Des. Codes Cryptogr. (2015) 77:725–771
DOI 10.1007/s10623-015-0131-1

Achieving short ciphertexts or short secret-keys for adaptively secure general inner-product encryption

Tatsuaki Okamoto[1] · **Katsuyuki Takashima**[2]

Received: 14 October 2014 / Revised: 5 August 2015 / Accepted: 10 August 2015 /
Published online: 7 September 2015

Abstract In this paper, we present two *non-zero inner-product* encryption (NIPE) schemes that are *adaptively secure* under a standard assumption, the decisional linear (DLIN) assumption, in the standard model. One of the proposed NIPE schemes features *constant-size ciphertexts* and the other features *constant-size secret-keys*. Our NIPE schemes imply an identity-based revocation (IBR) system with constant-size ciphertexts or constant-size secret-keys that is adaptively secure under the DLIN assumption. Any previous IBR scheme with constant-size ciphertexts or constant-size secret-keys was *not adaptively secure* in the standard model. This paper also presents two zero inner-product encryption (ZIPE) schemes each of which has constant-size ciphertexts or constant-size secret-keys and is adaptively secure under the DLIN assumption in the standard model. They imply an identity-based broadcast encryption system with constant-size ciphertexts or constant-size secret-keys that is adaptively secure under the DLIN assumption. We also extend the proposed ZIPE schemes in two directions, one is a *fully-attribute-hiding* ZIPE scheme with *constant-size secret-keys*, and the other a *hierarchical* ZIPE scheme with *constant-size ciphertexts*.

This is one of several papers published in *Designs, Codes and Cryptography* comprising the “Special Issue on Cryptography, Codes, Designs and Finite Fields: In Memory of Scott A. Vanstone”.

An extended abstract of a preliminary version [26] of this paper was presented in CANS 2011, the 10th International Conference on Cryptology and Network Security. This is the full version of the extended abstract [26] and provides significant technical contributions over [26], e.g., fully-attribute-hiding ZIPE scheme with constant-size secret-keys, a hierarchical ZIPE scheme with constant-size ciphertexts, and proofs of all lemmas for security. Refer to Sects. 10 and 12, and Appendix.

✉ Tatsuaki Okamoto
okamoto.tatsuaki@lab.ntt.co.jp

Katsuyuki Takashima
Takashima.Katsuyuki@aj.MitsubishiElectric.co.jp

[1] NTT, Musashino, Tokyo, Japan

[2] Mitsubishi Electric, Kamakura, Kanagawa, Japan

Keywords Inner-product encryption · Non-zero inner-product encryption · Functional encryption

Mathematics Subject Classification Primary 94A60 , 11T71 , 14G50 , 68P25

1 Introduction

1.1 Background

Functional encryption (FE) is an advanced concept of encryption or a generalization of public-key encryption (PKE) and identity-based encryption (IBE). In FE systems, a receiver can decrypt a ciphertext using a secret-key corresponding to a parameter v if and only if v is suitably related to another parameter x specified for the ciphertext, or $R(v, x) = 1$ for some relation R (i.e., relation R holds for (v, x)). More generally, a secret key in FE is associated with a function f and a ciphertext of plaintext x is decrypted to $f(x)$ by the secret key [9,28].

The first flavor of functional encryption traces back to the work of Sahai and Waters [29], which was subsequently extended in [2,3,6,10,13,14,17,18,20,25,32]. In their concept called attribute-based encryption (ABE), for example, parameter v for a secret-key is an access control policy, and parameter x for a ciphertext is a set of attributes. Decryption requires attribute set x to satisfy policy v, i.e., relation $R^{\mathsf{ABE}}(v, x) = 1$ iff x satisfies v. Identity-based broadcast encryption (IBBE) [1,8,12,16,30] and revocation (IBR) [21] schemes can also be thought of as functional encryption systems where a ciphertext is encrypted for a set of identities $S = \{ID_1, \ldots, ID_n\}$ in IBBE (resp. IBR) systems, and to decrypt it by a secret-key associated with ID requires that $ID \in S$ (resp. $ID \notin S$), i.e., relation $R^{\mathsf{IBBE}}(ID, S) = 1$ (resp. $R^{\mathsf{IBR}}(ID, S) = 1$) iff $ID \in S$ (resp. $ID \notin S$).

Katz et al. [19] introduced a functional encryption scheme for zero inner products, zero inner product encryption (ZIPE) where a ciphertext encrypted with vector $\vec{x}$ can be decrypted by any key associated with vector $\vec{v}$ such that $\vec{v} \cdot \vec{x} = 0$, i.e., relation $R^{\mathsf{ZIPE}}(\vec{v}, \vec{x}) = 1$ iff $\vec{v} \cdot \vec{x} = 0$. Their scheme is *selectively secure* in the standard model and the ciphertext size is *linear* in the dimension of vectors, n, although it achieves an additional security property, *attribute-hiding*, in which $\vec{x}$ is hidden from the ciphertext. As shown in [19], ZIPE provides functional encryption for a wide class of relations corresponding to equalities, polynomials and CNF/DNF formulae.

Attrapadung and Libert [4] proposed a ZIPE scheme as well as a non-zero IPE (NIPE) scheme, where NIPE relation $R^{\mathsf{NIPE}}(\vec{v}, \vec{x}) = 1$ iff $\vec{v} \cdot \vec{x} \neq 0$. NIPE supports a wide class of relations corresponding to the complement of those for ZIPE. In their ZIPE and NIPE schemes, without retaining the attribute-hiding property, the ciphertext size reduces to a *constant* in n (the dimension of vectors, $\vec{v}$ and $\vec{x}$), as long as the description of the vector is not considered a part of the ciphertext, which is a common assumption in the broadcast encryption/revocation applications. Hereafter in this paper, "constant" will be used in this sense. In addition, the number of pairing operations for decryption in [4] is constant. Their ZIPE system is *adaptively secure* in the standard model, but the NIPE scheme is *not adaptively secure* (co-selectively secure) in the standard model.

The ZIPE system [4] implies an *adaptively secure* identity-based broadcast encryption (IBBE) scheme with constant-size ciphertexts in the standard model, while previous IBBE schemes with constant-size ciphertexts were either only selective-ID secure [1,8,12] or secure in a non-standard model [16,30]. Among IBBE systems with short ciphertexts (including selective-ID secure ones), the IBBE scheme [4] is the only one relying on standard assumptions, namely the DBDH and DLIN assumptions. The NIPE scheme [4] implies a co-selectively secure (not adaptively secure) identity-based revocation (IBR) system [21] with constant-size ciphertexts in the standard model. Lewko et al. [21] presented IBR systems with constant-size public and secret keys that are not adaptively secure. Hence, the following problems are still remained.

1. No NIPE scheme with constant-size ciphertexts is *adaptively secure* in the standard model, and no IBR scheme with constant-size ciphertexts or constant-size secret-keys is *adaptively secure* in the standard model. No NIPE scheme with constant-size *secret-keys* has been presented.
2. No ZIPE (or no IBBE) scheme with constant-size ciphertexts is adaptively (or selectively) secure under a *single* standard assumption in the standard model. No ZIPE scheme with constant-size *secret-keys* has been presented.

1.2 Our result

We address the problems. Note that all of our results are obtained in the standard model.

1. This paper presents the first *adaptively secure* NIPE scheme that has constant-size ciphertexts or constant-size secret-keys (Sects. 6 and 7). The security assumption is a standard one, the decisional linear (DLIN) assumption. This implies the first *adaptively secure* IBR scheme with constant-size ciphertexts or constant-size secret-keys.
2. This paper also presents the first ZIPE scheme that has constant-size ciphertexts or constant-size secret-keys and is adaptively secure solely under a *single* standard assumption, the DLIN assumption (Sects. 8 and 9). This implies the first IBBE scheme with constant-size ciphertexts that is adaptively secure solely under a *single* standard assumption.
3. We present two extensions of the proposed ZIPE schemes. One is a *fully-attribute-hiding* ZIPE scheme with *constant-size secret-keys* (Sect. 10). It is obtained by applying the technique of the fully-attribute-hiding ZIPE scheme in [27] to the proposed ZIPE scheme with constant-size secret-keys in Sect. 9, while the ZIPE scheme in Sect. 9 is *weakly-attribute-hiding*. The other extension is a *hierarchical* ZIPE scheme with *constant-size ciphertexts* (Sect. 12). These schemes are adaptively secure under the DLIN assumption.

The number of pairing operations for decryption is constant in all the proposed schemes. We summarize a comparison of our results with those of [4] in Table 1 in Sect. 11 (see the items of ‘Security’, ‘Assump.’, ‘CT Size’ and ‘SK Size’ in Table 1, for the features discussed in Sects. 1.1 and 1.2).

1.3 Related works

Adaptively secure and *attribute-hiding* ZIPE scheme under the DLIN assumption has been presented [25], but the ciphertext-size is linear in n (*not constant*), while our ZIPE scheme has *constant-size* ciphertexts and is adaptively secure but *not attribute-hiding*.

After the publication of the preliminary version [26] of this paper, Chen–Wee [11] constructed a constant-size ciphertext and adaptively secure spatial encryption scheme, which includes ZIPE as a special case. Although both of our ZIPE scheme and Chen–Wee’s scheme have constant-size ciphertexts, the concrete size of a ciphertext in their scheme is shorter than ours.

1.4 Key techniques

All of the proposed schemes in this paper are constructed on dual system encryption [22,31] and dual pairing vector spaces (DPVS) [20,24,25]. See Sect. 1.5 for some notations in this section. In DPVS, a pair of dual (or orthonormal) bases, $\mathbb{B}$ and $\mathbb{B}^*$, are randomly generated using a *fully* random linear transformation $X \overset{\mathsf{U}}{\leftarrow} GL(N, \mathbb{F}_q)$ (N: dimension of $\mathsf{span}\langle\mathbb{B}\rangle$ and $\mathsf{span}\langle\mathbb{B}^*\rangle$) such that $\mathbb{B}$ and $\mathbb{B}^*$ are transformed from canonical basis $\mathbb{A}$ by X and $(X^{-1})^{\mathrm{T}}$, respectively (see Sect. 2 and [20,24,25]). In a typical application of DPVS to cryptography, a portion of $\mathbb{B}$ (say $\hat{\mathbb{B}}$) is used as a public key and the corresponding portion of $\mathbb{B}^*$ (say $\hat{\mathbb{B}}^*$) is used as a secret key or trapdoor.

In this paper, we develop a novel technique on DPVS, where we employ a *special form* of random linear transformation $X \in GL(N, \mathbb{F}_q)$, or $X \in \mathcal{L}(4, n, \mathbb{F}_q)$ of Eq. (3) in Sect. 6.2, in place of *fully* random linear transformation $X \overset{\mathsf{U}}{\leftarrow} GL(N, \mathbb{F}_q)$. This form of X provides us a framework to achieve short ciphertexts or short secret-keys as well as a small number of pairing operations in decryption. It, however, is a challenging task to find such a special form of X like Eq. (3) that meet the several requirements for the dual system encryption method to prove the adaptive security of ZIPE and NIPE schemes under the DLIN assumption. Such requirements are given hereafter. To reduce the security of our schemes, especially Problems 1 and 2 in this paper, to the DLIN assumption, the form of X should be *consistent* with the distribution of the DLIN problem. The form of X should be *sparse* enough to achieve short ciphertexts or secret-keys. We should also have a *special* pairwise independence lemma, Lemma 6 in Sect. 6.4, that is due to the special form of X, where linear random transformations U and Z are more restricted (or specific) than those of previous results, e.g., [25], with fully random X. See Sect. 6.1 for more details.

1.5 Notations

When A is a random variable or distribution, $y \overset{\mathsf{R}}{\leftarrow} A$ denotes that y is randomly selected from A according to its distribution. When A is a set, $y \overset{\mathsf{U}}{\leftarrow} A$ denotes that y is uniformly selected from A. A vector symbol denotes a vector representation over $\mathbb{F}_q$, e.g., $\vec{x}$ denotes $(x_1, \ldots, x_n) \in \mathbb{F}_q^n$. For two vectors $\vec{x} = (x_1, \ldots, x_n)$ and $\vec{v} = (v_1, \ldots, v_n)$, $\vec{x} \cdot \vec{v}$ denotes the inner-product $\sum_{i=1}^n x_i v_i$. The vector $\vec{0}$ is used to denote the zero vector in $\mathbb{F}_q^n$ for any n. X^{T} denotes the transpose of matrix X. I_ℓ denotes the $\ell \times \ell$ identity matrix. A boldface letter denotes an element of vector space $\mathbb{V}$, e.g., $\boldsymbol{x} \in \mathbb{V}$. When $\boldsymbol{b}_i \in \mathbb{V}$ $(i = 1, \ldots, \ell)$, $\mathsf{span}\langle \boldsymbol{b}_1, \ldots, \boldsymbol{b}_\ell\rangle \subseteq \mathbb{V}$ (resp. $\mathsf{span}\langle \vec{x}_1, \ldots, \vec{x}_\ell\rangle$) denotes the subspace generated by $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_\ell$ (resp. $\vec{x}_1, \ldots, \vec{x}_\ell$). For bases $\mathbb{B} := (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_N)$ and $\mathbb{B}^* := (\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_N^*)$, $(x_1, \ldots, x_N)_{\mathbb{B}} := \sum_{i=1}^N x_i \boldsymbol{b}_i$ and $(y_1, \ldots, y_N)_{\mathbb{B}^*} := \sum_{i=1}^N y_i \boldsymbol{b}_i^*$. An n-dimensional vector $\vec{e}_j$ denotes the canonical basis vector $(\overbrace{0 \cdots 0}^{j-1}, 1, \overbrace{0 \cdots 0}^{n-j}) \in \mathbb{F}_q^n$ for $j = 1, \ldots, n$. $GL(n, \mathbb{F}_q)$ denotes the general linear group of degree n over $\mathbb{F}_q$. For a linear subspace $V \subset \mathbb{F}_q^n$, $V^\perp$ denotes the orthogonal complement, i.e., $V^\perp := \{\vec{w} \in \mathbb{F}_q^n | \vec{w} \cdot \vec{v} = 0 \text{ for all } \vec{v} \in V\}$.

