} \end{pmatrix};$$

$$\widehat{h}_j^* \cdot \widetilde{h}_j^* := g_2^{\mathbf{B}^*\begin{pmatrix}\mathbf{0}_d \\ \widehat{\mathbf{r}}_j \\ \widetilde{\mathbf{r}}_j\end{pmatrix}};$$

$$\mathbf{g}'_j \cdot \widehat{\mathbf{g}}'_j := \Big(g_1^{\mathbf{B}\begin{pmatrix}\mathbf{s}'_j \\ \widehat{\mathbf{s}}'_j \\ \mathbf{0}_d\end{pmatrix}}, g_1^{\mathbf{B}\mathbf{A}_1\begin{pmatrix}\mathbf{s}'_j \\ \widehat{\mathbf{s}}'_j \\ \mathbf{0}_d\end{pmatrix}}, \ldots, g_1^{\mathbf{B}\mathbf{A}_n\begin{pmatrix}\mathbf{s}'_j \\ \widehat{\mathbf{s}}'_j \\ \mathbf{0}_d\end{pmatrix}}\Big);$$

$$\mathbf{g}_j \cdot \widehat{\mathbf{g}}_j := \Big(g_1^{\mathbf{B}\begin{pmatrix}\mathbf{s}_j \\ \widehat{\mathbf{s}}_j \\ \mathbf{0}_d\end{pmatrix}}, g_1^{\mathbf{B}\mathbf{A}_1\begin{pmatrix}\mathbf{s}_j \\ \widehat{\mathbf{s}}_j \\ \mathbf{0}_d\end{pmatrix}}, \ldots, g_1^{\mathbf{B}\mathbf{A}_n\begin{pmatrix}\mathbf{s}_j \\ \widehat{\mathbf{s}}_j \\ \mathbf{0}_d\end{pmatrix}}\Big);$$

$$\mathbf{g}_j \cdot \widetilde{\mathbf{g}}_j := \Big(g_1^{\mathbf{B}\begin{pmatrix}\mathbf{s}_j \\ \mathbf{0}_d \\ \widetilde{\mathbf{s}}_j\end{pmatrix}}, g_1^{\mathbf{B}\mathbf{A}_1\begin{pmatrix}\mathbf{s}_j \\ \mathbf{0}_d \\ \widetilde{\mathbf{s}}_j\end{pmatrix}}, \ldots, g_1^{\mathbf{B}\mathbf{A}_n\begin{pmatrix}\mathbf{s}_j \\ \mathbf{0}_d \\ \widetilde{\mathbf{s}}_j\end{pmatrix}}\Big);$$

where $\widehat{\mathbf{r}}_j, \widetilde{\mathbf{r}}_j, \mathbf{s}'_j, \widehat{\mathbf{s}}'_j, \mathbf{s}_j, \widehat{\mathbf{s}}_j, \widetilde{\mathbf{s}}_j \leftarrow \mathbb{Z}_p^d$. Then we prove the following lemma.

Lemma 2 ((d,d,q)-LLin $\Rightarrow$ LS2). *For any p.p.t. adversary $\mathcal{A}$, there exists an adversary $\mathcal{B}$ such that*

$$\mathsf{Adv}^{\mathrm{LS2}}_{\mathcal{A}}(k,q,q') \leqslant 2 \cdot \mathsf{Adv}^{(d,d,q)\text{-LLin}}_{\mathcal{B}}(k),$$

and $\mathsf{Time}(\mathcal{B}) \approx \mathsf{Time}(\mathcal{A}) + (q+q')d^2 \cdot \mathsf{poly}(k,n)$. *($\mathsf{poly}(k,n)$ is independent of $\mathcal{A}$)*

Overview of Proof. We will prove Lemma 2 in two steps with the help of a transitional distribution $T_{1/2} = \{\mathbf{g}_j \cdot \widehat{\mathbf{g}}_j \cdot \widetilde{\mathbf{g}}_j\}_{j\in[q]}$ where

$$\mathbf{g}_j \cdot \widehat{\mathbf{g}}_j \cdot \widetilde{\mathbf{g}}_j := \Big(g_1^{\mathbf{B}\begin{pmatrix}\mathbf{s}_j \\ \widehat{\mathbf{s}}_j \\ \widetilde{\mathbf{s}}_j\end{pmatrix}}, g_1^{\mathbf{B}\mathbf{A}_1\begin{pmatrix}\mathbf{s}_j \\ \widehat{\mathbf{s}}_j \\ \widetilde{\mathbf{s}}_j\end{pmatrix}}, \ldots, g_1^{\mathbf{B}\mathbf{A}_n\begin{pmatrix}\mathbf{s}_j \\ \widehat{\mathbf{s}}_j \\ \widetilde{\mathbf{s}}_j\end{pmatrix}}\Big).$$

In particular, we prove that, given D, distribution T_0 and $T_{1/2}$ are computational indistinguishable under the (d,d,q)-LLin assumption (see Lemma 3), and so do $T_{1/2}$ and T_1 (see Lemma 4). These immediately prove Lemma 2.

Lemma 3 (from T_0 to $T_{1/2}$). *For any p.p.t. adversary $\mathcal{A}$, there exists an adversary $\mathcal{B}$ such that*

$$\big|\Pr[\mathcal{A}(D,T_0)=1] - \Pr[\mathcal{A}(D,T_{1/2})=1]\big| \leqslant \mathsf{Adv}^{(d,d,q)\text{-LLin}}_{\mathcal{B}}(k),$$

and $\mathsf{Time}(\mathcal{B}) \approx \mathsf{Time}(\mathcal{A}) + (q+q')d^2 \cdot \mathsf{poly}(k,n)$. *($\mathsf{poly}(k,n)$ is independent of $\mathcal{A}$)*

Proof. Given an instance of (d,d,q)-LLin problem (i.e., set $\ell = d$)

$$\left(g_1, g_2, g_1^{a_1}, \ldots, g_1^{a_d}, \left\{g_1^{b_{i,j}}\right\}_{i,j\in[d]}, \left\{g_1^{a_1 s_{1,j}}, \ldots, g_1^{a_d s_{d,j}}\right\}_{j\in[q]}, \left\{g_1^{b_{i,1}s_{1,j}+\cdots+b_{i,d}s_{d,j}+s_{d+i,j}}\right\}_{i\in[d],j\in[q]}\right)$$

as input where either $s_{d+i,j} = 0$ or $s_{d+i,j} \leftarrow \mathbb{Z}_p^*$, adversary $\mathcal{B}$ works as follows:

Programming $\widehat{\mathbf{s}}_j$ and $\widetilde{\mathbf{s}}_j$ for $j \in [q]$. Adversary $\mathcal{B}$ implicitly sets

$$\widehat{\mathbf{s}}_j := (s_{1,j}, \ldots, s_{d,j})^\top \quad \text{and} \quad \widetilde{\mathbf{s}}_j := (s_{d+1,j}, \ldots, s_{2d,j})^\top.$$

Programming $\mathbf{B}, \mathbf{B}^*, \mathbf{A}_1, \ldots, \mathbf{A}_n, \mathbf{R}$. We define $\mathbf{W}$ as

$$\mathbf{W} := \left(\begin{array}{ccc|ccc} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ \hline & & & a_1 & & \\ & & & & \ddots & \\ & & & & & a_d \\ \hline & & & b_{1,1} & \cdots & b_{1,d} \\ & & & \vdots & & \vdots \\ & & & b_{d,1} & \cdots & b_{d,d} \end{array}\right. \left.\begin{array}{ccc} & & \\ & & \\ & & \\ \hline & & \\ & & \\ & & \\ \hline 1 & & \\ & \ddots & \\ & & 1 \end{array}\right) \in \mathbb{Z}_p^{3d \times 3d}$$

and set $\mathbf{W}^* := (\mathbf{W}^{-1})^\top$. Sample[7] $\bar{\mathbf{B}}, \bar{\mathbf{R}} \leftarrow \mathrm{GL}_{3d}(\mathbb{Z}_p)$ and set $\bar{\mathbf{B}}^* := (\bar{\mathbf{B}}^{-1})^\top$. Also sample $\bar{\mathbf{A}}_1, \ldots, \bar{\mathbf{A}}_n \leftarrow \mathbb{Z}_p^{3d \times 3d}$, and implicitly set

$$(\mathbf{B}, \mathbf{B}^*) := (\bar{\mathbf{B}}\mathbf{W}, \bar{\mathbf{B}}^*\mathbf{W}^*), \quad \mathbf{R} := \mathbf{W}^\top \bar{\mathbf{R}}, \quad \mathbf{A}_i := \mathbf{W}^{-1}\bar{\mathbf{A}}_i\mathbf{W}, \tag{1}$$

for $i \in [n]$. Observe that $\mathbf{B}, \mathbf{B}^*, \mathbf{R}$ and all $\mathbf{A}_i$ are distributed properly, and

$$\mathbf{B}\mathbf{A}_i = \bar{\mathbf{B}}\bar{\mathbf{A}}_i\mathbf{W}, \quad \mathbf{B}^*\mathbf{R} = \bar{\mathbf{B}}^*\bar{\mathbf{R}}, \quad \mathbf{B}^*\mathbf{A}_i^\top\mathbf{R} = \bar{\mathbf{B}}^*\bar{\mathbf{A}}_i^\top\bar{\mathbf{R}}. \tag{2}$$