2 Dual pairing vector spaces by direct product of symmetric pairing groups

In this paper, for simplicity of description, we will present the proposed schemes on the symmetric version of dual pairing vector spaces (DPVS) [23,24] constructed using symmetric bilinear pairing groups given in Definition 1. Owing to the abstraction of DPVS, the presentation and the security proof of the proposed schemes are essentially the same as those on the asymmetric version of DPVS, $(q, \mathbb{V}, \mathbb{V}^*, \mathbb{G}_T, \mathbb{A}, \mathbb{A}^*, e)$, for which see Appendix "Proofs of Lemmas 4–12 in Sect. 6" in the full version of [25]. The symmetric version is a specific (self-dual) case of the asymmetric version, where $\mathbb{V} = \mathbb{V}^*$ and $\mathbb{A} = \mathbb{A}^*$.

Definition 1 (*Symmetric bilinear pairing groups*) $(q, \mathbb{G}, \mathbb{G}_T, G, e)$ are a tuple of a prime q, cyclic additive group $\mathbb{G}$ and multiplicative group $\mathbb{G}_T$ of order q, $G \neq 0 \in \mathbb{G}$, and a polynomial-time computable nondegenerate bilinear pairing $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ i.e., $e(sG, tG) = e(G, G)^{st}$ and $e(G, G) \neq 1$.

Let $\mathcal{G}_{\mathsf{bpg}}$ be an algorithm that takes input 1^λ and outputs a description of bilinear pairing groups $(q, \mathbb{G}, \mathbb{G}_T, G, e)$ with security parameter λ.

Definition 2 (*Dual pairing vector spaces* (*DPVS*)) $(q, \mathbb{V}, \mathbb{G}_T, \mathbb{A}, e)$ by a direct product of symmetric pairing groups $(q, \mathbb{G}, \mathbb{G}_T, G, e)$ are a tuple of prime q, N-dimensional vector space $\mathbb{V} := \overbrace{\mathbb{G} \times \cdots \times \mathbb{G}}^{N}$ over $\mathbb{F}_q$, cyclic group $\mathbb{G}_T$ of order q, canonical basis $\mathbb{A} := (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N)$ of $\mathbb{V}$, where $\boldsymbol{a}_i := (\overbrace{0, \ldots, 0}^{i-1}, G, \overbrace{0, \ldots, 0}^{N-i})$, and pairing $e : \mathbb{V} \times \mathbb{V} \to \mathbb{G}_T$. The pairing is defined by $e(\boldsymbol{x}, \boldsymbol{y}) := \prod_{i=1}^{N} e(G_i, H_i) \in \mathbb{G}_T$ where $\boldsymbol{x} := (G_1, \ldots, G_N) \in \mathbb{V}$ and $\boldsymbol{y} := (H_1, \ldots, H_N) \in \mathbb{V}$. This is nondegenerate bilinear i.e., $e(s\boldsymbol{x}, t\boldsymbol{y}) = e(\boldsymbol{x}, \boldsymbol{y})^{st}$ and if $e(\boldsymbol{x}, \boldsymbol{y}) = 1$ for all $\boldsymbol{y} \in \mathbb{V}$, then $\boldsymbol{x} = \boldsymbol{0}$. For all i and j, $e(\boldsymbol{a}_i, \boldsymbol{a}_j) = e(G, G)^{\delta_{i,j}}$ where $\delta_{i,j} = 1$ if $i = j$, and 0 otherwise, and $e(G, G) \neq 1 \in \mathbb{G}_T$.

DPVS also has linear transformations $\phi_{i,j}$ on $\mathbb{V}$ s.t. $\phi_{i,j}(\boldsymbol{a}_j) = \boldsymbol{a}_i$ and $\phi_{i,j}(\boldsymbol{a}_k) = \boldsymbol{0}$ if $k \neq j$, which can be easily achieved by $\phi_{i,j}(\boldsymbol{x}) := (\overbrace{0, \ldots, 0}^{i-1}, G_j, \overbrace{0, \ldots, 0}^{N-i})$ where $\boldsymbol{x} := (G_1, \ldots, G_N)$. We call $\phi_{i,j}$ "canonical maps".

DPVS generation algorithm $\mathcal{G}_{\mathsf{dpvs}}$ takes input 1^λ ($\lambda \in \mathbb{N}$) and $N \in \mathbb{N}$, and outputs a description of $\mathsf{param}_{\mathbb{V}} := (q, \mathbb{V}, \mathbb{G}_T, \mathbb{A}, e)$ with security parameter λ and N-dimensional $\mathbb{V}$. It can be constructed by using $\mathcal{G}_{\mathsf{bpg}}$.

3 Definitions of zero and non-zero inner-product encryption (ZIPE/NIPE)

This section defines zero and non-zero inner-product encryption (ZIPE/NIPE) and their security. The relations R^{ZIPE} of ZIPE and R^{NIPE} of NIPE are defined over vectors $\vec{x} \in \mathbb{F}_q^n \setminus \{\vec{0}\}$ and $\vec{v} \in \mathbb{F}_q^n \setminus \{\vec{0}\}$, where $R^{\mathsf{ZIPE}}(\vec{v}, \vec{x}) := 1$ iff $\vec{x} \cdot \vec{v} = 0$, and $R^{\mathsf{NIPE}}(\vec{v}, \vec{x}) := 1$ iff $\vec{x} \cdot \vec{v} \neq 0$, respectively.

Definition 3 (*Zero and non-zero inner-product encryption: ZIPE/NIPE*) Let a relation R be R^{ZIPE} or R^{NIPE}. A zero (resp. non-zero) inner-product encryption scheme consists of four algorithms with $R := R^{\mathsf{ZIPE}}$ (resp. $R := R^{\mathsf{NIPE}}$).

Setup This is a randomized algorithm that takes as input security parameter. It outputs public parameters pk and master secret key sk.

KeyGen This is a randomized algorithm that takes as input vector $\vec{v}$, pk and sk. It outputs a decryption key $\mathsf{sk}_{\vec{v}}$.

Enc This is a randomized algorithm that takes as input message m, a vector, $\vec{x}$, and public parameters pk. It outputs a ciphertext $\mathsf{ct}_{\vec{x}}$.

Dec This takes as input ciphertext $\mathsf{ct}_{\vec{x}}$ that was encrypted under a vector $\vec{x}$, decryption key $\mathsf{sk}_{\vec{v}}$ for vector $\vec{v}$, and public parameters pk. It outputs either plaintext m or the distinguished symbol $\perp$.

A ZIPE (or NIPE) scheme should have the following correctness property: for all $(\mathsf{pk}, \mathsf{sk}) \overset{\mathsf{R}}{\leftarrow} \mathsf{Setup}(1^\lambda)$, all vectors $\vec{v}$, all decryption keys $\mathsf{sk}_{\vec{v}} \overset{\mathsf{R}}{\leftarrow} \mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \vec{v})$, all messages m, all vectors $\vec{x}$, all ciphertexts $\mathsf{ct}_{\vec{x}} \overset{\mathsf{R}}{\leftarrow} \mathsf{Enc}(\mathsf{pk}, m, \vec{x})$, it holds that $m = \mathsf{Dec}(\mathsf{pk}, \mathsf{sk}_{\vec{v}}, \mathsf{ct}_{\vec{x}})$ with overwhelming probability, if $R(\vec{v}, \vec{x}) = 1$.

We define three security notions in Definitions 4–6.

Definition 4 (*Adaptively payload-hiding security*) The model for proving the adaptively payload-hiding security of ZIPE (or NIPE) under chosen plaintext attacks is given hereafter.

Setup The challenger runs the setup algorithm, $(\mathsf{pk}, \mathsf{sk}) \overset{\mathsf{R}}{\leftarrow} \mathsf{Setup}(1^\lambda)$, and gives public parameters pk to the adversary.

Phase 1 The adversary is allowed to adaptively issue a polynomial number of queries, $\vec{v}$, to the challenger or oracle $\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \cdot)$ for private keys, $\mathsf{sk}_{\vec{v}}$, associated with $\vec{v}$.

Challenge The adversary submits two messages, $m^{(0)}$ and $m^{(1)}$, and a vector, $\vec{x}$, provided that no $\vec{v}$ queried to the challenger in Phase 1 satisfies $R(\vec{v}, \vec{x}) = 1$. The challenger flips a coin $b \overset{\mathsf{U}}{\leftarrow} \{0, 1\}$, and computes $\mathsf{ct}_{\vec{x}}^{(b)} \overset{\mathsf{R}}{\leftarrow} \mathsf{Enc}(\mathsf{pk}, m^{(b)}, \vec{x})$. It gives $\mathsf{ct}_{\vec{x}}^{(b)}$ to the adversary.

Phase 2 The adversary is allowed to adaptively issue a polynomial number of queries, $\vec{v}$, to the challenger or oracle $\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \cdot)$ for private keys, $\mathsf{sk}_{\vec{v}}$, associated with $\vec{v}$, provided that $R(\vec{v}, \vec{x}) \neq 1$.

Guess The adversary outputs a guess b' of b.

The advantage of adversary $\mathcal{A}$ in the above game, $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{ZIPE,PH}}(\lambda)$ (or $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{NIPE,PH}}(\lambda)$), is defined by $\Pr[b' = b] - 1/2$ for any security parameter λ. A ZIPE (or NIPE) scheme is adaptively payload-hiding secure if all polynomial time adversaries have at most a negligible advantage in the game.

Remark 1 We have two remarks on variants of the above security notion.

- In a weaker security notion, *selectively* payload-hiding, the adversary is required to declare the challenge vector $\vec{x}$ at the beginning of the game (before Setup). Similarly, the weaker (selective) security variants can be defined in place of the two (adaptive) security notions in Definitions 5 and 6.
- The above security notion, which is secure against chosen-plaintext attacks (CPA), can be easily extended to the security notion against chosen-ciphertext attacks (CCA) by allowing an adversary to give decryption queries in Phases 1 and 2. Since there is a standard (efficient) methodology to transform a CPA-secure FE (including NIPE/ZIPE) scheme to a CCA-secure FE scheme by using the Canetti–Halevi–Katz (CHK) transformation or the Boneh–Katz (BK) transformation [7] as is given in [25], we only present CPA-secure NIPE/ZIPE schemes in this paper.

Definition 5 (*Adaptively weakly-attribute-hiding security*) The model for proving the adaptively weakly-attribute-hiding security of ZIPE under chosen plaintext attacks is obtained from the above game by replacing **Challenge** and **Phase 2** steps by the following:

Challenge The adversary submits two messages, $(m^{(0)}, m^{(1)})$, and two vectors, $(\vec{x}^{(0)}, \vec{x}^{(1)})$, provided that no $\vec{v}$ queried to the challenger in Phase 1 satisfies $R(\vec{v}, \vec{x}^{(0)}) = 1$ or $R(\vec{v}, \vec{x}^{(1)}) = 1$. The challenger flips a coin $b \overset{\mathsf{U}}{\leftarrow} \{0, 1\}$, and computes $\mathsf{ct}_{\vec{x}^{(b)}} \overset{\mathsf{R}}{\leftarrow} \mathsf{Enc}(\mathsf{pk}, m^{(b)}, \vec{x}^{(b)})$. It gives $\mathsf{ct}_{\vec{x}^{(b)}}$ to the adversary.

Phase 2 The adversary is allowed to adaptively issue a polynomial number of queries, $\vec{v}$, to the challenger or oracle $\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \cdot)$ for private keys, $\mathsf{sk}_{\vec{v}}$, associated with $\vec{v}$, provided that $R(\vec{v}, \vec{x}^{(0)}) \neq 1$ and $R(\vec{v}, \vec{x}^{(1)}) \neq 1$.

The advantage of adversary $\mathcal{A}$ in the above game, $\mathsf{Adv}^{\mathsf{ZIPE,wAH}}_{\mathcal{A}}(\lambda)$, is defined by $\Pr[b' = b] - 1/2$ for any security parameter λ. A ZIPE scheme is adaptively weakly-attribute-hiding secure if all polynomial time adversaries have at most a negligible advantage in the game.

Informally, in adaptively fully-attribute-hiding security game, adversary is allowed to issue both types of key queries, $R(\vec{v}, \vec{x}^{(b)}) = 0$ and $R(\vec{v}, \vec{x}^{(b)}) = 1$, in a single security game. It gives a strong security than Definition 5 and is given in the following Definition 6.