Simulating pp. $\mathcal{B}$ can simulate

$$g_1^{\pi_{\mathrm{L}}(\mathbf{B})} = g_1^{\pi_{\mathrm{L}}(\bar{\mathbf{B}}\mathbf{W})} = g_1^{\bar{\mathbf{B}}\pi_{\mathrm{L}}(\mathbf{W})} \quad \text{and} \quad g_1^{\pi_{\mathrm{L}}(\mathbf{B}\mathbf{A}_i)} = g_1^{\pi_{\mathrm{L}}(\bar{\mathbf{B}}\bar{\mathbf{A}}_i\mathbf{W})} = g_1^{\bar{\mathbf{B}}\bar{\mathbf{A}}_i\pi_{\mathrm{L}}(\mathbf{W})},$$
$$g_2^{\mathbf{B}^*\mathbf{R}} = g_2^{\bar{\mathbf{B}}^*\bar{\mathbf{R}}} \quad \text{and} \quad g_2^{\mathbf{B}^*\mathbf{A}_i^\top\mathbf{R}} = g_2^{\bar{\mathbf{B}}^*\bar{\mathbf{A}}_i^\top\bar{\mathbf{R}}},$$

for $i \in [n]$ using the knowledge of $\pi_{\mathrm{L}}(\mathbf{W})$ and $\bar{\mathbf{B}}, \bar{\mathbf{B}}^*, \bar{\mathbf{A}}_1, \ldots, \bar{\mathbf{A}}_n, \bar{\mathbf{R}}$.

Simulating $\widehat{h}_j^* \cdot \widetilde{h}_j^*$ for $j \in [q + q']$. It is not hard to compute $\mathbf{W}^* \in \mathbb{Z}_p^{3d \times 3d}$ as

$$\mathbf{W}^* := \left(\begin{array}{ccc|ccc} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ \hline & & & a_1^{-1} & & \\ & & & & \ddots & \\ & & & & & a_d^{-1} \\ \hline & & & 1 & & \\ & & & & \ddots & \\ & & & & & 1 \end{array}\right. \left.\begin{array}{ccc} & & \\ & & \\ & & \\ \hline -a_1^{-1}b_{1,1} & \cdots & -a_1^{-1}b_{d,1} \\ \vdots & & \vdots \\ -a_d^{-1}b_{1,d} & \cdots & -a_d^{-1}b_{d,d} \\ \hline & & \\ & & \\ & & \end{array}\right).$$

[7] In our symbol system, a variable with a bar on the top, say $\bar{\mathbf{B}}$, is sampled by the simulator (i.e., $\mathcal{B}$) and is completely known to it.

For all $j \in [q+q']$, we sample $\bar{\mathbf{r}}_j \leftarrow \mathbb{Z}_p^{2d}$ and implicitly set

$$\begin{pmatrix} \mathbf{0}_d \\ \widehat{\mathbf{r}}_j \\ \widetilde{\mathbf{r}}_j \end{pmatrix} = (\mathbf{W}^*)^{-1} \begin{pmatrix} \mathbf{0}_d \\ \bar{\mathbf{r}}_j \end{pmatrix} = \mathbf{W}^\top \begin{pmatrix} \mathbf{0}_d \\ \bar{\mathbf{r}}_j \end{pmatrix}.$$

Since the right-bottom $2d \times 2d$ sub-matrix of $\mathbf{W}^*$ is full-rank with overwhelming probability, $\widehat{\mathbf{r}}_j$ and $\widetilde{\mathbf{r}}_j$ are distributed properly and $\mathcal{B}$ can simulate

$$\widehat{h}_j^* \cdot \widetilde{h}_j^* = g_2^{\mathbf{B}^* \begin{pmatrix} \mathbf{0}_d \\ \widehat{\mathbf{r}}_j \\ \widetilde{\mathbf{r}}_j \end{pmatrix}} = g_2^{\bar{\mathbf{B}}^* \mathbf{W}^* \begin{pmatrix} \mathbf{0}_d \\ \widehat{\mathbf{r}}_j \\ \widetilde{\mathbf{r}}_j \end{pmatrix}} = g_2^{\bar{\mathbf{B}}^* \begin{pmatrix} \mathbf{0}_d \\ \bar{\mathbf{r}}_j \end{pmatrix}}$$

using the knowledge of $\bar{\mathbf{B}}^*$ and $\bar{\mathbf{r}}_j$.

Simulating $\mathbf{g}'_j \cdot \widehat{\mathbf{g}}'_j$ for $j \in [q]$. $\mathcal{B}$ can sample $\mathbf{s}'_j, \widehat{\mathbf{s}}'_j \leftarrow \mathbb{Z}_p^d$ and simulate

$$g_1^{\mathbf{B} \begin{pmatrix} \mathbf{s}'_j \\ \widehat{\mathbf{s}}'_j \\ \mathbf{0}_d \end{pmatrix}} = g_1^{\bar{\mathbf{B}} \mathbf{W} \begin{pmatrix} \mathbf{s}'_j \\ \widehat{\mathbf{s}}'_j \\ \mathbf{0}_d \end{pmatrix}} \quad \text{and} \quad g_1^{\mathbf{B} \mathbf{A}_i \begin{pmatrix} \mathbf{s}'_j \\ \widehat{\mathbf{s}}'_j \\ \mathbf{0}_d \end{pmatrix}} = g_1^{\bar{\mathbf{B}} \bar{\mathbf{A}}_i \mathbf{W} \begin{pmatrix} \mathbf{s}'_j \\ \widehat{\mathbf{s}}'_j \\ \mathbf{0}_d \end{pmatrix}}$$

for $i \in [n]$ and using the knowledge of $g_1^{\mathbf{W}}$ and $\bar{\mathbf{B}}, \bar{\mathbf{A}}_1, \ldots, \bar{\mathbf{A}}_n$.

Simulating the challenge. Algorithm $\mathcal{B}$ can sample $\mathbf{s}_j \leftarrow \mathbb{Z}_p^d$ and simulate

$$g_1^{\mathbf{B} \begin{pmatrix} \mathbf{s}_j \\ \widehat{\mathbf{s}}_j \\ \widetilde{\mathbf{s}}_j \end{pmatrix}} = g_1^{\bar{\mathbf{B}} \mathbf{W} \begin{pmatrix} \mathbf{s}_j \\ \widehat{\mathbf{s}}_j \\ \widetilde{\mathbf{s}}_j \end{pmatrix}} \quad \text{and} \quad g_1^{\mathbf{B} \mathbf{A}_i \begin{pmatrix} \mathbf{s}_j \\ \widehat{\mathbf{s}}_j \\ \widetilde{\mathbf{s}}_j \end{pmatrix}} = g_1^{\bar{\mathbf{B}} \bar{\mathbf{A}}_i \mathbf{W} \begin{pmatrix} \mathbf{s}_j \\ \widehat{\mathbf{s}}_j \\ \widetilde{\mathbf{s}}_j \end{pmatrix}}$$

for $i \in [n]$ and $j \in [q]$ using the knowledge of $\bar{\mathbf{B}}, \bar{\mathbf{A}}_1, \ldots, \bar{\mathbf{A}}_n$ and

$$g_1^{\mathbf{W} \begin{pmatrix} \mathbf{s}_j \\ \widehat{\mathbf{s}}_j \\ \widetilde{\mathbf{s}}_j \end{pmatrix}} = g_1^{\begin{pmatrix} \mathbf{s}_j \\ a_1 s_{1,j} \\ \vdots \\ a_d s_{d,j} \\ b_{1,1} s_{1,j} + \cdots + b_{1,d} s_{d,j} + s_{d+1,j} \\ \vdots \\ b_{d,1} s_{1,j} + \cdots + b_{d,d} s_{d,j} + s_{2d,j} \end{pmatrix}}.$$

Analysis. Observe that if all $s_{d+i,j} = 0$, then all $\widetilde{\mathbf{s}}_j = \mathbf{0}$ and the output challenge is distributed as $\{\mathbf{g}_j \cdot \widehat{\mathbf{g}}_j\}_{j \in [q]}$; otherwise, if all $s_{d+i,j} \leftarrow \mathbb{Z}_p^*$, then all $\widetilde{\mathbf{s}}_j \leftarrow (\mathbb{Z}_p^*)^d$ and the output challenge is distributed as $\{\mathbf{g}_j \cdot \widehat{\mathbf{g}}_j \cdot \widetilde{\mathbf{g}}_j\}_{j \in [q]}$. Therefore we may conclude that $\left|\Pr[\mathcal{A}(D, T_0) = 1] - \Pr[\mathcal{A}(D, T_{1/2}) = 1]\right| \leqslant \mathsf{Adv}_{\mathcal{B}}^{(d,d,q)\text{-LLin}}(k)$. □