Definition 6 (*Adaptively fully-attribute-hiding security*) The model for proving the adaptively fully-attribute-hiding security of ZIPE under chosen plaintext attacks is obtained from the above game by replacing **Challenge** and **Phase 2** steps by the following:

Challenge The adversary submits challenge attribute vector $(\vec{x}^{(0)}, \vec{x}^{(1)})$ and challenge plaintexts $(m^{(0)}, m^{(1)})$, subject to the following restrictions:

- $\vec{v} \cdot \vec{x}^{(0)} \neq 0$ and $\vec{v} \cdot \vec{x}^{(1)} \neq 0$ for all the key queried predicate vectors, $\vec{v}$.
- Two challenge plaintexts are equal, i.e., $m^{(0)} = m^{(1)}$, and any key query $\vec{v}$ satisfies $R(\vec{v}, \vec{x}^{(0)}) = R(\vec{v}, \vec{x}^{(1)})$, i.e., one of the following conditions.
  - $\vec{v} \cdot \vec{x}^{(0)} = 0$ and $\vec{v} \cdot \vec{x}^{(1)} = 0$,
  - $\vec{v} \cdot \vec{x}^{(0)} \neq 0$ and $\vec{v} \cdot \vec{x}^{(1)} \neq 0$,

The challenger flips a coin $b \overset{\mathsf{U}}{\leftarrow} \{0, 1\}$, and computes $\mathsf{ct}_{\vec{x}^{(b)}} \overset{\mathsf{R}}{\leftarrow} \mathsf{Enc}(\mathsf{pk}, m^{(b)}, \vec{x}^{(b)})$. It gives $\mathsf{ct}_{\vec{x}^{(b)}}$ to the adversary.

Phase 2 The adversary is allowed to adaptively issue a polynomial number of queries, $\vec{v}$, to the challenger or oracle $\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \cdot)$ for private keys, $\mathsf{sk}_{\vec{v}}$, associated with $\vec{v}$, subject to the restriction given in the challenge step.

The advantage of adversary $\mathcal{A}$ in the above game is defined as $\mathsf{Adv}^{\mathsf{ZIPE,AH}}_{\mathcal{A}}(\lambda) := \mathsf{Pr}[\mathcal{A}$ wins $] - 1/2$ for any security parameter λ. An IPE scheme is adaptively fully-attribute-hiding (AH) (against chosen plaintext attacks) if all probabilistic polynomial-time adversaries $\mathcal{A}$ have at most negligible advantage in the above game.

For each run of the game, the variable s is defined as $s := 0$ if $m^{(0)} \neq m^{(1)}$ for challenge plaintexts $m^{(0)}$ and $m^{(1)}$, and $s := 1$ otherwise.

4 Decisional linear (DLIN) assumption

Definition 7 The DLIN problem is to guess $\beta \in \{0, 1\}$, given $(\mathsf{param}_{\mathbb{G}}, G, \xi G, \kappa G, \delta\xi G, \sigma\kappa G, Y_\beta) \overset{\mathsf{R}}{\leftarrow} \mathcal{G}^{\mathsf{DLIN}}_{\beta}(1^\lambda)$, where $\mathcal{G}^{\mathsf{DLIN}}_{\beta}(1^\lambda) : \mathsf{param}_{\mathbb{G}} := (q, \mathbb{G}, \mathbb{G}_T, G, e) \overset{\mathsf{R}}{\leftarrow} \mathcal{G}_{\mathsf{bpg}}(1^\lambda)$,

$\kappa, \delta, \xi, \sigma \xleftarrow{\mathsf{U}} \mathbb{F}_q,\ Y_0 := (\delta+\sigma)G,\ Y_1 \xleftarrow{\mathsf{U}} \mathbb{G}$, return $(\mathsf{param}_{\mathbb{G}}, G, \xi G, \kappa G, \delta\xi G, \sigma\kappa G, Y_\beta)$, for $\beta \xleftarrow{\mathsf{U}} \{0, 1\}$. For a probabilistic machine $\mathcal{E}$, we define the advantage of $\mathcal{E}$ for the DLIN problem as: $\mathsf{Adv}_{\mathcal{E}}^{\mathsf{DLIN}}(\lambda) := \left| \mathsf{Pr}\left[\mathcal{E}(1^\lambda, \varrho) \to 1 \middle| \varrho \xleftarrow{\mathsf{R}} \mathcal{G}_0^{\mathsf{DLIN}}(1^\lambda) \right] - \mathsf{Pr}\left[\mathcal{E}(1^\lambda, \varrho) \to 1 \middle| \varrho \xleftarrow{\mathsf{R}} \mathcal{G}_1^{\mathsf{DLIN}}(1^\lambda) \right] \right|$. The DLIN assumption is: For any probabilistic polynomial-time adversary $\mathcal{E}$, the advantage $\mathsf{Adv}_{\mathcal{E}}^{\mathsf{DLIN}}(\lambda)$ is negligible in λ.

5 Special matrix subgroups

Lemmas 1–3 are key lemmas for the security proof for our (H)IPE schemes. For a positive integer n, let

$$\mathcal{H}(n, \mathbb{F}_q) := \left\{ \begin{pmatrix} u & & & u'_1 \\ & \ddots & & \vdots \\ & & u & u'_{n-1} \\ & & & u'_n \end{pmatrix} \middle| \begin{array}{l} u, u'_l \in \mathbb{F}_q \text{ for } l = 1, \ldots, n, \\ \text{a blank element in the matrix} \\ \text{denotes } 0 \in \mathbb{F}_q \end{array} \right\}, \tag{1}$$

$$\widetilde{\mathcal{H}}(n, \mathbb{F}_q) := \left\{ \begin{pmatrix} u'_1 & & & \\ u'_2 & u & & \\ \vdots & & \ddots & \\ u'_n & & & u \end{pmatrix} \middle| \begin{array}{l} u, u'_l \in \mathbb{F}_q \text{ for } l = 1, \ldots, n, \\ \text{a blank element in the matrix} \\ \text{denotes } 0 \in \mathbb{F}_q \end{array} \right\}. \tag{2}$$

Lemma 1 *$\mathcal{H}(n, \mathbb{F}_q) \cap GL(n, \mathbb{F}_q)$ and $\widetilde{\mathcal{H}}(n, \mathbb{F}_q) \cap GL(n, \mathbb{F}_q)$ are subgroups of $GL(n, \mathbb{F}_q)$.*

Lemma 1 is directly verified from the definition of groups. □

For positive integers w and n, let

$$\mathcal{L}(w, n, \mathbb{F}_q) := \left\{ X := \begin{pmatrix} X_{1,1} & \cdots & X_{1,w} \\ \vdots & & \vdots \\ X_{w,1} & \cdots & X_{w,w} \end{pmatrix} \middle| X_{i,j} := \begin{pmatrix} \mu_{i,j} & & & \mu'_{i,j,1} \\ & \ddots & & \vdots \\ & & \mu_{i,j} & \mu'_{i,j,n-1} \\ & & & \mu'_{i,j,n} \end{pmatrix} \begin{array}{c} \in \mathcal{H}(n, \mathbb{F}_q) \\ \text{for } i, j = \\ 1, \ldots, w \end{array} \right\} \bigcap GL(wn, \mathbb{F}_q), \tag{3}$$

$$\widetilde{\mathcal{L}}(w, n, \mathbb{F}_q) := \left\{ X := \begin{pmatrix} X_{1,1} & \cdots & X_{1,w} \\ \vdots & & \vdots \\ X_{w,1} & \cdots & X_{w,w} \end{pmatrix} \middle| X_{i,j} := \begin{pmatrix} \mu'_{i,j,1} & & & \\ \mu'_{i,j,2} & \mu_{i,j} & & \\ \vdots & & \ddots & \\ \mu'_{i,j,n} & & & \mu_{i,j} \end{pmatrix} \begin{array}{c} \in \widetilde{\mathcal{H}}(n, \mathbb{F}_q) \\ \text{for } i, j = \\ 1, \ldots, w \end{array} \right\} \bigcap GL(wn, \mathbb{F}_q). \tag{4}$$

Lemma 2 *$\mathcal{L}(w, n, \mathbb{F}_q)$ and $\widetilde{\mathcal{L}}(w, n, \mathbb{F}_q)$ are subgroups of $GL(wn, \mathbb{F}_q)$.*

$$\mathcal{L}^{+}(w,n,\mathbb{F}_q) := \left\{ X := \begin{pmatrix} \chi_{0,0} & \chi_{0,1}\vec{e}_n & \cdots & \chi_{0,w}\vec{e}_n \\ \vec{\chi}^{\mathrm{T}}_{1,0} & X_{1,1} & \cdots & X_{1,w} \\ \vdots & \vdots & & \vdots \\ \vec{\chi}^{\mathrm{T}}_{w,0} & X_{w,1} & \cdots & X_{w,w} \end{pmatrix} \middle| \begin{array}{l} X_{i,j} \in \mathcal{H}(n,\mathbb{F}_q), \\ \vec{\chi}_{i,0} := (\chi_{i,0,l})_{l=1,\ldots,n} \in \mathbb{F}_q^n, \\ \chi_{0,0},\, \chi_{0,j} \in \mathbb{F}_q \\ \text{for } i, j = 1,\ldots,w \end{array} \right\}$$
$$\bigcap GL(wn+1,\mathbb{F}_q). \tag{5}$$

Lemma 3 *$\mathcal{L}^{+}(w,n,\mathbb{F}_q)$ is a subgroup of $GL(wn+1,\mathbb{F}_q)$.*

Proofs of Lemmas 2 and 3 are given in Appendix "Proofs of Lemmas 2 and 3 in Sect. 5".

6 NIPE scheme with constant-size ciphertexts

6.1 Key ideas in constructing the proposed NIPE scheme

In this section, we will explain the key ideas of constructing and proving the security of the proposed NIPE scheme.

First, we will show how short ciphertexts and efficient decryption can be achieved in our scheme. Here, we will use a simplified (or toy) version of the proposed NIPE scheme, for which the security is no more ensured in the standard model under the DLIN assumption.

A ciphertext in the simplified NIPE scheme consists of two vector elements, $(\boldsymbol{c}_0, \boldsymbol{c}_1) \in \mathbb{G}^5 \times \mathbb{G}^n$, and $c_3 \in \mathbb{G}_T$. A secret-key consists of two vector elements, $(\boldsymbol{k}^*_0, \boldsymbol{k}^*_1) \in \mathbb{G}^5 \times \mathbb{G}^n$. Therefore, to achieve constant-size ciphertexts, we have to compress $\boldsymbol{c}_1 \in \mathbb{G}^n$ to a constant size in n. We now employ a special form of basis generation matrix, $X := \begin{pmatrix} \mu & & & \mu'_1 \\ & \ddots & & \vdots \\ & & \mu & \mu'_{n-1} \\ & & & \mu'_n \end{pmatrix} \in \mathcal{H}(n,\mathbb{F}_q)$ of Eq. (1) in Sect. 6.2, where $\mu, \mu'_1, \ldots, \mu'_n \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q$ and a blank in the matrix denotes $0 \in \mathbb{F}_q$. The system parameter or DPVS public basis is $\mathbb{B} := \begin{pmatrix} \boldsymbol{b}_1 \\ \vdots \\ \boldsymbol{b}_n \end{pmatrix} := \begin{pmatrix} \mu G & & & \mu'_1 G \\ & \ddots & & \vdots \\ & & \mu G & \mu'_{n-1} G \\ & & & \mu'_n G \end{pmatrix}$. Let a ciphertext associated with $\vec{x} := (x_1,\ldots,x_n)$ be $\boldsymbol{c}_1 := (\omega\vec{x})_{\mathbb{B}} = \omega(x_1\boldsymbol{b}_1 + \cdots + x_n\boldsymbol{b}_n) = (x_1\omega\mu G, \ldots, x_{n-1}\omega\mu G, \omega(\sum_{i=1}^n x_i\mu'_i)G)$, where $\omega \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q$. Then, $\boldsymbol{c}_1$ can be compressed to only *two* group elements $(C_1 := \omega\mu G,\ C_2 := \omega(\sum_{i=1}^n x_i\mu'_i)G)$ as well as $\vec{x}$, since $\boldsymbol{c}_1$ can be obtained by $(x_1C_1, \ldots, x_{n-1}C_1, C_2)$ (note that $x_iC_1 = x_i\omega\mu G$ for $i = 1,\ldots,n-1$). That is, a ciphertext (excluding $\vec{x}$) can be just two group elements, or the size is constant in n.

Let $\mathbb{B}^* := (\boldsymbol{b}^*_i)$ be the dual orthonormal basis of $\mathbb{B} := (\boldsymbol{b}_i)$, and $\mathbb{B}^*$ be the master secret key in the simplified NIPE scheme. We specify $(\boldsymbol{c}_0, \boldsymbol{k}^*_0, c_3)$ such that $e(\boldsymbol{c}_0, \boldsymbol{k}^*_0) = g_T^{\zeta} \cdot g_T^{\omega\delta}$ and $c_3 := g_T^{\zeta} m \in \mathbb{G}_T$. We also set a secret-key for $\vec{v}$ as $\boldsymbol{k}^*_1 := (\delta\vec{v})_{\mathbb{B}^*} = \delta(v_1\boldsymbol{b}^*_1 + \cdots + v_n\boldsymbol{b}^*_n)$. From the dual orthonormality of $\mathbb{B}$ and $\mathbb{B}^*$, it then holds that $e(\boldsymbol{c}_1, \boldsymbol{k}^*_1) = g_T^{\omega\delta(\vec{x}\cdot\vec{v})}$. Hence, a decryptor can compute $g_T^{\omega\delta}$ if and only if $\vec{x}\cdot\vec{v} \neq 0$, i.e., can obtain plaintext m by $c_3 \cdot e(\boldsymbol{c}_0, \boldsymbol{k}^*_0)^{-1} \cdot e(\boldsymbol{c}_1, \boldsymbol{k}^*_1)^{(\vec{x}\cdot\vec{v})^{-1}}$. Since $\boldsymbol{c}_1$ is expressed as $(x_1C_1, \ldots, x_{n-1}C_1, C_2) \in \mathbb{G}^n$ and $\boldsymbol{k}^*_1$ is parsed as a n-tuple $(K_1, \ldots, K_n) \in \mathbb{G}^n$, the value of $e(\boldsymbol{c}_1, \boldsymbol{k}^*_1)$ is $\prod_{i=1}^{n-1} e(x_iC_1, K_i) \cdot$

$e(C_2, K_n) = \prod_{i=1}^{n-1} e(C_1, x_i K_i) \cdot e(C_2, K_n) = e(C_1, \sum_{i=1}^{n-1} x_i K_i) \cdot e(C_2, K_n)$. That is, $n-1$ scalar multiplications in $\mathbb{G}$ and *two* pairing operations are enough for computing $e(\boldsymbol{c}_1, \boldsymbol{k}_1^*)$. Therefore, only a small (constant) number of pairing operations are required for decryption.