Lemma 4 (from $T_{1/2}$ to T_1). *For any p.p.t. adversary $\mathcal{A}$, there exists an adversary $\mathcal{B}$ such that*

$$\left|\Pr[\mathcal{A}(D, T_{1/2}) = 1] - \Pr[\mathcal{A}(D, T_1) = 1]\right| \leqslant \mathsf{Adv}_{\mathcal{B}}^{(d,d,q)\text{-LLin}}(k),$$

and $\mathsf{Time}(\mathcal{B}) \approx \mathsf{Time}(\mathcal{A}) + (q+q')d^2 \cdot \mathsf{poly}(k, n)$. *($\mathsf{poly}(k, n)$ is independent of $\mathcal{A}$)*

Proof. Given an instance of (d, d, q)-LLin problem, adversary $\mathcal{B}$ behaves in a similar manner to $\mathcal{B}$ in the proof of Lemma 3 with the differences that:

Programming $\widehat{\mathbf{s}}_j$ and $\widetilde{\mathbf{s}}_j$ for $j \in [q]$. Adversary $\mathcal{B}$ implicitly sets

$$\widehat{\mathbf{s}}_j = (s_{2d,j}, \ldots, s_{d+1,j})^\top \text{ and } \widetilde{\mathbf{s}}_j = (s_{d,j}, \ldots, s_{1,j})^\top.$$

Defining $\mathbf{W}$. Adversary $\mathcal{B}$ defines $\mathbf{W}$ as

$$\mathbf{W} := \left(\begin{array}{c|c|c} 1 & & \\ & \ddots & \\ & 1 & \\ \hline & 1 & b_{d,d} \cdots b_{d,1} \\ & \ddots & \vdots \quad \vdots \\ & 1 & b_{1,d} \cdots b_{1,1} \\ \hline & & a_d \\ & & \ddots \\ & & a_1 \end{array}\right) \in \mathbb{Z}_p^{3d \times 3d}.$$

In fact, $\mathbf{B}, \mathbf{B}^*, \mathbf{A}_i, \mathbf{R}$ are programmed as Eq. (1). All entries in PP and $\{\mathbf{g}'_j \cdot \widehat{\mathbf{g}}'_j\}$ can be simulated exactly as in the proof of Lemma 3. The strategy for creating $\{\widehat{h}^*_j \cdot \widetilde{h}^*_j\}$ and the challenge there also works well. □

Combining Lemma 1 and Lemma 2, we have Corollary 1 showing that our instantiation satisfies *left subgroup indistinguishability 2* requirement with tight reduction, i.e., with security loss $2d$, to the d-Lin assumption.

Corollary 1 (d-Lin $\Rightarrow$ LS2). *For any p.p.t. adversary $\mathcal{A}$, there exists an adversary $\mathcal{B}$ such that*

$$\mathsf{Adv}^{\mathrm{LS2}}_{\mathcal{A}}(k, q, q') \leqslant 2d \cdot \mathsf{Adv}^{d\text{-Lin}}_{\mathcal{B}}(k) + 2/(p-1),$$

and $\mathsf{Time}(\mathcal{B}) \approx \mathsf{Time}(\mathcal{A}) + (q+q')d^2 \cdot \mathsf{poly}(k, n)$. *($\mathsf{poly}(k, n)$ is independent of $\mathcal{A}$)*

Nested-hiding indistinguishability. Since $\widehat{h}^{**}_j$ and $\widetilde{h}^{**}_j$ are respective random vectors in d-dimensional subspace $g_2^{\pi_{\mathrm{M}}(\mathbf{B}^*)}$ and $g_2^{\pi_{\mathrm{R}}(\mathbf{B}^*)}$ now, we must "create" more entropy from $\mathbf{h}_j$ than Chen and Wee did in [15]. To do so, we establish a generalized version of many-tuple lemma (see Lemma 5) in [15], which takes the (d, d, d)-LLin assumption as starting point instead of the d-Lin assumption.

Lemma 5 (Generalized Many-Tuple Lemma). *There exists an efficient algorithm that on input $q \in \mathbb{Z}^+$, a finite cyclic group G generated by $g \in G$ and*

$$\left(g, g^{a_1}, \ldots, g^{a_d}, \left\{g^{b_{i,j}}\right\}_{i,j\in[d]}, \left\{g^{a_1 r_{1,j}}, \ldots, g^{a_d r_{d,j}}\right\}_{j\in[d]}, \left\{g^{b_{i,1}r_{1,j}+\cdots+b_{i,d}r_{d,j}+r_{d+i,j}}\right\}_{i,j\in[d]}\right),$$

outputs $\left(g^{\mathbf{VZ}}, g^{\mathbf{Z}}\right)$ for some matrix $\mathbf{V} \in \mathbb{Z}_p^{d\times d}$ along with $\left\{\left(g^{\mathbf{t}_j}, g^{\mathbf{V}\mathbf{t}_j+\boldsymbol{\tau}_j}\right)\right\}_{j\in[q]}$, where $\mathbf{t}_j \leftarrow \mathbb{Z}_p^d$, all $\boldsymbol{\tau}_j$ are either $\mathbf{0}_d$ or uniformly distributed over $\mathbb{Z}_p^d$. And $\mathbf{Z}$ is an invertible diagonal matrix.

Then the proof for the NH property can be obtained by properly embedding matrix $\mathbf{V}$ into $\mathbf{A}_{2\eta-1}$ and $\mathbf{A}_{2\eta}$ and matrix $\mathbf{Z}$ into $\mathbf{R}$, and naturally extending Chen and Wee's simulation strategy [15].

5 Concrete IBE from *d*-Linear Assumption

This section describes the concrete IBE scheme derived from our prime-order instantiation in Section 4 following Hofheinz *et al.*'s framework [21]. Let GrpGen be the bilinear group generator described in Section 4.1 and $\pi_{\mathrm{L}}(\cdot)$ be the function mapping from a $3d \times 3d$ matrix to its left-most d columns.

- $\mathsf{Param}(1^k, n)$: Run $(p, G_1, G_2, G_T, e) \leftarrow \mathsf{GrpGen}(1^k)$. Sample $\mathbf{B}, \mathbf{R} \leftarrow \mathrm{GL}_{3d}(\mathbb{Z}_p)$ and $\mathbf{A}_1, \ldots, \mathbf{A}_{2n} \leftarrow \mathbb{Z}_p^{3d\times 3d}$, and set $\mathbf{B}^* := (\mathbf{B}^{-1})^\top$. Output
$$\mathrm{GP} := \begin{pmatrix} g_1^{\pi_{\mathrm{L}}(\mathbf{B})}, g_1^{\pi_{\mathrm{L}}(\mathbf{B}\mathbf{A}_1)}, \cdots, g_1^{\pi_{\mathrm{L}}(\mathbf{B}\mathbf{A}_{2n})} \\ g_2^{\mathbf{B}^*\mathbf{R}}, \quad g_2^{\mathbf{B}^*\mathbf{A}_1^\top\mathbf{R}}, \quad \cdots, g_2^{\mathbf{B}^*\mathbf{A}_{2n}^\top\mathbf{R}} \end{pmatrix}.$$
- $\mathsf{Setup}(\mathrm{GP})$: Sample $\mathbf{k} \leftarrow \mathbb{Z}_p^{3d}$ and output
$$\mathrm{MPK} := \left(g_1^{\pi_{\mathrm{L}}(\mathbf{B})}, g_1^{\pi_{\mathrm{L}}(\mathbf{B}\mathbf{A}_1)}, \ldots, g_1^{\pi_{\mathrm{L}}(\mathbf{B}\mathbf{A}_{2n})}; e(g_1, g_2)^{\pi_{\mathrm{L}}(\mathbf{B})^\top\mathbf{k}}\right) \in (G_1^{3d\times d})^{2n+1} \times G_T^d;$$
$$\mathrm{MSK} := \left(g_2^{\mathbf{B}^*\mathbf{R}}, g_2^{\mathbf{B}^*\mathbf{A}_1^\top\mathbf{R}}, \ldots, g_2^{\mathbf{B}^*\mathbf{A}_{2n}^\top\mathbf{R}}; g_2^{\mathbf{k}}\right) \in (G_2^{3d\times 3d})^{2n+1} \times G_2^{3d}.$$
- $\mathsf{KeyGen}(\mathrm{MPK}, \mathrm{MSK}, \mathbf{y})$: Let $\mathbf{y} = (y_1, \ldots, y_n) \in \{0,1\}^n$. Sample $\mathbf{r} \leftarrow \mathbb{Z}_p^{3d}$ and output
$$\mathrm{SK}_{\mathbf{y}} := \left(g_2^{\mathbf{B}^*\mathbf{R}\mathbf{r}}, \ g_2^{\mathbf{k}+\mathbf{B}^*(\mathbf{A}_{2-y_1}+\cdots+\mathbf{A}_{2n-y_n})^\top\mathbf{R}\mathbf{r}}\right) \in G_2^{3d} \times G_2^{3d}.$$
- $\mathsf{Enc}(\mathrm{MPK}, \mathbf{x}, \mathrm{M})$: Let $\mathbf{x} = (x_1, \ldots, x_n) \in \{0,1\}^n$ and $\mathrm{M} \in \mathbb{G}_T$. Sample $\mathbf{s} \leftarrow \mathbb{Z}_p^d$ and output
$$\mathrm{CT}_{\mathbf{x}} := \left(g_1^{\pi_{\mathrm{L}}(\mathbf{B})\mathbf{s}}, \ g_1^{\pi_{\mathrm{L}}(\mathbf{B}(\mathbf{A}_{2-x_1}+\cdots+\mathbf{A}_{2n-x_n}))\mathbf{s}}, \ e(g_1, g_2)^{\mathbf{s}^\top\pi_{\mathrm{L}}(\mathbf{B})^\top\mathbf{k}} \cdot \mathrm{M}\right)$$
$$\in G_1^{3d} \times G_1^{3d} \times G_T.$$
- $\mathsf{Dec}(\mathrm{MPK}, \mathrm{SK}, \mathrm{CT})$: Let $\mathrm{SK} = (K_0, K_1)$ and $\mathrm{CT} = (C_0, C_1, C_2)$. Output
$$\mathrm{M} := C_2 \cdot e(C_1, K_0)/e(C_0, K_1).$$

Note that we only put necessary entries for Enc into MPK, while entries from GP (or PP) for running KeyGen are put into MSK. We describe the following theorem.