We then explain how our *full* NIPE scheme is constructed on the above-mentioned simplified NIPE scheme. The target of designing the full NIPE scheme is to achieve adaptive security under the DLIN assumption. Here, we adopt a strategy similar to that of [25], in which the dual system encryption methodology is employed in a modular or hierarchical manner. That is, two top level assumptions, the security of Problems 1 and 2, are directly used in the dual system encryption methodology and these assumptions are reduced to a primitive assumption, the DLIN assumption.

To meet the requirements for applying to the dual system encryption methodology and reducing to the DLIN assumption, the underlying vector space as well as the basis generator matrix X is four times larger than that of the above-mentioned simplified scheme. For example, $\boldsymbol{k}_1^* :=(\ \delta\vec{v}, \ 0^n, \ \vec{\varphi}_1, \ 0^n \ )_{\mathbb{B}^*}$, $\boldsymbol{c}_1 = (\omega\vec{x}, 0^n, 0^n, \eta_1\vec{x})_{\mathbb{B}}$, and $X := \begin{pmatrix} X_{1,1} & \cdots & X_{1,4} \\ \vdots & & \vdots \\ X_{4,1} & \cdots & X_{4,4} \end{pmatrix} \in \mathcal{L}(4, n, \mathbb{F}_q)$ of Eq. (3) in Sect. 6.2, where each $X_{i,j}$ is of the form of $X \in \mathcal{H}(n, \mathbb{F}_q)$ in the simplified scheme. The vector space consists of four orthogonal subspaces, i.e., real encoding part, hidden part, secret-key randomness part, and ciphertext randomness part. The simplified NIPE scheme corresponds to the first real encoding part.

A key fact in the security reduction is that $\mathcal{L}(4, n, \mathbb{F}_q)$ is a *subgroup* of $GL(4n, \mathbb{F}_q)$ (Lemma 2), which enables a *random-self-reducibility* argument for reducing the DLIN problem to Problems 1 and 2 in this paper. The property that $\mathcal{H}(n, \mathbb{F}_q) \cap GL(n, \mathbb{F}_q)$ is a *subgroup* of $GL(n, \mathbb{F}_q)$ is also crucial for a special form of pairwise independence lemma in this paper (Lemma 6), where $\mathcal{H}(n, \mathbb{F}_q)$ is specified in $\mathcal{L}(4, n, \mathbb{F}_q)$ or X. Our Problem 2, which is based on this lemma, employs special form matrices $U \xleftarrow{\mathsf{U}} \mathcal{H}(n, \mathbb{F}_q) \cap GL(n, \mathbb{F}_q)$ and $Z :=(U^{-1})^{\mathrm{T}}$. Informally, our pairwise independence lemma implies that, for all $(\vec{x}, \vec{v})$, a pair, $(\vec{x}U, \vec{v}Z)$, is uniformly distributed over $(\mathsf{span}\langle\vec{x}, \vec{e}_n\rangle \setminus \mathsf{span}\langle\vec{e}_n\rangle) \times (\mathbb{F}_q^n \setminus \mathsf{span}\langle\vec{e}_n\rangle^{\perp})$ with preserving the inner-product value, $\vec{x} \cdot \vec{v}$, i.e., $(\vec{x}U, \vec{v}Z)$ reveal no information but $\vec{x}$ and $\vec{x} \cdot \vec{v}$.

A difference of matrix X with the ZIPE scheme will be noted in Remark 10.

6.2 Dual orthonormal basis generator

We describe random dual orthonormal basis generator $\mathcal{G}_{\mathsf{ob}}^{\mathsf{NIPE,CT}}$ below, which is used as a subroutine in the proposed NIPE scheme.

$$
\begin{aligned}
&\mathcal{G}_{\mathsf{ob}}^{\mathsf{NIPE,CT}}(1^\lambda, 4, n): \quad \mathsf{param}_{\mathbb{G}} :=(q, \mathbb{G}, \mathbb{G}_T, G, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{bpg}}(1^\lambda), \quad N_0 :=5, \ N_1 :=4n, \\
&\quad \mathsf{param}_{\mathbb{V}_t} :=(q, \mathbb{V}_t, \mathbb{G}_T, \mathbb{A}_t, e) :=\mathcal{G}_{\mathsf{dpvs}}(1^\lambda, N_t, \mathsf{param}_{\mathbb{G}}) \ \text{ for } t = 0, 1, \\
&\quad \psi \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times, \ g_T :=e(G, G)^\psi, \ \mathsf{param}_n :=(\{\mathsf{param}_{\mathbb{V}_t}\}_{t=0,1}, \ g_T), \\
&\quad X_0 :=(\chi_{0,i,j})_{i,j=1,\ldots,5} \xleftarrow{\mathsf{U}} GL(N_0, \mathbb{F}_q), \ X_1 \xleftarrow{\mathsf{U}} \mathcal{L}(4, n, \mathbb{F}_q), \ \text{hereafter}, \\
&\quad \{\mu_{i,j}, \mu'_{i,j,l}\}_{i,j=1,\ldots4;l=1,\ldots,n} \ \text{denotesnon-zeroentriesof} X_1 \ \text{as in Eq. (3)}, \\
&\quad \boldsymbol{b}_{0,i} :=(\chi_{0,i,1}, \ldots, \chi_{0,i,5})_{\mathbb{A}} = \textstyle\sum_{j=1}^5 \chi_{0,i,j}\boldsymbol{a}_j \ \text{ for } i = 1, \ldots, 5, \ \mathbb{B}_0 :=(\boldsymbol{b}_{0,1}, \ldots, \boldsymbol{b}_{0,5}), \\
&\quad B_{i,j} :=\mu_{i,j}G, \ B'_{i,j,l} :=\mu'_{i,j,l}G \ \text{ for } i, j = 1, \ldots, 4; l = 1, \ldots, n, \\
&\quad \text{for } t = 0, 1, \ (\vartheta_{t,i,j})_{i,j=1,\ldots,N_t} :=\psi \cdot (X_t^{\mathrm{T}})^{-1},
\end{aligned}
$$

$\boldsymbol{b}^*_{t,i} := (\vartheta_{t,i,1}, \dots, \vartheta_{t,i,N_t})_{\mathbb{A}} = \sum_{j=1}^{N_t} \vartheta_{t,i,j}\boldsymbol{a}_j$ for $i = 1, \dots, N_t$, $\mathbb{B}^*_t := (\boldsymbol{b}^*_{t,1}, \dots, \boldsymbol{b}^*_{t,N_t})$,
return $(\mathsf{param}_n, \mathbb{B}_0, \mathbb{B}^*_0, \{B_{i,j}, B'_{i,j,l}\}_{i,j=1,\dots,4;l=1,\dots,n}, \mathbb{B}^*_1)$.

Remark 2 Let

$$\left.\begin{array}{l} \begin{pmatrix} \boldsymbol{b}_{1,(i-1)n+1} \\ \vdots \\ \boldsymbol{b}_{1,in} \end{pmatrix} := \begin{pmatrix} B_{i,1} & & & B'_{i,1,1} & & B_{i,4} & & & B'_{i,4,1} \\ & \ddots & & \vdots & \cdots & & \ddots & & \vdots \\ & & B_{i,1} & B'_{i,1,n-1} & & & & B_{i,4} & B'_{i,4,n-1} \\ & & & B'_{i,1,n} & & & & & B'_{i,4,n} \end{pmatrix} \\ \qquad \text{for } i = 1, \dots, 4, \\ \mathbb{B}_1 := (\boldsymbol{b}_{1,1}, \dots, \boldsymbol{b}_{1,4n}), \end{array}\right\} \tag{6}$$

where a blank element in the matrix denotes $0 \in \mathbb{G}$. $\mathbb{B}_1$ is the dual orthonormal basis of $\mathbb{B}^*_1$, i.e., $e(\boldsymbol{b}_{1,i}, \boldsymbol{b}^*_{1,i}) = g_T$ and $e(\boldsymbol{b}_{1,i}, \boldsymbol{b}^*_{1,j}) = 1$ for $1 \le i \ne j \le 4n$.

6.3 Construction

In the description of the scheme, we assume that input vector, $\vec{x} := (x_1, \dots, x_n)$, has an index l $(1 \le l \le n-1)$ with $x_l \ne 0$, and that input vector, $\vec{v} := (v_1, \dots, v_n)$, satisfies $v_n \ne 0$. The plaintext space is $\mathbb{G}_T$.

$\mathsf{Setup}(1^\lambda,\ n) : (\mathsf{param}_n, \mathbb{B}_0, \mathbb{B}^*_0, \{B_{i,j}, B'_{i,j,l}\}_{i,j=1,\dots,4;l=1,\dots,n}, \mathbb{B}^*_1) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}^{\mathsf{NIPE,CT}}(1^\lambda, 4, n)$,
$\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5})$, $\widehat{\mathbb{B}}^*_0 := (\boldsymbol{b}^*_{0,1}, \boldsymbol{b}^*_{0,3}, \boldsymbol{b}^*_{0,4})$, $\widehat{\mathbb{B}}^*_1 := (\boldsymbol{b}^*_{1,1}, \dots, \boldsymbol{b}^*_{1,n}, \boldsymbol{b}^*_{1,2n+1}, \dots, \boldsymbol{b}^*_{1,3n})$,
return $\mathsf{pk} := (1^\lambda, \mathsf{param}_n, \widehat{\mathbb{B}}_0, \{B_{i,j}, B'_{i,j,l}\}_{i=1,4;\,j=1,\dots,4;l=1,\dots,n})$, $\mathsf{sk} := \{\widehat{\mathbb{B}}^*_t\}_{t=0,1}$.

$\mathsf{KeyGen}(\mathsf{pk},\ \mathsf{sk},\ \vec{v}) :\quad \delta, \varphi_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q,\ \vec{\varphi}_1 \xleftarrow{\mathsf{U}} \mathbb{F}_q^n,\quad \boldsymbol{k}^*_0 := (\delta,\ 0,\ 1,\ \varphi_0,\ 0)_{\mathbb{B}^*_0}$,

$\boldsymbol{k}^*_1 := (\overbrace{\delta\vec{v}}^{n},\ \overbrace{0^n}^{n},\ \overbrace{\vec{\varphi}_1}^{n},\ \overbrace{0^n}^{n})_{\mathbb{B}^*_1}$, return $\mathsf{sk}_{\vec{v}} := (\vec{v}, \boldsymbol{k}^*_0, \boldsymbol{k}^*_1)$.

$\mathsf{Enc}(\mathsf{pk},\ m,\ \vec{x}) :\quad \omega, \eta_0, \eta_1, \zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q,\quad \boldsymbol{c}_0 := (-\omega,\ 0,\ \zeta,\ 0,\ \eta_0)_{\mathbb{B}_0},\ \ c_3 := g_T^{\zeta} m$,
$C_{1,j} := \omega B_{1,j} + \eta_1 B_{4,j},\ \ C_{2,j} := \sum_{l=1}^{n} x_l(\omega B'_{1,j,l} + \eta_1 B'_{4,j,l})$ for $j = 1, \dots, 4$,
return $\mathsf{ct}_{\vec{x}} := (\vec{x}, \boldsymbol{c}_0, \{C_{1,j}, C_{2,j}\}_{j=1,\dots,4}, c_3)$.

$\mathsf{Dec}(\mathsf{pk},\ \mathsf{sk}_{\vec{v}} := (\vec{v}, \boldsymbol{k}^*_0, \boldsymbol{k}^*_1),\ \mathsf{ct}_{\vec{x}} := (\vec{x}, \boldsymbol{c}_0, \{C_{1,j}, C_{2,j}\}_{j=1,\dots,4}, c_3)) :$
Parse $\boldsymbol{k}^*_1$ as a $4n$-tuple $(K^*_1, \dots, K^*_{4n}) \in \mathbb{G}^{4n}$,
$D^*_j := \sum_{l=1}^{n-1} ((\vec{x} \cdot \vec{v})^{-1} x_l)\, K^*_{(j-1)n+l}$ for $j = 1, \dots, 4$,
$F := e(\boldsymbol{c}_0, \boldsymbol{k}^*_0) \cdot \prod_{j=1}^{4} \Big(e(C_{1,j}, D^*_j) \cdot e(C_{2,j}, K^*_{jn}) \Big)$, return $m' := c_3/F$.

Remark 3 A part of output of $\mathsf{Setup}(1^\lambda, n)$, $\{B_{i,j}, B'_{i,j,l}\}_{i=1,4;\,j=1,\dots,4;l=1,\dots,n}$, can be identified with $\widehat{\mathbb{B}}_1 := (\boldsymbol{b}_{1,1}, \dots, \boldsymbol{b}_{1,n}, \boldsymbol{b}_{1,3n+1}, \dots, \boldsymbol{b}_{1,4n})$ through the form of Eq. (6), while $\mathbb{B}_1 := (\boldsymbol{b}_{1,1}, \dots, \boldsymbol{b}_{1,4n})$ is identified with $\{B_{i,j}, B'_{i,j,l}\}_{i,j=1,\dots,4;\,l=1,\dots,n}$ by Eq. (6). Decryption Dec can be alternatively described as:

$$\mathsf{Dec}'(\mathsf{pk},\ \mathsf{sk}_{\vec{v}} :=(\vec{v}, \boldsymbol{k}_0^*, \boldsymbol{k}_1^*),\ \mathsf{ct}_{\vec{x}} :=(\vec{x}, \boldsymbol{c}_0, \{C_{1,j}, C_{2,j}\}_{j=1,\dots,4}, c_3)) :$$

$$\boldsymbol{c}_1 := (\ \overbrace{x_1 C_{1,1}, \dots, x_{n-1} C_{1,1}, C_{2,1}}^{n},\ \ \dots,\ \overbrace{x_1 C_{1,4}, \dots, x_{n-1} C_{1,4}, C_{2,4}}^{n}\ ),$$

$$\text{that is, } \boldsymbol{c}_1 = (\ \overbrace{\omega\vec{x},}^{n}\ \overbrace{0^n,}^{n}\ \overbrace{0^n,}^{n}\ \overbrace{\eta_1\vec{x}}^{n}\ )_{\mathbb{B}_1},\ F :=e(\boldsymbol{c}_0, \boldsymbol{k}_0^*) \cdot e(\boldsymbol{c}_1, (\vec{x}\cdot\vec{v})^{-1}\boldsymbol{k}_1^*),$$

$$\text{return } m' :=c_3/F.$$

[Correctness] Using the alternate decryption Dec', $F = e(\boldsymbol{c}_0, \boldsymbol{k}_0^*) \cdot e(\boldsymbol{c}_1, (\vec{x}\cdot\vec{v})^{-1}\boldsymbol{k}_1^*) = g_T^{-\omega\delta+\zeta} g_T^{\omega\delta(\vec{x}\cdot\vec{v})/(\vec{x}\cdot\vec{v})} = g_T^{\zeta}$ if $\vec{x}\cdot\vec{v} \neq 0$.

6.4 Security

The proofs of Lemmas 4–12 are given in Appendix "Proofs of Lemmas 4–12 in Sect. 6".

Theorem 1 *The proposed NIPE scheme is adaptively payload-hiding against chosen plaintext attacks under the DLIN assumption.*

For any machine $\mathcal{A}$, there exist probabilistic machines $\mathcal{E}_1, \mathcal{E}_{2\text{-}1}$ and $\mathcal{E}_{2\text{-}2}$ whose running times are essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{NIPE,PH}}(\lambda) \leq \mathsf{Adv}_{\mathcal{E}_1}^{\mathsf{DLIN}}(\lambda) + \sum_{h=1}^{\nu}\left(\mathsf{Adv}_{\mathcal{E}_{2\text{-}h\text{-}1}}^{\mathsf{DLIN}}(\lambda) + \mathsf{Adv}_{\mathcal{E}_{2\text{-}h\text{-}2}}^{\mathsf{DLIN}}(\lambda)\right) + \epsilon$, where $\mathcal{E}_{2\text{-}h\text{-}1}(\cdot) :=\mathcal{E}_{2\text{-}1}(h, \cdot)$, $\mathcal{E}_{2\text{-}h\text{-}2}(\cdot) :=\mathcal{E}_{2\text{-}2}(h, \cdot)$, ν is the maximum number of $\mathcal{A}$'s key queries and $\epsilon :=(11\nu + 6)/q$.

6.4.1 Lemmas for the Proof of Theorem 1

We will show Lemmas 4–6 for the proof of Theorem 1.

Definition 8 (*Problem 1*) Problem 1 is to guess β, given $(\mathsf{param}_n, \mathbb{B}_0, \widehat{\mathbb{B}}_0^*, \boldsymbol{e}_{\beta,0},\ \{B_{i,j}, B'_{i,j,l}\}_{i,j=1,\dots,4;l=1,\dots,n}, \widehat{\mathbb{B}}_1^*, \{E_{\beta,j}, E'_{\beta,j,l}\}_{j=1,\dots,4;l=1,\dots,n}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\beta}^{\mathsf{P1}}(1^\lambda, n)$, where

$$\mathcal{G}_{\beta}^{\mathsf{P1}}(1^\lambda, n) :\quad (\mathsf{param}_n, \mathbb{B}_0, \mathbb{B}_0^*, \{B_{i,j}, B'_{i,j,l}\}_{i,j=1,\dots,4;l=1,\dots,n}, \widehat{\mathbb{B}}_1^*) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}^{\mathsf{NIPE,CT}}(1^\lambda, 4, n),$$
$$\widehat{\mathbb{B}}_0^* :=(\boldsymbol{b}_{0,1}^*, \boldsymbol{b}_{0,3}^*, \dots, \boldsymbol{b}_{0,5}^*),\ \widehat{\mathbb{B}}_1^* :=(\boldsymbol{b}_{1,1}^*, \dots, \boldsymbol{b}_{1,n}^*, \boldsymbol{b}_{t,2n+1}^*, \dots, \boldsymbol{b}_{t,4n}^*),$$
$$\omega, \tau, \eta_0, \eta_1 \xleftarrow{\mathsf{U}} \mathbb{F}_q,\ U \xleftarrow{\mathsf{U}} \mathcal{H}(n, \mathbb{F}_q) \cap GL(n, \mathbb{F}_q),\ \text{hereafter, } u, u'_n \in \mathbb{F}_q^{\times},$$
$$u'_1, \dots, u'_{n-1} \in \mathbb{F}_q \text{ denote non-zero entries of } U, \text{ as in Eq. (1)},$$
$$\boldsymbol{e}_{0,0} :=(\omega, 0, 0, 0, \eta_0)_{\mathbb{B}_0},\quad \boldsymbol{e}_{1,0} :=(\omega, \tau, 0, 0, \eta_0)_{\mathbb{B}_0},$$
$$\text{for } j = 1, \dots, 4;$$
$$E_{0,j} :=\omega B_{1,j} + \eta_1 B_{4,j},\ E'_{0,j,l} :=\omega B'_{1,j,l} + \eta_1 B'_{4,j,l} \text{ for } l = 1, \dots, n,$$
$$E_{1,j} :=\omega B_{1,j} + \tau u B_{2,j} + \eta_1 B_{4,j},$$
$$E'_{1,j,l} :=\omega B'_{1,j,l} + \tau u B'_{2,j,l} + \tau u'_l B'_{2,j,n} + \eta_1 B'_{4,j,l}$$
$$\text{for } l = 1, \dots, n-1, \text{ and } E'_{1,j,n} :=\omega B'_{1,j,n} + \tau u'_n B'_{2,j,n} + \eta_1 B'_{4,j,n},$$
$$\text{return } (\mathsf{param}_n, \mathbb{B}_0, \widehat{\mathbb{B}}_0^*, \boldsymbol{e}_{\beta,0}, \{B_{i,j}, B'_{i,j,l}\}_{i,j=1,\dots,4;l=1,\dots,n}, \widehat{\mathbb{B}}_1^*,$$
$$\{E_{\beta,j}, E'_{\beta,j,l}\}_{j=1,\dots,4;l=1,\dots,n}),$$

for $\beta \overset{\mathsf{U}}{\leftarrow} \{0,1\}$. For a probabilistic machine $\mathcal{B}$, we define the advantage of $\mathcal{B}$ as the quantity $\mathsf{Adv}^{\mathsf{P1}}_{\mathcal{B}}(\lambda) := \left| \Pr\left[\mathcal{B}(1^\lambda, \varrho) \to 1 \,\middle|\, \varrho \overset{\mathsf{R}}{\leftarrow} \mathcal{G}^{\mathsf{P1}}_0(1^\lambda, n) \right] - \Pr\left[\mathcal{B}(1^\lambda, \varrho) \to 1 \,\middle|\, \varrho \overset{\mathsf{R}}{\leftarrow} \mathcal{G}^{\mathsf{P1}}_1(1^\lambda, n) \right] \right|$.

Remark 4 A part of output of $\mathcal{G}^{\mathsf{P1}}_\beta(1^\lambda, n)$, $\{B_{i,j}, B'_{i,j,l}\}_{i,j=1,\ldots,4;l=1,\ldots,n}$, is identified with $\mathbb{B}_1 := (\boldsymbol{b}_{1,1}, \ldots, \boldsymbol{b}_{1,4n})$ [(Eq. (6)]. If we make $\boldsymbol{e}_{\beta,1,l} \in \mathbb{V}_1$ for $\beta = 0, 1; l = 1, \ldots, n$ as:

$$\begin{aligned}
\boldsymbol{e}_{\beta,1,l} &:= \Big(\overbrace{0^{l-1},\ E_{\beta,1},\ 0^{n-l-1},\ E'_{\beta,1,l}}^{n},\ \ldots,\ \overbrace{0^{l-1},\ E_{\beta,4},\ 0^{n-l-1},\ E'_{\beta,4,l}}^{n} \Big) \\
&\qquad\qquad\qquad\qquad\qquad\qquad \text{for } l = 1, \ldots, n-1, \\
\boldsymbol{e}_{\beta,1,n} &:= \Big(\qquad 0^{n-1},\ E'_{\beta,1,n},\qquad \ldots,\qquad 0^{n-1},\ E'_{\beta,4,n} \qquad \Big),
\end{aligned}$$

they are expressed over $\mathbb{B}_1$ as:

$$\begin{aligned}
\boldsymbol{e}_{0,1,l} &:= (\overbrace{\omega\vec{e}_l,}^{n} \quad \overbrace{0^n,}^{n} \quad \overbrace{0^n,}^{n} \quad \overbrace{\eta_1\vec{e}_l}^{n})_{\mathbb{B}_1} \quad \text{for } l = 1, \ldots, n, \\
\boldsymbol{e}_{1,1,l} &:= (\omega\vec{e}_l, \quad \tau\vec{e}_l U, \quad 0^n, \quad \eta_1\vec{e}_l)_{\mathbb{B}_1} \quad \text{for } l = 1, \ldots, n.
\end{aligned}$$

Using these vector expressions, the output of $\mathcal{G}^{\mathsf{P1}}_\beta(1^\lambda, n)$ is expressed as $(\mathsf{param}_n, \mathbb{B}_0, \widehat{\mathbb{B}}^*_0, \boldsymbol{e}_{\beta,0}, \mathbb{B}_1, \widehat{\mathbb{B}}^*_1, \{\boldsymbol{e}_{\beta,1,l}\}_{l=1,\ldots,n})$.

Lemma 4 *For any machine $\mathcal{B}$, there exists a probabilistic machine $\mathcal{E}$, whose running times are essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}^{\mathsf{P1}}_{\mathcal{B}}(\lambda) \le \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}}(\lambda) + 5/q$.*

Definition 9 (*Problem 2*) Problem 2 is to guess β, given $(\mathsf{param}_n, \widehat{\mathbb{B}}_0, \mathbb{B}^*_0, \boldsymbol{h}^*_{\beta,0}, \boldsymbol{e}_0, \{B_{i,j}, B'_{i,j,l}\}_{i=1,3,4;\,j=1,\ldots,4;l=1,\ldots,n}, \mathbb{B}^*_1, \{\boldsymbol{h}^*_{\beta,1,l}, E_j, E'_{j,l}\}_{j=1,\ldots,4;l=1,\ldots,n}) \overset{\mathsf{R}}{\leftarrow} \mathcal{G}^{\mathsf{P2}}_\beta(1^\lambda, n)$, where

$$\begin{aligned}
&\mathcal{G}^{\mathsf{P2}}_\beta(1^\lambda, n) : \quad (\mathsf{param}_n, \mathbb{B}_0, \widehat{\mathbb{B}}^*_0, \{B_{i,j}, B'_{i,j,l}\}_{i,j=1,\ldots,4;l=1,\ldots,n}, \widehat{\mathbb{B}}^*_1) \overset{\mathsf{R}}{\leftarrow} \mathcal{G}^{\mathsf{NIPE,CT}}_{\mathsf{ob}}(1^\lambda, 4, n), \\
&\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \ldots, \boldsymbol{b}_{0,5}), \quad \delta, \rho, \varphi_0, \omega, \tau \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q,\ \vec{\varphi}_l \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q^n \text{ for } l = 1, \ldots, n, \\
&U \overset{\mathsf{U}}{\leftarrow} \mathcal{H}(n, \mathbb{F}_q) \cap GL(n, \mathbb{F}_q),\ Z := (U^{-1})^{\mathrm{T}}, \\
&\text{hereafter, } u, u'_n \in \mathbb{F}_q^\times, u'_1, \ldots, u'_{n-1} \in \mathbb{F}_q \text{ and } z, z'_n \in \mathbb{F}_q^\times, z'_1, \ldots, z'_{n-1} \in \mathbb{F}_q \\
&\ \text{denote non-zero entries of } U \text{ and } Z^{\mathrm{T}},\ \text{as in Eq. (1)},\ \text{respectively}, \\
&\boldsymbol{h}^*_{0,0} := (\delta, 0, 0, \varphi_0, 0)_{\mathbb{B}^*_0},\ \boldsymbol{h}^*_{1,0} := (\delta, \rho, 0, \varphi_0, 0)_{\mathbb{B}^*_0},\ \boldsymbol{e}_0 := (\omega, \tau, 0, 0, 0)_{\mathbb{B}_0}, \\
&\vec{e}_l := (0^{l-1}, 1, 0^{n-l}) \in \mathbb{F}_q^n \ \text{ for } l = 1, \ldots, n; \\
&\boldsymbol{h}^*_{0,1,l} := (\overbrace{\delta\vec{e}_l,}^{n} \quad \overbrace{0^n,}^{n} \quad \overbrace{\vec{\varphi}_l,}^{n} \quad \overbrace{0^n}^{n})_{\mathbb{B}^*_1} \ \text{ for } l = 1, \ldots, n, \\
&\boldsymbol{h}^*_{1,1,l} := (\delta\vec{e}_l, \quad \rho\vec{e}_l Z, \quad \vec{\varphi}_l, \quad 0^n)_{\mathbb{B}^*_1} \ \text{ for } l = 1, \ldots, n, \\
&\text{for } j = 1, \ldots, 4; \quad E_j := \omega B_{1,j} + \tau u B_{2,j}, \\
&\quad E'_{j,l} := \omega B'_{1,j,l} + \tau u B'_{2,j,l} + \tau u'_l B'_{2,j,n} \ \ \text{for } l = 1, \ldots, n-1,
\end{aligned}$$

$$\begin{array}{l} \quad E'_{j,n} :=\omega B'_{1,j,n} + \tau u'_n B'_{2,j,n}, \\ \text{return } (\mathsf{param}_n, \widehat{\mathbb{B}}_0, \mathbb{B}^*_0, \boldsymbol{h}^*_{\beta,0}, \boldsymbol{e}_0, \{B_{i,j}, B'_{i,j,l}\}_{i=1,3,4;\, j=1,\ldots,4;\, l=1,\ldots,n}, \mathbb{B}^*_1, \\ \qquad \{\boldsymbol{h}^*_{\beta,1,l}, E_j, E'_{j,l}\}_{j=1,\ldots,4;\, l=1,\ldots,n}), \end{array}$$

for $\beta \overset{\mathsf{U}}{\leftarrow} \{0,1\}$. For a probabilistic adversary $\mathcal{B}$, the advantage of $\mathcal{B}$ for Problem 2, $\mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}}(\lambda)$, is similarly defined as in Definition 8.