Theorem 1. *For any p.p.t. adversary $\mathcal{A}$ making at most q_K key extraction queries and at most q_C challenge queries for pairwise distinct challenge identity against at most λ instances, there exists adversary $\mathcal{B}$ such that*
$$\mathsf{Adv}_{\mathcal{A}}^{\mathrm{IBE}}(k, \lambda, q_K, q_C, 1) \leqslant d \cdot (5n+1) \cdot \mathsf{Adv}_{\mathcal{B}}^{d\text{-Lin}}(k) + 2^{-\Omega(k)},$$
where $\mathsf{Time}(\mathcal{B}) \approx \mathsf{Time}(\mathcal{A}) + (\lambda + q_C + q_K) \cdot d^2 \cdot \mathsf{poly}(k, n)$ and $\mathsf{poly}(k, n)$ is independent of $\mathsf{Time}(\mathcal{A})$.

6 Achieving Stronger Security Guarantee

This section will investigate two flavors of stronger adaptive security: *B-weak* and *full* adaptive security (see Section 2) by enhancing the *non-degeneracy* property and updating the proof of "ENDSG implies IBE".

6.1 Warmup: Achieving B-weak Adaptive Security

Recall that the original non-degeneracy property said that:

(Non-degeneracy (Recalled).) Over the probability space defined by $\widehat{g}_0 \leftarrow \widehat{\mathsf{SampG}_0}(\mathrm{PP}, \mathrm{SP})$, with overwhelming probability $1 - 2^{-\Omega(k)}$, $e(\widehat{g}_0, \widehat{h}^*)$ is distributed uniformly over $\mathbb{G}_T$ when sampling $\widehat{h}^* \leftarrow \widehat{\mathsf{SampH}}^*(\mathrm{PP}, \mathrm{SP})$.

We observe that $\widehat{h}^*$ in our prime-order instantiation (see Section 4) actually contains higher entropy than those in Hofheinz *et al.*'s composite-order instantiation [21]. In particular, $\widehat{h}^*$ is uniformly distributed over a d-dimension subspace of G_2^{3d} containing p^d elements (vectors), while $e(\widehat{g}_0, \widehat{h}^*)$ is an element in G_T containing just p elements. This suggests that, given $e(\widehat{g}_0, \widehat{h}^*)$, there may be leftover entropy in $\widehat{h}^*$, and our prime-order instantiation may achieve stronger non-degeneracy even relying on no computational assumption.

To formally investigate the above idea, we describe the notion of *B-bounded non-degeneracy* which roughly ensures the non-degeneracy when a single $\widehat{h}^*$ is paired with at most B $\widehat{g}_0$'s.

(B-bounded non-degeneracy.) Over the probability space defined by sampling $(\widehat{g}_{0,1}, \ldots, \widehat{g}_{0,B}) \leftarrow \widehat{\mathsf{SampG}_0}^B(\mathrm{PP}, \mathrm{SP})$, with overwhelming probability $1 - 2^{-\Omega(k)}$, $(e(\widehat{g}_{0,1}, \widehat{h}^*), \ldots, e(\widehat{g}_{0,B}, \widehat{h}^*))$ is distributed uniformly over $\mathbb{G}_T^B$ when sampling $\widehat{h}^* \leftarrow \widehat{\mathsf{SampH}}^*(\mathrm{PP}, \mathrm{SP})$.

It is obvious that the ENDSG with B-bounded non-degeneracy almost-tightly implies a B-weak adaptively secure IBE in the MIMC setting. We now prove that our prime-order instantiation in Section 4 indeed reaches this stronger version of non-degeneracy.

Lemma 6. *Our prime-order instantiation of ENDSG in Section 4 based on the d-Lin assumption is d-bounded non-degenerated.*

Proof. The proof is just a simple statistical argument extended from the proof for the original non-degeneracy. For $\widehat{\mathbf{s}}_1, \ldots, \widehat{\mathbf{s}}_d \leftarrow \mathbb{Z}_p^d$ and $\widehat{\mathbf{r}} \leftarrow \mathbb{Z}_p^d$, we have that

$$\begin{pmatrix} e(g_1^{\mathbf{E}\widehat{\mathbf{s}}_1}, g_2^{\pi_{\mathrm{M}}(\mathbf{B}^*)\widehat{\mathbf{r}}}) \\ \vdots \\ e(g_1^{\mathbf{E}\widehat{\mathbf{s}}_d}, g_2^{\pi_{\mathrm{M}}(\mathbf{B}^*)\widehat{\mathbf{r}}}) \end{pmatrix} = \begin{pmatrix} e(g_1, g_2)^{\widehat{\mathbf{s}}_1^\top \widehat{\mathbf{r}}} \\ \vdots \\ e(g_1, g_2)^{\widehat{\mathbf{s}}_d^\top \widehat{\mathbf{r}}} \end{pmatrix} = e(g_1, g_2)^{\begin{pmatrix} \widehat{\mathbf{s}}_1^\top \\ \vdots \\ \widehat{\mathbf{s}}_d^\top \end{pmatrix} \widehat{\mathbf{r}}}.$$

With probability at least $1 - \frac{1}{p-1}$, the matrix $(\widehat{\mathbf{s}}_1, \ldots, \widehat{\mathbf{s}}_d)^\top$ is full-rank, in which case $(\widehat{\mathbf{s}}_1, \ldots, \widehat{\mathbf{s}}_d)^\top \widehat{\mathbf{r}}$ is distributed uniformly over $\mathbb{Z}_p^d$ when picking $\widehat{\mathbf{r}} \leftarrow \mathbb{Z}_p^d$. □

Therefore, when we build our instantiation with parameter $d > 1$, we actually obtain an IBE with strictly stronger security guarantee which ensures the confidentiality of at most d ciphertexts for each identity. As a special case, if we set $d = 1$ (i.e., the SXDH assumption), the resulting IBE is still weak secure.

6.2 Computational Non-degeneracy and Full Adaptive Security

The attempt in the previous subsection more or less suggests that it is probably inevitable to introduce additional computational arguments in order to achieve fully adaptive security where a single $\widehat{h}^*$ can be paired with polynomially many $\widehat{g}_0$'s without violating the non-degeneracy property.

As a first step, we describe a computational version of non-degeneracy which is essentially similar to the s-BDDH assumption [21]. Our presentation follows the style of our revised ENDSG (in Section 3) in order to keep generality.

(Computational non-degeneracy (ND).) For any p.p.t. adversary $\mathcal{A}$, the following advantage function is negligible in k,

$$\mathsf{Adv}^{\mathrm{ND}}_{\mathcal{A}}(k, q, q', q'') := \left|\Pr[\mathcal{A}(D, T_0) = 1] - \Pr[\mathcal{A}(D, T_1) = 1]\right|,$$

where

$$D := \left(\mathrm{PP}, \left\{\widehat{h}^*_j \cdot \widetilde{h}^*_j\right\}_{j\in[q']}, \{\widehat{\mathbf{g}}_{j,j'}\}_{j\in[q],j'\in[q'']}\right),$$

$$T_0 := \left\{e(\widehat{g}_{0,j,j'}, \widehat{h}^{**}_j)\right\}_{j\in[q],j'\in[q'']},\ T_1 := \{R_{j,j'}\}_{j\in[q],j'\in[q'']}$$

and $\widehat{h}^*_j \leftarrow \widehat{\mathsf{SampH}}^*(\mathrm{PP}, \mathrm{SP})$, $\widetilde{h}^*_j \leftarrow \widetilde{\mathsf{SampH}}^*(\mathrm{PP}, \mathrm{SP})$, $\widehat{h}^{**}_j \leftarrow \widehat{\mathsf{SampH}}^*(\mathrm{PP}, \mathrm{SP})$, $\widehat{\mathbf{g}}_{j,j'} = (\widehat{g}_{0,j,j'}, \widehat{g}_{1,j,j'}, \ldots, \widehat{g}_{n,j,j'}) \leftarrow \widehat{\mathsf{SampG}}(\mathrm{PP}, \mathrm{SP})$ and $R_{j,j'} \leftarrow \mathbb{G}_T$.