Remark 5 A part of output of $\mathcal{G}^{\mathsf{P2}}_{\beta}(1^{\lambda}, n)$, $\{B_{i,j}, B'_{i,j,l}\}_{i=1,3,4;\, j=1,\ldots,4;\, l=1,\ldots,n}$, can be identified with $\widehat{\mathbb{B}}_1 :=(\boldsymbol{b}_{1,1}, \ldots, \boldsymbol{b}_{1,n}, \boldsymbol{b}_{1,2n+1}, \ldots, \boldsymbol{b}_{1,4n})$ in the form of Eq. (6), while $\mathbb{B}_1 :=(\boldsymbol{b}_{1,1}, \ldots, \boldsymbol{b}_{1,4n})$ is identified with $\{B_{i,j}, B'_{i,j,l}\}_{i,j=1,\ldots,4;\, l=1,\ldots,n}$ by Eq. (6). If we make $\boldsymbol{e}_{1,l} \in \mathbb{V}_1$ for $l = 1, \ldots, n$ as:

$$\begin{array}{l} \boldsymbol{e}_{1,l} := (\overbrace{0^{l-1},\ E_1,\ 0^{n-l-1},\ E'_{1,l}}^{n},\ \ldots,\ \overbrace{0^{l-1},\ E_4,\ 0^{n-l-1},\ E'_{4,l}}^{n}) \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{for } l = 1, \ldots, n-1, \\ \boldsymbol{e}_{1,n} := (\qquad 0^{n-1},\ E'_{1,n}, \qquad \ldots, \qquad 0^{n-1},\ E'_{4,n} \qquad), \end{array}$$

they are expressed over $\mathbb{B}_1$ as:

$$\boldsymbol{e}_{1,l} := (\overbrace{\omega \vec{e}_l,}^{n} \ \overbrace{\tau \vec{e}_l U,}^{n} \ \overbrace{0^n,}^{n} \ \overbrace{0^n}^{n})_{\mathbb{B}_1} \quad \text{for } l = 1, \ldots, n.$$

Using these vector expressions, the output of $\mathcal{G}^{\mathsf{P2}}_{\beta}(1^{\lambda}, n)$ is expressed as $(\mathsf{param}_n, \widehat{\mathbb{B}}_0, \mathbb{B}^*_0, \boldsymbol{h}^*_{\beta,0}, \boldsymbol{e}_0, \widehat{\mathbb{B}}_1, \mathbb{B}^*_1, \{\boldsymbol{h}^*_{\beta,1,l}, \boldsymbol{e}_{1,l}\}_{l=1,\ldots,n})$.

Lemma 5 *For any machine $\mathcal{B}$, there exists a probabilistic machine $\mathcal{E}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}}(\lambda) \le \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}}(\lambda) + 5/q$.*

Lemma 6 *Let $\vec{e}_n :=(0, \ldots, 0, 1) \in \mathbb{F}_q^n$. For all $\vec{x} \in \mathbb{F}_q^n \setminus \mathsf{span}\langle \vec{e}_n \rangle$ and $\pi \in \mathbb{F}_q$, let $W_{\vec{x},\pi} :=\{(\vec{r}, \vec{w}) \in (\mathsf{span}\langle \vec{x}, \vec{e}_n \rangle \setminus \mathsf{span}\langle \vec{e}_n \rangle) \times (\mathbb{F}_q^n \setminus \mathsf{span}\langle \vec{e}_n \rangle^{\perp}) \mid \vec{r} \cdot \vec{w} = \pi\}$.*

For all $(\vec{x}, \vec{v}) \in \left(\mathbb{F}_q^n \setminus \mathsf{span}\langle \vec{e}_n \rangle\right) \times \left(\mathbb{F}_q^n \setminus \mathsf{span}\langle \vec{e}_n \rangle^{\perp}\right)$, for all $(\vec{r}, \vec{w}) \in W_{\vec{x},(\vec{x}\cdot\vec{v})}$, $\Pr\left[\,\vec{x}U = \vec{r} \wedge \ \vec{v}Z = \vec{w}\,\right] = 1 \big/ \sharp\, W_{\vec{x},(\vec{x}\cdot\vec{v})}$, where $U \overset{\mathsf{U}}{\leftarrow} \mathcal{H}(n, \mathbb{F}_q) \cap GL(n, \mathbb{F}_q)$ and $Z :=(U^{-1})^{\mathrm{T}}$.

6.4.2 Proof outline

At the top level of strategy of the security proof, we follow the dual system encryption methodology proposed by Waters [31]. In the methodology, ciphertexts and secret keys have two forms, *normal* and *semi-functional*. In the proof herein, we also introduce other forms of secret keys called *1st-pre-semi-functional* and *2nd-pre-semi-functional*. The real system uses only normal ciphertexts and normal secret keys, and semi-functional ciphertexts and semi-functional/1st-pre-semi-functional/2nd-pre-semi-functional keys are used only in a sequence of security games for the security proof. To prove this theorem, we employ Game 0 (original adaptive-security game) through Game 3. In Game 1, the challenge ciphertext is changed to semi-functional. When at most ν secret key queries are issued by an adversary, there are 3ν game changes from Game 1 (Game 2-0-3), Game 2-1-1, Game 2-1-2, Game 2-1-3 through Game 2-ν-3.

In Game 2-h-1, the first $(h-1)$ keys are semi-functional and the h-th key is *1st-pre-semi-functional*, while the remaining keys are normal, and the challenge ciphertext is semi-functional. In Game 2-h-2, the first $(h-1)$ keys are semi-functional and the h-th key is *2nd-pre-semi-functional*, while the remaining keys are normal, and the challenge ciphertext is semi-functional. In Game 2-h-3, the first h keys are semi-functional (i.e., and the h-th key is *semi-functional*), while the remaining keys are normal, and the challenge ciphertext is semi-functional.

The final game (Game 3) with advantage 0 is conceptually changed from Game 2-ν-3. As usual, we prove that the advantage gaps between neighboring games are negligible.

When at most ν key queries are issued by an adversary, we set a sequence of $\mathsf{sk} :=\mathsf{sk}_{\vec{v}}$'s, i.e., $(\mathsf{sk}^{(1)*}, \ldots, \mathsf{sk}^{(\nu)*})$, in the order of the adversary's queries. Here we focus on $\vec{\boldsymbol{k}}_{\vec{v}}^{(h)*} :=(\boldsymbol{k}_0^{(h)*}, \boldsymbol{k}_1^{(h)*})$, and $\vec{\boldsymbol{c}}_{\vec{x}} :=(\boldsymbol{c}_0, \{C_{1,j}, C_{2,j}\}_{j=1,\ldots,4}, c_3)$, and ignore the other part of $\mathsf{sk}_{\vec{v}}$ (resp. $\mathsf{ct}_{\vec{x}}$), i.e., $\vec{v}$ (resp. i.e., $\vec{x}$), and call them secret key and ciphertext, respectively, in this proof outline. In addition, we ignore a negligible factor in the (informal) descriptions of this proof outline. For example, we say "A is bounded by B" when $A \leq B + \epsilon(\lambda)$ where $\epsilon(\lambda)$ is negligible in security parameter λ.

A *normal* secret key, $\vec{\boldsymbol{k}}_{\vec{v}}^{(h)*\mathsf{norm}}$, is the correct form of the secret key of the proposed NIPE scheme, and is expressed by Eq. (7). Similarly, a *normal* ciphertext $\vec{\boldsymbol{c}}_{\vec{x}}^{\mathsf{norm}}$, is expressed by Eq. (8). A *1st-pre-semi-functional* secret key, $\vec{\boldsymbol{k}}_{\vec{v}}^{(h)*\ \mathsf{1st\text{-}psemi}}$, is expressed by Eq. (10), a *2nd-pre-semi-functional* secret key, $\vec{\boldsymbol{k}}_{\vec{v}}^{(h)*\ \mathsf{2nd\text{-}psemi}}$, is expressed by Eq. (11), a *semi-functional* secret key, $\vec{\boldsymbol{k}}_{\vec{v}}^{(h)*\ \mathsf{semi}}$, is expressed by Eq. (12), and a *semi-functional* ciphertext, $\vec{\boldsymbol{c}}_{\vec{x}}^{\mathsf{semi}}$, is expressed by Eq. (9).

To prove that the advantage gap between Games 0 and 1 is bounded by the advantage of Problem 1 (to guess $\beta \in \{0, 1\}$), we construct a simulator of the challenger of Game 0 (or 1) (against an adversary $\mathcal{A}$) by using an instance with $\beta \overset{\mathsf{U}}{\leftarrow} \{0, 1\}$ of Problem 1. We then show that the distribution of the secret keys and challenge ciphertext replied by the simulator is equivalent to those of Game 0 when $\beta = 0$ and Game 1 when $\beta = 1$. That is, the advantage gap between Games 0 and 1 is bounded by the advantage of Problem 1 (Lemma 7). The advantage of Problem 1 is proven to be bounded by that of the DLIN assumption (Lemma 4). The advantage gap between Games 2-$(h-1)$-3 and 2-h-1 is similarly shown to be bounded by the advantage of Problem 2 (i.e., advantage of the DLIN assumption) (Lemmas 8 and 5). The distributions of *1st-pre-semi-functional* secret key $\vec{\boldsymbol{k}}_{\vec{v}}^{(h)*\ \mathsf{1st\text{-}psemi}}$ (Eq. (10)) and *2nd-pre-semi-functional* secret key $\vec{\boldsymbol{k}}_{\vec{v}}^{(h)*\ \mathsf{2nd\text{-}psemi}}$ (Eq. (11)) are distinguishable by the simulator or challenger, but the joint distributions of $(\vec{\boldsymbol{k}}_{\vec{v}}^{(h)*\ \mathsf{1st\text{-}psemi}}, \vec{\boldsymbol{c}}_{\vec{x}}^{\mathsf{semi}})$ and $(\vec{\boldsymbol{k}}_{\vec{v}}^{(h)*\ \mathsf{2nd\text{-}psemi}}, \vec{\boldsymbol{c}}_{\vec{x}}^{\mathsf{semi}})$ along with the other keys are (information theoretically) equivalent for the adversary's view, when $\vec{x}\cdot\vec{v} = 0$, i.e., $R^{\mathsf{NIPE}}(\vec{x}, \vec{v}) \neq 1$. Therefore, as shown in Lemma 9, the advantages of Games 2-h-1 and 2-h-2 are equivalent. The advantage gap between Games 2-h-2 and 2-h-3 is similarly shown to be bounded by the advantage of Problem 2 (i.e., advantage of the DLIN assumption) (Lemmas 10 and 5). Finally we show that Game 2-ν-3 can be conceptually changed to Game 3 (Lemma 11) by using the fact that basis vectors $\boldsymbol{b}_{0,2}$ and $\boldsymbol{b}_{0,3}^*$ are unknown to the adversary.

6.4.3 Proof of Theorem 1

To prove Theorem 1, we consider the following $(3\nu+3)$ games. In Game 0, a part framed by a box indicates coefficients to be changed in a subsequent game. In the other games, a part framed by a box indicates coefficients that were changed in a game from the previous game.