It is not hard to see that an ENDSG with computational non-degeneracy property almost-tightly implies a fully adaptively secure IBE in MIMC setting, where we ensure the confidentiality of polynomial-many ciphertexts for each identity. The detailed proof can be found in the full version of the paper.

6.3 Computational Non-degeneracy from d-Linear Assumption

We now prove that the prime-order instantiation proposed in Section 4 has realized the computational non-degeneracy. And this immediately implies that the concrete IBE scheme shown in Section 5 is fully adaptively secure in MIMC setting with almost-tight reduction.

As before, we first rewrite all entries involved in the ND advantage function $\mathsf{Adv}^{\mathrm{ND}}_{\mathcal{A}}(k,q,q',q'')$ in terms of $\mathbf{B},\mathbf{B}^*,\mathbf{A}_i,\mathbf{R}$ as follows

$$\begin{aligned}
\mathrm{PP} &:= \begin{pmatrix} g_1^{\pi_{\mathrm{L}}(\mathbf{B})}, g_1^{\pi_{\mathrm{L}}(\mathbf{B}\mathbf{A}_1)}, \dots, g_1^{\pi_{\mathrm{L}}(\mathbf{B}\mathbf{A}_n)} \\ g_2^{\mathbf{B}^*\mathbf{R}}, \quad g_2^{\mathbf{B}^*\mathbf{A}_1^\top\mathbf{R}}, \dots, g_2^{\mathbf{B}^*\mathbf{A}_n^\top\mathbf{R}} \end{pmatrix}; \\
\widehat{h}_j^* \cdot \widetilde{h}_j^* &:= g_2^{\mathbf{B}^*\begin{pmatrix}\mathbf{0}_d \\ \widehat{\mathbf{r}}'_j \\ \widetilde{\mathbf{r}}'_j\end{pmatrix}}; \\
\widehat{\mathbf{g}}_{j,j'} &:= \left(g_1^{\pi_{\mathrm{M}}(\mathbf{B})\widehat{\mathbf{s}}_{j,j'}}, g_1^{\pi_{\mathrm{M}}(\mathbf{B}\mathbf{A}_1)\widehat{\mathbf{s}}_{j,j'}}, \dots, g_1^{\pi_{\mathrm{M}}(\mathbf{B}\mathbf{A}_n)\widehat{\mathbf{s}}_{j,j'}}\right); \\
e(\widehat{g}_{0,j,j'}, \widehat{h}_j^{**}) &:= e(g_1^{\pi_{\mathrm{M}}(\mathbf{B})\widehat{\mathbf{s}}_{j,j'}}, g_2^{\pi_{\mathrm{M}}(\mathbf{B}^*)\widehat{\mathbf{r}}_j}) = e(g_1,g_2)^{\widehat{\mathbf{s}}_{j,j'}^\top\widehat{\mathbf{r}}_j}; \\
R_{j,j'} &:= e(\widehat{g}_{0,j,j'}, \widehat{h}_j^{**}) \cdot e(g_1,g_2)^{\widehat{\gamma}_{j,j'}} = e(g_1,g_2)^{\widehat{\mathbf{s}}_{j,j'}^\top\widehat{\mathbf{r}}_j} \cdot e(g_1,g_2)^{\widehat{\gamma}_{j,j'}};
\end{aligned}$$

where $\widehat{\mathbf{r}}'_j, \widetilde{\mathbf{r}}'_j, \widehat{\mathbf{r}}_j, \widehat{\mathbf{s}}_{j,j'} \leftarrow \mathbb{Z}_p^d$ and $\widehat{\gamma}_{j,j'} \leftarrow \mathbb{Z}_p$. Then we prove the following lemma.

Lemma 7 ($(d,1,qq'')$-LLin $\Rightarrow$ ND). *For any p.p.t. adversary $\mathcal{A}$, there exists an adversary $\mathcal{B}$ such that*

$$\mathsf{Adv}^{\mathrm{ND}}_{\mathcal{A}}(k,q,q',q'') \leqslant \mathsf{Adv}^{(d,1,qq'')\text{-LLin}}_{\mathcal{B}}(k),$$

and $\mathsf{Time}(\mathcal{B}) \approx \mathsf{Time}(\mathcal{A}) + (qq''+q')d^2 \cdot \mathsf{poly}(k,n)$. *($\mathsf{poly}(k,n)$ is independent of $\mathcal{A}$)*

Overview of Proof. From the observation that all $\widehat{h}_j^{**} = g_2^{\pi_{\mathrm{M}}(\mathbf{B}^*)\widehat{\mathbf{r}}_j}$ are independently distributed and will never be given to $\mathcal{A}$ individually, we essentially prove a stronger result:

" Given D, $g_1^{\widehat{\mathbf{s}}_{j,j'}^\top\widehat{\mathbf{r}}_j}$ are computationally indistinguishable from $g_1^{\widehat{\mathbf{s}}_{j,j'}^\top\widehat{\mathbf{r}}_j + \widehat{\gamma}_{j,j'}}$. "

It is direct to based the pseudo-randomness of the challenge terms on the (d,q,q'')-LLin assumption. However the assumption is reduced to d-Lin assumption with reduction loss $\mathcal{O}(q)$. In order to obtain a tight reduction, we further rewrite the challenge term as

$$g_1^{\widehat{\mathbf{s}}_{j,j'}^\top\widehat{\mathbf{r}}_j} = g_1^{\widehat{\mathbf{s}}_{j,j'}^\top\mathbf{V}^\top\bar{\mathbf{r}}_j} = g_1^{\bar{\mathbf{r}}_j^\top\mathbf{V}\widehat{\mathbf{s}}_{j,j'}}$$

where $\mathbf{V}$ is a $(d+1)\times d$ matrix over $\mathbb{Z}_p$ of rank d and $\bar{\mathbf{r}}_j \leftarrow \mathbb{Z}_p^{d+1}$. Clearly, we implicitly define $\widehat{\mathbf{r}}_j := \mathbf{V}^\top\bar{\mathbf{r}}_j$. Since the matrix $\mathbf{V}$ is shared by all $\widehat{\mathbf{r}}_j$'s in challenge terms, we could now deal with polynomially many distinct $\widehat{\mathbf{r}}_j$'s *uniformly* which results in a proof with *constant* security loss.

Proof. Given an instance of $(d,1,qq'')$-LLin problem (i.e., set $\ell=1$ and $q=qq''$)

$$\left(\begin{array}{c} g_1, g_2, g_1^{a_1}, \dots, g_1^{a_d}, \left\{g_1^{b_i}\right\}_{i\in[d]}, \left\{g_1^{a_1 s_{1,j,j'}}, \dots, g_1^{a_d s_{d,j,j'}}\right\}_{j\in[q],j'\in[q'']}, \\ \left\{g_1^{b_1 s_{1,j,j'}+\cdots+b_d s_{d,j,j'}+s_{d+1,j,j'}}\right\}_{j\in[q],j'\in[q'']} \end{array}\right)$$

as input where either $s_{d+1,j,j'}=0$ or $s_{d+1,j,j'} \leftarrow \mathbb{Z}_p^*$, $\mathcal{B}$ works as follows:

Programming $\widehat{\mathbf{s}}_{j,j'}$ for $j \in [q], j' \in [q']$. Adversary $\mathcal{B}$ implicitly sets

$$\widehat{\mathbf{s}}_{j,j'} := (s_{1,j,j'}, \ldots, s_{d,j,j'})^\top.$$

Programming $\mathbf{B}, \mathbf{B}^*, \mathbf{A}_1, \ldots, \mathbf{A}_n, \mathbf{R}$. Define $\mathbf{W}$ as

$$\mathbf{W} := \left(\begin{array}{ccc|ccc|ccc} 1 & & & & & & & & \\ & \ddots & & & & & & & \\ & & 1 & & & & & & \\ \hline & & & a_1 & & & & & \\ & & & & \ddots & & & & \\ & & & & & a_d & & & \\ \hline & & & & & & 1 & & \\ & & & & & & & \ddots & \\ & & & & & & & & 1 \end{array}\right) \in \mathbb{Z}_p^{3d \times 3d}$$

and set $\mathbf{W}^* := (\mathbf{W}^{-1})^\top$. Sample $\bar{\mathbf{B}}, \bar{\mathbf{R}} \leftarrow \mathrm{GL}_{3d}(\mathbb{Z}_p)$ and set $\bar{\mathbf{B}}^* := (\bar{\mathbf{B}}^{-1})^\top$. Sample $\bar{\mathbf{A}}_1, \ldots, \bar{\mathbf{A}}_n \leftarrow \mathbb{Z}_p^{3d \times 3d}$, and implicitly set $\mathbf{B}$, $\mathbf{B}^*$, $\mathbf{R}$, and all $\mathbf{A}_i$ as Eq. (1). Of course, we also have the same relation as Eq. (2).