Game 0 Original game. That is, the reply to a key query for $\vec{v}$ is

$$\boldsymbol{k}_0^* := (\delta,\ \boxed{0},\ 1,\ \varphi_0,\ 0)_{\mathbb{B}_0^*}, \qquad \boldsymbol{k}_1^* := (\delta\vec{v},\ \boxed{0^n},\ \vec{\varphi}_1,\ 0^n)_{\mathbb{B}_1^*}, \tag{7}$$

where $\delta, \varphi_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q, \vec{\varphi}_1 \xleftarrow{\mathsf{U}} \mathbb{F}_q^n$ and $\vec{v} := (v_1, \ldots, v_n) \in \mathbb{F}_q^n$ with $v_n \neq 0$. The challenge ciphertext for challenge plaintexts $(m^{(0)}, m^{(1)})$ and $\vec{x}$, $(\vec{x}, \boldsymbol{c}_0, \{C_{1,j}, C_{2,j}\}_{j=1,\ldots,4}, c_3)$, which is identified with $(\vec{x}, \boldsymbol{c}_0, \boldsymbol{c}_1, c_3)$ in Remark 3, is

$$\boldsymbol{c}_0 := (-\omega,\ \boxed{0},\ \boxed{\zeta},\ 0,\ \eta_0)_{\mathbb{B}_0},\ \ \boldsymbol{c}_1 := (\omega\vec{x},\ \boxed{0^n},\ 0^n,\ \eta_1\vec{x})_{\mathbb{B}_1},\ c_3 := g_T^{\zeta} m, \tag{8}$$

where $b \xleftarrow{\mathsf{U}} \{0,1\}$; $\omega, \zeta, \eta_0, \eta_1 \xleftarrow{\mathsf{U}} \mathbb{F}_q$ and $\vec{x} := (x_1, \ldots, x_n) \in \mathbb{F}_q^n$ with $x_l \neq 0$ for some $l \in \{1, \ldots, n-1\}$.

Game 1 Same as Game 0 except that the challenge ciphertext for challenge plaintexts $(m^{(0)}, m^{(1)})$ and $\vec{x}$ is

$$\boldsymbol{c}_0 := (-\omega,\ \boxed{-\tau},\ \zeta,\ 0,\ \eta_0)_{\mathbb{B}_0},\ \ \boldsymbol{c}_1 := (\omega\vec{x},\ \boxed{\tau\vec{x}U},\ 0^n,\ \eta_1\vec{x})_{\mathbb{B}_1},\ c_3 := g_T^{\zeta} m, \tag{9}$$

where $\tau \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $U \xleftarrow{\mathsf{U}} \mathcal{H}(n, \mathbb{F}_q) \cap GL(n, \mathbb{F}_q)$, and all the other variables are generated as in Game 0.

Game 2-*h*-1(h= **1**, …, ***v***) Game 2-0-3 is Game 1. Game 2-h-1 is the same as Game 2-$(h-1)$-3 except that the reply to the h-th key query for $\vec{v}$, $(\boldsymbol{k}_0^*, \boldsymbol{k}_1^*)$, is

$$\boldsymbol{k}_0^* := (\delta,\ \boxed{\rho},\ 1,\ \varphi_0,\ 0)_{\mathbb{B}_0^*}, \qquad \boldsymbol{k}_1^* := (\delta\vec{v},\ \boxed{\rho\vec{v}Z},\ \vec{\varphi}_1,\ 0^n)_{\mathbb{B}_1^*}, \tag{10}$$

where $\rho \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $Z := (U^{-1})^{\mathrm{T}}$ for $U \xleftarrow{\mathsf{U}} \mathcal{H}(n, \mathbb{F}_q) \cap GL(n, \mathbb{F}_q)$ used in Eq. (9) and all the other variables are generated as in Game 2-$(h-1)$-3.

Game 2-*h*-2(*h*= 1, …, *v*) Game 2-h-2 is the same as Game 2-h-1 except that a part of the reply to the h-th key query for $\vec{v}$, $(\boldsymbol{k}_0^*, \boldsymbol{k}_1^*)$, is

$$\boldsymbol{k}_0^* := (\delta,\ \boxed{w},\ 1,\ \varphi_0,\ 0)_{\mathbb{B}_0^*}, \qquad \boldsymbol{k}_1^* := (\delta\vec{v},\ \rho\vec{v}Z,\ \vec{\varphi}_1,\ 0^n)_{\mathbb{B}_1^*}, \tag{11}$$

where $w \xleftarrow{\mathsf{U}} \mathbb{F}_q$ and all the other variables are generated as in Game 2-h-1.

Game 2-*h*-3(*h*= 1, …, *v*) Game 2-h-3 is the same as Game 2-h-2 except that the reply to the h-th key query for $\vec{v}$, $(\boldsymbol{k}_0^*, \boldsymbol{k}_1^*)$, is

$$\boldsymbol{k}_0^* := (\delta,\ w,\ 1,\ \varphi_0,\ 0)_{\mathbb{B}_0^*}, \qquad \boldsymbol{k}_1^* := (\delta\vec{v},\ \boxed{0^n},\ \vec{\varphi}_1,\ 0^n)_{\mathbb{B}_1^*}, \tag{12}$$

where all the variables are generated as in Game 2-h-2.

Game 3 Same as Game 2-ν-3 except that $\boldsymbol{c}_0$ and c_3 of the challenge ciphertext are

$$\boldsymbol{c}_0 := (-\omega,\ -\tau,\ \boxed{\zeta'},\ 0,\ \eta_0)_{\mathbb{B}_0}, \quad c_3 := g_T^{\zeta} m^{(b)},$$

where $\zeta' \xleftarrow{\mathsf{U}} \mathbb{F}_q$ (i.e., independent from $\zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$), and all the other variables are generated as in Game 2-ν-3.

Let $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}h\text{-}\iota)}(\lambda)$ $(h = 1, \ldots, \nu; \iota = 1, 2, 3)$ and $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)$ be the advantage of $\mathcal{A}$ in Game 0, 1, 2-h-ι and 3, respectively. $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$ is equivalent to $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{NIPE,PH}}(\lambda)$ and it is obtained that $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda) = 0$ by Lemma 12. We will show five lemmas (Lemmas 7–11) that evaluate the gaps between pairs of $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}h\text{-}\iota)}(\lambda)$

for $h = 1, \ldots, \nu$; $\iota = 1, 2, 3$ and $\mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda)$. From these lemmas and Lemmas 4 and 5, we obtain Theorem 1. □

Lemma 7 *For any machine $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_1$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $|\mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(1)}_{\mathcal{A}}(\lambda)| \leq \mathsf{Adv}^{\mathsf{P1}}_{\mathcal{B}_1}(\lambda)$.*

Lemma 8 *For any machine $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_{2\text{-}1}$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $|\mathsf{Adv}^{(2\text{-}(h-1)\text{-}3)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(2\text{-}h\text{-}1)}_{\mathcal{A}}(\lambda)| \leq \mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}_{2\text{-}h\text{-}1}}(\lambda)$, where $\mathcal{B}_{2\text{-}h\text{-}1}(\cdot) :=\mathcal{B}_{2\text{-}1}(h, \cdot)$.*

Lemma 9 *For any machine $\mathcal{A}$, for any security parameter λ, $|\mathsf{Adv}^{(2\text{-}h\text{-}1)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(2\text{-}h\text{-}2)}_{\mathcal{A}}(\lambda)| \leq 1/q$.*

Lemma 10 *For any machine $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_{2\text{-}2}$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $|\mathsf{Adv}^{(2\text{-}h\text{-}2)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(2\text{-}h\text{-}3)}_{\mathcal{A}}(\lambda)| \leq \mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}_{2\text{-}h\text{-}2}}(\lambda)$, where $\mathcal{B}_{2\text{-}h\text{-}2}(\cdot) :=\mathcal{B}_{2\text{-}2}(h, \cdot)$.*

Lemma 11 *For any machine $\mathcal{A}$, for any security parameter λ, $|\mathsf{Adv}^{(2\text{-}\nu\text{-}3)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda)| \leq 1/q$.*

Lemma 12 *For any machine $\mathcal{A}$, for any security parameter λ, $\mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda) = 0$.*

7 NIPE scheme with constant-size secret-keys

7.1 Dual orthonormal basis generator

We describe random dual orthonormal basis generator $\mathcal{G}^{\mathsf{NIPE,SK}}_{\mathsf{ob}}$ below, which is used as a subroutine in the proposed NIPE scheme, where $\mathcal{G}^{\mathsf{NIPE,CT}}_{\mathsf{ob}}$ is given in Sect. 6.2.

$$\begin{aligned}
&\mathcal{G}^{\mathsf{NIPE,SK}}_{\mathsf{ob}}(1^\lambda, 4, n):\ (\mathsf{param}_n, \mathbb{D}_0, \mathbb{D}^*_0, \{D_{i,j}, D'_{i,j,l}\}_{i,j=1,\ldots,4;l=1,\ldots,n}, \mathbb{D}^*_1) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{NIPE,CT}}_{\mathsf{ob}} \\
&\quad (1^\lambda, 4, n), \\
&\quad \mathbb{B}_0 :=\mathbb{D}^*_0,\ \mathbb{B}^*_0 :=\mathbb{D}_0,\ \mathbb{B}_1 :=\mathbb{D}^*_1,\ B^*_{i,j} :=D_{i,j},\ B'^*_{i,j,l} :=D'_{i,j,l} \\
&\qquad\qquad\qquad\qquad\qquad\qquad \text{for } i, j = 1, \ldots, 4;\ l = 1, \ldots, n, \\
&\quad \text{return } (\mathsf{param}_n, \mathbb{B}_0, \mathbb{B}^*_0, \mathbb{B}_1, \{B^*_{i,j}, B'^*_{i,j,l}\}_{i,j=1,\ldots,4;l=1,\ldots,n}).
\end{aligned}$$

Remark 6 From Remark 2, $\{B^*_{i,j}, B'^*_{i,j,l}\}_{i,j=1,\ldots,4;l=1,\ldots,n}$ is identified with basis $\mathbb{B}^*_1 :=(\boldsymbol{b}^*_{1,1}, \ldots, \boldsymbol{b}^*_{1,4n})$ dual to $\mathbb{B}_1$.

7.2 Construction and security

In the description of the scheme, we assume that input vector, $\vec{v} :=(v_1, \ldots, v_n)$, has an index l $(1 \leq l \leq n-1)$ with $v_l \neq 0$, and that input vector, $\vec{x} :=(x_1, \ldots, x_n)$, satisfies $x_n \neq 0$. The plaintext space is $\mathbb{G}_T$.

$$\mathsf{Setup}(1^{\lambda},\ n) : (\mathsf{param}_n, \mathbb{B}_0, \mathbb{B}_0^*, \mathbb{B}_1, \{B^*_{i,j}, B'^*_{i,j,l}\}_{i,j=1,\ldots,4;\, l=1,\ldots,n}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}^{\mathsf{NIPE,SK}}(1^{\lambda}, 4, n),$$
$$\widehat{\mathbb{B}}_0 :=(\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5}),\ \widehat{\mathbb{B}}_0^* :=(\boldsymbol{b}^*_{0,1}, \boldsymbol{b}^*_{0,3}, \boldsymbol{b}^*_{0,4}),$$
$$\widehat{\mathbb{B}}_1 :=(\boldsymbol{b}_{1,1}, \ldots, \boldsymbol{b}_{1,n}, \boldsymbol{b}_{1,3n+1}, \ldots, \boldsymbol{b}_{1,4n}),$$
$$\text{return } \mathsf{pk} :=(1^{\lambda}, \mathsf{param}_n, \{\widehat{\mathbb{B}}_t\}_{t=0,1}),\ \mathsf{sk} :=(\widehat{\mathbb{B}}_0^*, \{B^*_{i,j}, B'^*_{i,j,l}\}_{i=1,3;\, j=1,\ldots,4;\, l=1,\ldots,n}).$$

$$\mathsf{KeyGen}(\mathsf{pk},\ \mathsf{sk},\ \vec{v}) :\quad \delta, \varphi_0, \varphi_1 \xleftarrow{\mathsf{U}} \mathbb{F}_q,\quad \boldsymbol{k}^*_0 :=(\delta,\ 0,\ 1,\ \varphi_0,\ 0)_{\mathbb{B}_0^*},$$
$$K^*_{1,j} :=\delta B^*_{1,j} + \varphi_1 B^*_{3,j},\quad K^*_{2,j} := \textstyle\sum_{l=1}^{n} v_l(\delta B'^*_{1,j,l} + \varphi_1 B'^*_{3,j,l}) \text{ for } j = 1, \ldots, 4,$$
$$\text{return } \mathsf{sk}_{\vec{v}} :=(\vec{v}, \boldsymbol{k}^*_0, \{K^*_{1,j}, K^*_{2,j}\}_{j=1,\ldots,4}).$$

$$\mathsf{Enc}(\mathsf{pk},\ m,\ \vec{x}) :\quad \omega, \eta_0, \zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q,\ \vec{\eta}_1 \xleftarrow{\mathsf{U}} \mathbb{F}_q^n,\quad \boldsymbol{c}_0 :=(-\omega,\ 0,\ \zeta,\ 0,\ \eta_0)_{\mathbb{B}_0},$$
$$\boldsymbol{c}_1 :=(\omega\vec{x},\ 0^n,\ 0^n,\ \vec{\eta}_1)_{\mathbb{B}_1},\quad c_3 :=g_T^{\zeta} m,\quad \text{return } \mathsf{ct}_{\vec{x}} :=(\vec{x}, \boldsymbol{c}_0, \boldsymbol{c}_1, c_3).$$

$$\mathsf{Dec}(\mathsf{pk},\ \mathsf{sk}_{\vec{v}} :=(\vec{v}, \boldsymbol{k}^*_0, \{K^*_{1,j}, K^*_{2,j}\}_{j=1,\ldots,4}),\ \mathsf{ct}_{\vec{x}} :=(\vec{x}, \boldsymbol{c}_0, \boldsymbol{c}_1, c_3)) :$$
$$\text{Parse } \boldsymbol{c}_1 \text{ as a } 4n\text{-tuple } (C_1, \ldots, C_{4n}) \in \mathbb{G}^{4n},$$
$$D_j := \textstyle\sum_{l=1}^{n-1} ((\vec{x} \cdot \vec{v})^{-1} v_l)\, C_{(j-1)n+l} \text{ for } j = 1, \ldots, 4,$$
$$F :=e(\boldsymbol{c}_0, \boldsymbol{k}^*_0) \cdot \textstyle\prod_{j=1}^{4} \left(e(D_j, K^*_{1,j}) \cdot e(C_{jn}, K^*_{2,j})\right),\quad \text{return } m' :=c_3/F.$$