Simulating pp. Algorithm $\mathcal{B}$ can simulate

$$g_1^{\pi_{\mathrm{L}}(\mathbf{B})} = g_1^{\pi_{\mathrm{L}}(\bar{\mathbf{B}}\mathbf{W})} = g_1^{\bar{\mathbf{B}}\pi_{\mathrm{L}}(\mathbf{W})} \text{ and } g_1^{\pi_{\mathrm{L}}(\mathbf{B}\mathbf{A}_i)} = g_1^{\pi_{\mathrm{L}}(\bar{\mathbf{B}}\bar{\mathbf{A}}_i\mathbf{W})} = g_1^{\bar{\mathbf{B}}\bar{\mathbf{A}}_i\pi_{\mathrm{L}}(\mathbf{W})},$$
$$g_2^{\mathbf{B}^*\mathbf{R}} = g_2^{\bar{\mathbf{B}}^*\bar{\mathbf{R}}} \text{ and } g_2^{\mathbf{B}^*\mathbf{A}_1^\top\mathbf{R}} = g_2^{\bar{\mathbf{B}}^*\bar{\mathbf{A}}_i^\top\bar{\mathbf{R}}},$$

for $i \in [n]$ using the knowledge of $\pi_{\mathrm{L}}(\mathbf{W})$ and $\bar{\mathbf{B}}, \bar{\mathbf{B}}^*, \bar{\mathbf{A}}_1, \ldots, \bar{\mathbf{A}}_n, \bar{\mathbf{R}}$.

Simulating $\widehat{h}_j^* \cdot \widetilde{h}_j^*$ for $j \in [q']$. It is not hard to compute $\mathbf{W}^* \in \mathbb{Z}_p^{3d \times 3d}$ as

$$\mathbf{W}^* := \left(\begin{array}{ccc|ccc|ccc} 1 & & & & & & & & \\ & \ddots & & & & & & & \\ & & 1 & & & & & & \\ \hline & & & a_1^{-1} & & & & & \\ & & & & \ddots & & & & \\ & & & & & a_d^{-1} & & & \\ \hline & & & & & & 1 & & \\ & & & & & & & \ddots & \\ & & & & & & & & 1 \end{array}\right).$$

Observe that the right-bottom $2d \times 2d$ sub-matrix of $\mathbf{W}^*$ is full-rank with overwhelming probability, adversary $\mathcal{B}$ can simulate all $\widehat{h}_j^* \cdot \widetilde{h}_j^*$ as in the proof of Lemma 3 for the same reason.

Simulating $\widehat{\mathbf{g}}_{j,j'}$ for $j \in [q], j' \in [q']$. Algorithm $\mathcal{B}$ can simulate

$$g_1^{\mathbf{B}\begin{pmatrix} \mathbf{0}_d \\ \widehat{\mathbf{s}}_{j,j'} \\ \mathbf{0}_d \end{pmatrix}} = g_1^{\bar{\mathbf{B}}\mathbf{W}\begin{pmatrix} \mathbf{0}_d \\ \widehat{\mathbf{s}}_{j,j'} \\ \mathbf{0}_d \end{pmatrix}} \text{ and } g_1^{\mathbf{B}\mathbf{A}_i\begin{pmatrix} \mathbf{0}_d \\ \widehat{\mathbf{s}}_{j,j'} \\ \mathbf{0}_d \end{pmatrix}} = g_1^{\bar{\mathbf{B}}\bar{\mathbf{A}}_i\mathbf{W}\begin{pmatrix} \mathbf{0}_d \\ \widehat{\mathbf{s}}_{j,j'} \\ \mathbf{0}_d \end{pmatrix}}$$

for $i \in [n]$ using the knowledge of $\bar{\mathbf{B}}, \bar{\mathbf{A}}_1, \ldots, \bar{\mathbf{A}}_n$ and

$$g_1^{\mathbf{W}\begin{pmatrix}\mathbf{0}_d\\ \widehat{\mathbf{s}}_{j,j'}\\ \mathbf{0}_d\end{pmatrix}} = g_1^{\begin{pmatrix}\mathbf{0}_d\\ a_1 s_{1,j,j'}\\ \vdots\\ a_d s_{d,j,j'}\\ \mathbf{0}_d\end{pmatrix}}.$$

Simulating the challenge. Define matrix $\mathbf{V} \in \mathbb{Z}_p^{(d+1)\times d}$ of rank d as

$$\mathbf{V} := \begin{pmatrix} a_1 & & \\ & \ddots & \\ & & a_d \\ b_1 & \cdots & b_d \end{pmatrix}.$$

For all $j \in [q]$, algorithm $\mathcal{B}$ samples $\bar{\mathbf{r}}_j \leftarrow \mathbb{Z}_p^{d+1}$ and implicitly set $\widehat{\mathbf{r}}_j^\top := \bar{\mathbf{r}}_j^\top \mathbf{V}$. Algorithm $\mathcal{B}$ computes

$$g_1^{\widehat{\mathbf{r}}_j^\top \widehat{\mathbf{s}}_{j,j'} + \widehat{\gamma}_{j,j'}} = g_1^{\bar{\mathbf{r}}_j^\top \begin{pmatrix} a_1 s_{1,j,j'} \\ \vdots \\ a_d s_{d,j,j'} \\ b_1 s_{1,j,j'} + \cdots + b_d s_{d,j,j'} + s_{d+1,j,j'} \end{pmatrix}}$$

and outputs $e(g_1^{\widehat{\mathbf{r}}_j^\top \widehat{\mathbf{s}}_{j,j'} + \widehat{\gamma}_{j,j'}}, g_2)$ as challenges.

Analysis. Observe that, if $s_{d+1,j,j'} = 0$, the output challenge is distributed as

$$e(g_1^{\bar{\mathbf{r}}_j^\top (\mathbf{V}\widehat{\mathbf{s}}_{j,j'})}, g_2) = e(g_1, g_2)^{\widehat{\mathbf{s}}_{j,j'}^\top \widehat{\mathbf{r}}_j}$$

which is identical to T_0 where $\widehat{\gamma}_{j,j'} = 0$; if $s_{d+1,j,j'} \leftarrow \mathbb{Z}_p^*$, the output challenge is distributed as

$$e(g_1^{\bar{\mathbf{r}}_j^\top (\mathbf{V}\widehat{\mathbf{s}}_{j,j'} + \mathbf{e}_{d+1} s_{d+1,j,j'})}, g_2) = e(g_1, g_2)^{\widehat{\mathbf{s}}_{j,j'}^\top \widehat{\mathbf{r}}_j} \cdot \boxed{e(g_1, g_2)^{s_{d+1,j,j'} \mathbf{e}_{d+1}^\top \bar{\mathbf{r}}_j}}$$

which is identical to T_1 where $\widehat{\gamma}_{j,j'} := s_{d+1,j,j'} \mathbf{e}_{d+1}^\top \bar{\mathbf{r}}_j$ (in the box) is uniformly distributed over $\mathbb{Z}_p$. Therefore we may conclude that $\mathsf{Adv}^{\mathrm{ND}}_{\mathcal{A}}(k, q, q', q'') \leqslant \mathsf{Adv}^{(d,1,qq'')\text{-LLin}}_{\mathcal{B}}(k)$. □

Applying Lemma 1, we obtain the following corollary.

Corollary 2 (d-Lin ⇒ ND). *For any p.p.t. adversary $\mathcal{A}$, there exists an adversary $\mathcal{B}$ such that*

$$\mathsf{Adv}^{\mathrm{ND}}_{\mathcal{A}}(k, q, q', q'') \leqslant \mathsf{Adv}^{d\text{-Lin}}_{\mathcal{B}}(k) + 1/(p-1),$$

and $\mathsf{Time}(\mathcal{B}) \approx \mathsf{Time}(\mathcal{A}) + (qq''+q')d^2 \cdot \mathsf{poly}(k, n)$. *($\mathsf{poly}(k, n)$ is independent of $\mathcal{A}$)*

7 Towards More Efficient Solution: An Overview

7.1 Motivation and Technique

To obtain more efficient solutions, a promising idea is to reduce the dimension of two semi-functional spaces. Because we hope to continue to base our construction on the standard d-Lin assumption, we found the attempt gives rise to two technical problems due to the lack of dimensions.

- We can not prove *Left Subgroup Indistinguishability 2* (LS2) property using the technique provided by Chen and Wee in [16]. In particular, the simulator will need some elements in another source group (i.e., G_2) to simulate $\widehat{h}^* \cdot \widetilde{h}^*$ which is not given in the standard d-Lin assumption.
- We can not prove *Computational Non-degeneracy* (ND) property as before since neither $\widehat{g}_0$ nor $\widehat{h}^*$ has enough dimensions to program the d-Lin problem during the simulation.

The second issue is easy to solve by the observation that there are two semi-functional spaces and we only use one of them so far. We first define a variant of computational non-degeneracy property taking the $\sim$-semi-functional space into account. As long as two semi-functional spaces together has at least d dimensions, this computational non-degeneracy property should be proved as before. On the other hand, from the view of IBE, we could use the pseudo-randomness of $e(\widehat{g}_0 \cdot \widetilde{g}_0, \widehat{h}^* \cdot \widetilde{h}^*)$ to prove the security (decoupling challenge messages and ciphertexts) instead of just $e(\widehat{g}_0, \widehat{h}^*)$. To make the intuition explicit and general, we define three Left-subgroup indistinguishability (LS) requirements as: (1) LS1: $\mathbf{g} \approx \mathbf{g} \cdot \widehat{\mathbf{g}} \cdot \widetilde{\mathbf{g}}$; (2) LS2: $\mathbf{g} \cdot \widehat{\mathbf{g}} \cdot \widetilde{\mathbf{g}} \approx \mathbf{g} \cdot \widetilde{\mathbf{g}}$; (3) LS3: $\mathbf{g} \cdot \widehat{\mathbf{g}} \cdot \widetilde{\mathbf{g}} \approx \mathbf{g} \cdot \widehat{\mathbf{g}}$, where $\approx$ stands for "computationally indistinguishable".

In contrast, the first issue is seemingly hard to circumvent. Therefore, we decide to prove the LS2 property under an enhanced d-Lin assumption where we give adversary more elements on another source group G_2 for simulating $\widehat{h}^* \cdot \widetilde{h}^*$, which is called d-linear assumption with auxiliary input (d-LinAI) for *an even positive integer* d. Even though this assumption is non-standard in general, we point out that the concrete assumption with $d = 2$ is implied by the external decision linear assumption (XDLIN) [1] (see below), which has been formally introduced and used to build other cryptographic primitives.

We further fine-tune the ENDSG by hiding public parameters for SampH from the adversary when defining computational requirements, including LS1, LS2, LS3, NH, and ND. We argue that the absence of this part of public parameters will not arise difficulty in building IBE since they always correspond to the master secret key which is not necessary to be public according to the security model. Instead, we give the adversary enough samples from $\mathbb{H}^{n+1}$ which is sufficient for answering key extraction queries in the proof of "ENDSG implies IBE". We hope it will bring us a simple, clean and efficient solution.

In summary, we have fine-tuned the ENDSG in three aspects: (1) update non-degeneracy requirement; (2) re-define LS requirements; (3) hide parameters for SampH. Due to the lack of space, the fine-tuned ENDSG is given in the full

version of the paper and we also verify there that these modifications won't prevent ENDSG from almost-tightly deriving a fully secure IBE in MIMC setting.

The starting point of instantiating the fine-tuned ENDSG is the prime-order instantiation of dual system groups recently proposed by Chen *et al.* [13], which is quite simple due to a new basis randomizing technique. We technically work with $2d \times 2d$ matrix (for even positive integer d) and generate the basis using the dual pairing vector space method [29, 30, 26]. The first d-dimension subspace is normal space, the remaining two $d/2$-dimension subspaces act as $\wedge$-semi-functional subspace and $\sim$-semi-functional subspace, respectively. Note that the latter two are now smaller but enough for our proof (the entire semi-functional space has d dimensions). Finally, the basis is then randomized following [13]. Its security is tightly based on the d-LinAI assumption, which leads to an almost-tightly secure IBE in the MIMC setting with full security and higher efficiency. We describe, in the next subsection, the d-LinAI assumption and the resulting IBE scheme. More details could be found in the full version of the paper.

7.2 Concrete IBE from d-Linear Assumption with Auxiliary Input

Assume a prime-order bilinear group generator $\mathsf{GrpGen}(1^k)$ as defined in Section 4. The *d-linear assumption in G_1 with auxiliary input in G_2* (d-LinAI) is defined as follows, the analogous assumption in G_2 can be defined by exchanging the role of G_1 and G_2. We prove that the assumption holds in the generic model [34] in the full version of the paper. *Note that we always let d be an even positive integer.*

Assumption 3 (d-Linear Assumption in G_1 with Auxiliary Input). For any p.p.t. adversary $\mathcal{A}$, the following advantage function is negligible in k,

$$\mathsf{Adv}^{d\text{-LinAI}}_{\mathcal{A}}(k) := \left|\Pr[\mathcal{A}(D, \textsc{Aux}, T_0) = 1] - \Pr[\mathcal{A}(D, \textsc{Aux}, T_1) = 1]\right|,$$

where

$$D := \left(\mathcal{G}, g_1, g_2, g_1^{a_1}, \ldots, g_1^{a_d}, g_1^{a_{d+1}}, g_1^{a_1 s_1}, \ldots, g_1^{a_d s_d}\right)$$

$$\textsc{Aux} := \left(g_2^{a a_1^{-1} a_{d+1}}, \ldots, g_2^{a a_{d/2}^{-1} a_{d+1}}, g_2^{a}\right)$$

$$T_0 := g_1^{a_{d+1}(s_1+\cdots+s_d)},\ T_1 := g_1^{a_{d+1}(s_1+\cdots+s_d)+\boxed{s_{d+1}}}$$

and $\mathcal{G} \leftarrow \mathsf{GrpGen}(1^k)$, $a_1, \ldots, a_{d+1}, s_{d+1} \leftarrow \mathbb{Z}_p^*$, $a := a_1 \cdots a_{d/2}$, $s_1, \ldots, s_d \leftarrow \mathbb{Z}_p$.

Let $\pi_{\mathrm{L}}(\cdot)$ be the function mapping from a $2d \times 2d$ matrix to its left-most d columns. Given an bilinear group generator GrpGen such that d-LinAI assumption holds, the resulting IBE scheme built according to the main idea shown in the previous subsection is defined as follows.

- $\mathsf{Param}(1^k, n)$: Run $(p, G_1, G_2, G_T, e) \leftarrow \mathsf{GrpGen}(1^k)$. Sample $\mathbf{D} \leftarrow \mathrm{GL}_{2d}(\mathbb{Z}_p)$ and $\mathbf{W}_1, \ldots, \mathbf{W}_{2n} \leftarrow \mathbb{Z}_p^{2d \times 2d}$, and set $\mathbf{D}^* := (\mathbf{D}^{-1})^\top$. Output

$$\mathrm{GP} := \begin{pmatrix} g_1^{\pi_{\mathrm{L}}(\mathbf{D})}, & g_1^{\mathbf{W}_1^\top \pi_{\mathrm{L}}(\mathbf{D})}, & \ldots, & g_1^{\mathbf{W}_{2n}^\top \pi_{\mathrm{L}}(\mathbf{D})} \\ g_2^{\pi_{\mathrm{L}}(\mathbf{D}^*)}, & g_2^{\mathbf{W}_1 \pi_{\mathrm{L}}(\mathbf{D}^*)}, & \ldots, & g_2^{\mathbf{W}_{2n} \pi_{\mathrm{L}}(\mathbf{D}^*)} \end{pmatrix}.$$

– $\mathsf{Setup}(\mathrm{GP})$: Sample $\mathbf{k} \leftarrow \mathbb{Z}_p^{2d}$ and output

$$\mathrm{MPK} := \left(g_1^{\pi_{\mathrm{L}}(\mathbf{D})}, g_1^{\mathbf{W}_1^\top \pi_{\mathrm{L}}(\mathbf{D})}, \ldots, g_1^{\mathbf{W}_{2n}^\top \pi_{\mathrm{L}}(\mathbf{D})}; e(g_1, g_2)^{\pi_{\mathrm{L}}(\mathbf{D})^\top \mathbf{k}}\right) \in (G_1^{2d\times d})^{2n+1} \times G_T^d;$$
$$\mathrm{MSK} := \left(g_2^{\pi_{\mathrm{L}}(\mathbf{D}^*)}, g_2^{\mathbf{W}_1 \pi_{\mathrm{L}}(\mathbf{D}^*)}, \ldots, g_2^{\mathbf{W}_{2n} \pi_{\mathrm{L}}(\mathbf{D}^*)}; g_2^{\mathbf{k}}\right) \in (G_2^{2d\times d})^{2n+1} \times G_2^{2d}.$$

– $\mathsf{KeyGen}(\mathrm{MPK}, \mathrm{MSK}, \mathbf{y})$: Let $\mathbf{y} = (y_1, \ldots, y_n) \in \{0,1\}^n$. Sample $\mathbf{r} \leftarrow \mathbb{Z}_p^d$ and output

$$\mathrm{SK}_{\mathbf{y}} := \left(g_2^{\pi_{\mathrm{L}}(\mathbf{D}^*)\mathbf{r}},\ g_2^{\mathbf{k}+(\mathbf{W}_{2-y_1}+\cdots+\mathbf{W}_{2n-y_n})\pi_{\mathrm{L}}(\mathbf{D}^*)\mathbf{r}}\right) \in G_2^{2d} \times G_2^{2d}.$$

– $\mathsf{Enc}(\mathrm{MPK}, \mathbf{x}, \mathrm{M})$: Let $\mathbf{x} = (x_1, \ldots, x_n) \in \{0,1\}^n$ and $\mathrm{M} \in \mathbb{G}_T$. Sample $\mathbf{s} \leftarrow \mathbb{Z}_p^d$ and output

$$\mathrm{CT}_{\mathbf{x}} := \left(g_1^{\pi_{\mathrm{L}}(\mathbf{D})\mathbf{s}},\ g_1^{(\mathbf{W}_{2-x_1}+\cdots+\mathbf{W}_{2n-x_n})^\top \pi_{\mathrm{L}}(\mathbf{D})\mathbf{s}},\ e(g_1, g_2)^{\mathbf{s}^\top \pi_{\mathrm{L}}(\mathbf{D})^\top \mathbf{k}} \cdot \mathrm{M}\right)$$
$$\in G_1^{2d} \times G_1^{2d} \times G_T.$$

– $\mathsf{Dec}(\mathrm{MPK}, \mathrm{SK}, \mathrm{CT})$. Let $\mathrm{SK} = (K_0, K_1)$ and $\mathrm{CT} = (C_0, C_1, C_2)$. Output

$$\mathrm{M} := C_2 \cdot e(C_1, K_0)/e(C_0, K_1).$$

One may argue that the d-LinAI assumption is not standard and complex. We show that, by setting $d = 2$, we derive the DLIN assumption with auxiliary input $\mathrm{AUX} := (g_2^{a_3}, g_2^{a_1})$. It is easy to verify that this special instantiation is implied by the *External Decision Linear Assumption* [1]. Motivated by this observation, we remark that we may build the above IBE system using *symmetric* bilinear pairings and base the security on the well-known and *standard* Decisional Linear Assumption, where $G_1 = G_2$ and AUX in G_2 is automatically revealed.

Acknowledgement. We want to thank Hoeteck Wee for helpful discussions and all the anonymous reviewers for their helpful comments on earlier drafts of this paper. This work is supported by the National Natural Science Foundation of China (Grant Nos. 61472142, 61411146001, 61321064, 61371083, 61373154, 61172085, 61170080, U1135004), 973 Program (No. 2014CB360501), Science and Technology Commission of Shanghai Municipality (Grant Nos. 14YF1404200, 13JC1403500), the Specialized Research Fund for the Doctoral Program of Higher Education of China through the Prioritized Development Projects under Grant 20130073130004.

References

1. Masayuki Abe, Melissa Chase, Bernardo David, Markulf Kohlweiss, Ryo Nishimaki, and Miyako Ohkubo. Constant-size structure-preserving signatures: Generic constructions and simple assumptions. In *Advances in Cryptology–ASIACRYPT 2012*, pages 4–24. Springer, 2012.

2. Shashank Agrawal and Melissa Chase. A study of pair encodings: Predicate encryption in prime order groups. *TCC 2016*, 2016.
3. Shweta Agrawal, Dan Boneh, and Xavier Boyen. Efficient lattice (H)IBE in the standard model. In *Advances in Cryptology - EUROCRYPT 2010*, pages 553–572, 2010.
4. Shweta Agrawal, Dan Boneh, and Xavier Boyen. Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical IBE. In *Advances in Cryptology - CRYPTO 2010*, pages 98–115, 2010.
5. Nuttapong Attrapadung. Dual system encryption via doubly selective security: Framework, fully secure functional encryption for regular languages, and more. In *Advances in Cryptology - EUROCRYPT 2014*, pages 557–577, 2014.
6. Nuttapong Attrapadung, Goichiro Hanaoka, and Shota Yamada. A framework for identity-based encryption with almost tight security. In *Advances in Cryptology–ASIACRYPT 2015*, pages 521–549. Springer, 2015.
7. Nuttapong Attrapadung and Shota Yamada. Duality in abe: Converting attribute based encryption for dual predicate and dual policy via computational encodings. In *Topics in Cryptology—CT-RSA 2015*, pages 87–105. Springer, 2015.
8. Mihir Bellare, Brent Waters, and Scott Yilek. Identity-based encryption secure against selective opening attack. In *TCC 2011*, pages 235–252, 2011.
9. Olivier Blazy, Eike Kiltz, and Jiaxin Pan. (hierarchical) identity-based encryption from affine message authentication. In *Advances in Cryptology–CRYPTO 2014*, pages 408–425. Springer, 2014.
10. Dan Boneh and Xavier Boyen. Efficient selective-id secure identity-based encryption without random oracles. In *Advances in Cryptology - EUROCRYPT 2004*, pages 223–238, 2004.
11. Dan Boneh and Xavier Boyen. Secure identity based encryption without random oracles. In *Advances in Cryptology - CRYPTO 2004*, pages 443–459, 2004.
12. Dan Boneh and Matthew K. Franklin. Identity-based encryption from the weil pairing. In *Advances in Cryptology - CRYPTO 2001*, pages 213–229, 2001.
13. Jie Chen, Romain Gay, and Hoeteck Wee. Improved dual system abe in prime-order groups via predicate encodings. In *Advances in Cryptology-EUROCRYPT 2015*, pages 595–624. Springer, 2015.
14. Jie Chen, Hoon Wei Lim, San Ling, Huaxiong Wang, and Hoeteck Wee. Shorter IBE and signatures via asymmetric pairings. In *Pairing-Based Cryptography - Pairing 2012*, pages 122–140, 2012.
15. Jie Chen and Hoeteck Wee. Fully, (almost) tightly secure IBE and dual system groups. In *Advances in Cryptology - CRYPTO 2013 - Part II*, pages 435–460, 2013.
16. Jie Chen and Hoeteck Wee. Dual system groups and its applications - compact HIBE and more. *IACR Cryptology ePrint Archive*, 2014:265, 2014.
17. Clifford Cocks. An identity based encryption scheme based on quadratic residues. In *Cryptography and Coding, 8th IMA International Conference, Cirencester, UK, December 17-19, 2001, Proceedings*, pages 360–363, 2001.
18. Craig Gentry. Practical identity-based encryption without random oracles. In *Advances in Cryptology - EUROCRYPT 2006*, pages 445–464, 2006.
19. Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing*, pages 197–206, 2008.
20. Junqing Gong, Zhenfu Cao, Shaohua Tang, and Jie Chen. Extended dual system group and shorter unbounded hierarchical identity based encryption. *Designs, Codes and Cryptography*, 2015. DOI 10.1007/s10623-015-0117-z.

21. Dennis Hofheinz, Jessica Koch, and Christoph Striecks. Identity-based encryption with (almost) tight security in the multi-instance, multi-ciphertext setting. In *Public-Key Cryptography - PKC 2015*, 2015.
22. Charanjit S. Jutla and Arnab Roy. Shorter quasi-adaptive NIZK proofs for linear subspaces. In *Advances in Cryptology - ASIACRYPT 2013 -Part I*, pages 1–20, 2013.
23. Charanjit S Jutla and Arnab Roy. Switching lemma for bilinear tests and constant-size nizk proofs for linear subspaces. In *Advances in Cryptology–CRYPTO 2014*, pages 295–312. Springer, 2014.
24. Allison Lewko and Brent Waters. Unbounded hibe and attribute-based encryption. In *Advances in Cryptology–EUROCRYPT 2011*, pages 547–567. Springer, 2011.
25. Allison B. Lewko. Tools for simulating features of composite order bilinear groups in the prime order setting. In *Advances in Cryptology - EUROCRYPT 2012*, pages 318–335, 2012.
26. Allison B. Lewko, Tatsuaki Okamoto, Amit Sahai, Katsuyuki Takashima, and Brent Waters. Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In *Advances in Cryptology - EUROCRYPT 2010*, pages 62–91, 2010.
27. Allison B. Lewko and Brent Waters. New proof methods for attribute-based encryption: Achieving full security through selective techniques. In *Advances in Cryptology - CRYPTO 2012*, pages 180–198, 2012.
28. Moni Naor and Omer Reingold. Number-theoretic constructions of efficient pseudo-random functions. *J. ACM*, 51(2):231–262, 2004.
29. Tatsuaki Okamoto and Katsuyuki Takashima. Homomorphic encryption and signatures from vector decomposition. In *Pairing-Based Cryptography–Pairing 2008*, pages 57–74. Springer, 2008.
30. Tatsuaki Okamoto and Katsuyuki Takashima. Hierarchical predicate encryption for inner-products. In *Advances in Cryptology–ASIACRYPT 2009*, pages 214–231. Springer, 2009.
31. Tatsuaki Okamoto and Katsuyuki Takashima. Fully secure unbounded inner-product and attribute-based encryption. In *Advances in Cryptology - ASIACRYPT 2012*, pages 349–366, 2012.
32. Somindu C. Ramanna, Sanjit Chatterjee, and Palash Sarkar. Variants of waters' dual system primitives using asymmetric pairings - (extended abstract). In *Public Key Cryptography - PKC 2012*, pages 298–315, 2012.
33. Adi Shamir. Identity-based cryptosystems and signature schemes. In *Advances in Cryptology, Proceedings of CRYPTO '84*, pages 47–53, 1984.
34. Victor Shoup. Lower bounds for discrete logarithms and related problems. In *Advances in Cryptology–EUROCRYPT 97*, pages 256–266. Springer, 1997.
35. Brent Waters. Efficient identity-based encryption without random oracles. In *Advances in Cryptology - EUROCRYPT 2005*, pages 114–127, 2005.
36. Brent Waters. Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In *Advances in Cryptology - CRYPTO 2009*, pages 619–636, 2009.
37. Hoeteck Wee. Dual system encryption via predicate encodings. In *Theory of Cryptography 2014*, pages 616–637, 2014.