Remark 7 A part of output of $\mathsf{Setup}(1^{\lambda}, n)$, $\{B^*_{i,j}, B'^*_{i,j,l}\}_{i=1,3;j=1,\ldots,4;l=1,\ldots,n}$, can be identified with $\widehat{\mathbb{B}}_1^* :=(\boldsymbol{b}^*_{1,1}, \ldots, \boldsymbol{b}^*_{1,n}, \boldsymbol{b}^*_{1,2n+1}, \ldots, \boldsymbol{b}^*_{1,3n})$, while $\mathbb{B}_1^* :=(\boldsymbol{b}^*_{1,1}, \ldots, \boldsymbol{b}^*_{1,4n})$ is identified with $\{B^*_{i,j}, B'^*_{i,j,l}\}_{i,j=1,\ldots,4;l=1,\ldots,n}$ in Remark 6. Decryption Dec can be alternatively described as:

$$\mathsf{Dec}'(\mathsf{pk},\ \mathsf{sk}_{\vec{v}} :=(\vec{v}, \boldsymbol{k}^*_0, \{K^*_{1,j}, K^*_{2,j}\}_{j=1,\ldots,4}),\ \mathsf{ct}_{\vec{x}} :=(\vec{x}, \boldsymbol{c}_0, \boldsymbol{c}_1, c_3)) :$$
$$\boldsymbol{k}^*_1 := (\ \overbrace{v_1 K^*_{1,1}, \ldots, v_{n-1} K^*_{1,1}, K^*_{2,1}}^{n},\ \ldots,\ \overbrace{v_1 K^*_{1,4}, \ldots, v_{n-1} K^*_{1,4}, K^*_{2,4}}^{n}\ ),$$
$$\text{that is, } \boldsymbol{k}^*_1 = (\delta\vec{v},\ 0^n,\ 0^n,\ \varphi_1\vec{v})_{\mathbb{B}_1^*},\ F :=e(\boldsymbol{c}_0, \boldsymbol{k}^*_0) \cdot e((\vec{x} \cdot \vec{v})^{-1}\boldsymbol{c}_1, \boldsymbol{k}^*_1),$$
$$\text{return } m' :=c_3/F.$$

Theorem 2 *The proposed NIPE scheme is adaptively payload-hiding against chosen plaintext attacks under the DLIN assumption.*

For any machine $\mathcal{A}$, there exist probabilistic machines $\mathcal{E}_1, \mathcal{E}_{2\text{-}1}$ and $\mathcal{E}_{2\text{-}2}$ whose running times are essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{NIPE,PH}}(\lambda) \le \mathsf{Adv}_{\mathcal{E}_1}^{\mathsf{DLIN}}(\lambda) + \sum_{h=1}^{\nu}\left(\mathsf{Adv}_{\mathcal{E}_{2\text{-}h\text{-}1}}^{\mathsf{DLIN}}(\lambda) + \mathsf{Adv}_{\mathcal{E}_{2\text{-}h\text{-}2}}^{\mathsf{DLIN}}(\lambda)\right) + \epsilon$, where $\mathcal{E}_{2\text{-}h\text{-}1}(\cdot) :=\mathcal{E}_{2\text{-}1}(h, \cdot)$, $\mathcal{E}_{2\text{-}h\text{-}2}(\cdot) :=\mathcal{E}_{2\text{-}2}(h, \cdot)$, ν is the maximum number of $\mathcal{A}$'s key queries and $\epsilon :=(11\nu + 6)/q$.

Theorem 2 is proven similarly to Theorem 1.

8 ZIPE scheme with constant-size ciphertexts

8.1 Dual orthonormal basis generator

We describe random dual orthonormal basis generator $\mathcal{G}_{\mathsf{ob}}^{\mathsf{ZIPE,CT}}$ below, which is used as a subroutine in the proposed Zero IPE scheme. Since the definition is employed for the scheme with $w = 5$ in Sect. 10, we describe $\mathcal{G}_{\mathsf{ob}}^{\mathsf{ZIPE,CT}}$ for general w. (We use only the cases with $w = 4, 5$).

$$\mathcal{G}_{\mathsf{ob}}^{\mathsf{ZIPE,CT}}(1^\lambda, w, n): \ \ \mathsf{param}_{\mathbb{G}} := (q, \mathbb{G}, \mathbb{G}_T, G, e) \overset{\mathsf{R}}{\leftarrow} \mathcal{G}_{\mathsf{bpg}}(1^\lambda), \ N := wn + 1,$$

$$\psi \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q^\times, \ g_T := e(G, G)^\psi, \ \mathsf{param}_{\mathbb{V}} := (q, \mathbb{V}, \mathbb{G}_T, \mathbb{A}, e) := \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, N, \mathsf{param}_{\mathbb{G}}),$$

$$\mathsf{param}_n := (\mathsf{param}_{\mathbb{V}}, \ g_T), \ X \overset{\mathsf{U}}{\leftarrow} \mathcal{L}^+(w, n, \mathbb{F}_q), \text{ hereafter,}$$

$$\{\chi_{0,0}, \chi_{0,j}, \chi_{i,0,l}, \mu_{i,j}, \mu'_{i,j,l}\}_{i,j=1,\ldots w; l=1,\ldots,n} \text{ denotes non-zero entries of } X,$$

$$\text{where } \{\mu_{i,j}, \mu'_{i,j,l}\} \text{ are non-zero entries of submatrices } X_{i,j} \text{ of } X$$

$$\text{as given in Eqs. (5) and (1)}, \ \ (\vartheta_{i,j})_{i,j=0,\ldots,wn} := \psi \cdot (X^{\mathrm{T}})^{-1},$$

$$B_{0,0} := \chi_{0,0} G, \ B_{0,j} := \chi_{0,j} G, \ B_{i,0,l} := \chi_{i,0,l} G, \ B_{i,j} := \mu_{i,j} G, \ B'_{i,j,l} := \mu'_{i,j,l} G$$

$$\text{for } i, j = 1, \ldots, w; l = 1, \ldots, n,$$

$$\boldsymbol{b}_i^* := (\vartheta_{i,1}, \ldots, \vartheta_{i,N})_{\mathbb{A}} = \textstyle\sum_{j=0}^{wn} \vartheta_{i,j} \boldsymbol{a}_j \text{ for } i = 0, \ldots, wn, \ \mathbb{B}^* := (\boldsymbol{b}_0^*, \ldots, \boldsymbol{b}_{wn}^*),$$

$$\text{return } (\mathsf{param}_n, \{B_{0,0}, B_{0,j}, B_{i,0,l}, B_{i,j}, B'_{i,j,l}\}_{i,j=1,\ldots,w; l=1,\ldots,n}, \mathbb{B}^*).$$

Remark 8 $\{B_{0,0}, B_{0,j}, B_{i,0,l}, B_{i,j}, B'_{i,j,l}\}_{i,j=1,\ldots,w; l=1,\ldots,n}$ is identified with basis $\mathbb{B} := (\boldsymbol{b}_0, \ldots, \boldsymbol{b}_{wn})$ dual to $\mathbb{B}^*$ as in Remark 2.

8.2 Construction and security

In the description of the scheme, we assume that input vector, $\vec{x} := (x_1, \ldots, x_n)$, has an index l $(1 \le l \le n - 1)$ with $x_l \neq 0$, and that input vector, $\vec{v} := (v_1, \ldots, v_n)$, satisfies $v_n \neq 0$. The plaintext space is $\mathbb{G}_T$.

$$\mathsf{Setup}(1^\lambda, \ n):$$

$$(\mathsf{param}_n, \{B_{0,0}, B_{0,j}, B_{i,0,l}, B_{i,j}, B'_{i,j,l}\}_{i,j=1,\ldots,4; \, l=1,\ldots,n}, \mathbb{B}^*) \overset{\mathsf{R}}{\leftarrow} \mathcal{G}_{\mathsf{ob}}^{\mathsf{ZIPE,CT}}(1^\lambda, 4, n),$$

$$\widehat{\mathbb{B}}^* := (\boldsymbol{b}_0^*, \ldots, \boldsymbol{b}_n^*, \boldsymbol{b}_{2n+1}^*, \ldots, \boldsymbol{b}_{3n}^*),$$

$$\text{return } \mathsf{pk} := (1^\lambda, \mathsf{param}_n, \{B_{0,0}, B_{0,j}, B_{i,0,l}, B_{i,j}, B'_{i,j,l}\}_{i=1,4; \, j=1,\ldots,4; l=1,\ldots,n}),$$

$$\mathsf{sk} := \widehat{\mathbb{B}}^*.$$

$$\mathsf{KeyGen}(\mathsf{pk}, \ \mathsf{sk}, \ \vec{v}): \ \delta \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q, \ \vec{\varphi} \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q^n, \ \boldsymbol{k}^* := (\,1, \ \overbrace{\delta\vec{v},}^{n} \ \overbrace{0^n,}^{n} \ \overbrace{\vec{\varphi},}^{n} \ \overbrace{0^n}^{n}\ )_{\mathbb{B}^*},$$

$$\text{return } \mathsf{sk}_{\vec{v}} := \boldsymbol{k}^*.$$

$$\mathsf{Enc}(\mathsf{pk}, \ m, \ \vec{x}): \ \ \omega, \eta, \zeta \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q, \ \ C_0 := \zeta B_{0,0} + \textstyle\sum_{l=1}^n x_l (\omega B_{1,0,l} + \eta B_{4,0,l}),$$

$$c_3 := g_T^\zeta m, \ \ C_{1,j} := \omega B_{1,j} + \eta B_{4,j},$$

$$C_{2,j} := \zeta B_{0,j} + \textstyle\sum_{l=1}^n x_l (\omega B'_{1,j,l} + \eta B'_{4,j,l}) \text{ for } j = 1, \ldots, 4,$$

$$\text{return } \mathsf{ct}_{\vec{x}} := (\vec{x}, C_0, \{C_{1,j}, C_{2,j}\}_{j=1,\ldots,4}, c_3).$$

$$\mathsf{Dec}(\mathsf{pk}, \ \mathsf{sk}_{\vec{v}} := \boldsymbol{k}^*, \ \mathsf{ct}_{\vec{x}} := (\vec{x}, C_0, \{C_{1,j}, C_{2,j}\}_{j=1,\ldots,4}, c_3)):$$

$$\text{Parse } \boldsymbol{k}^* \text{ as a } (4n+1)\text{-tuple } (K_0^*, \ldots, K_{4n}^*) \in \mathbb{G}^{4n+1},$$

$$D_j^* := \textstyle\sum_{l=1}^{n-1} x_l K_{(j-1)n+l}^* \text{ for } j = 1, \ldots, 4,$$

$$F := e(C_0, K_0^*) \cdot \textstyle\prod_{j=1}^4 \left(e(C_{1,j}, D_j^*) \cdot e(C_{2,j}, K_{jn}^*) \right), \ \ \text{return } m' := c_3 / F.$$

Remark 9 A part of output of $\mathsf{Setup}(1^{\lambda}, n)$, $\{B_{0,0}, B_{0,j}, B_{i,0,l}, B_{i,j}, B'_{i,j,l}\}_{i=1,4;j=1,\ldots,4;l=1,\ldots,n}$, can be identified with $\widehat{\mathbb{B}} :=(\boldsymbol{b}_0, \ldots, \boldsymbol{b}_n, \boldsymbol{b}_{3n+1}, \ldots, \boldsymbol{b}_{4n})$, while $\mathbb{B} :=(\boldsymbol{b}_0, \ldots, \boldsymbol{b}_{4n})$ is identified with $\{B_{0,0}, B_{0,j}, B_{i,0,l}, B_{i,j}, B'_{i,j,l}\}_{i,j=1,\ldots,4;l=1,\ldots,n}$ in Remark 8. Decryption Dec can be alternatively described as:

$$\mathsf{Dec}'(\mathsf{pk}, \ \mathsf{sk}_{\vec{v}} :=\boldsymbol{k}^*, \ \mathsf{ct}_{\vec{x}} :=(\vec{x}, C_0, \{C_{1,j}, C_{2,j}\}_{j=1,\ldots,4}, c_3)) :$$

$$\boldsymbol{c} := \Big(C_0, \overbrace{x_1 C_{1,1}, \ldots, x_{n-1} C_{1,1}, C_{2,1}}^{n}, \ \ldots, \ \overbrace{x_1 C_{1,4}, \ldots, x_{n-1} C_{1,4}, C_{2,4}}^{n} \Big),$$

$$\text{that is, } \boldsymbol{c} = (\zeta, \ \overbrace{\omega\vec{x}}^{n}, \ \overbrace{0^n}^{n}, \ \overbrace{0^n}^{n}, \ \overbrace{\eta\vec{x}}^{n})_{\mathbb{B}}, \ F :=e(\boldsymbol{c}, \boldsymbol{k}^*), \ \text{return } \ m' :=c_3/F.$$

[Correctness] Using the alternate decryption Dec', $F = e(\boldsymbol{c}, \boldsymbol{k}) = g_T^{\zeta+\omega\delta\vec{x}\cdot\vec{v}} = g_T^{\zeta}$ if $\vec{x}\cdot\vec{v} = 0$.

Remark 10 The proposed ZIPE in this section employs a single basis, $\mathbb{B}$, generated by $X \in GL(4n+1, \mathbb{F}_q)$ [or $X \in \mathcal{L}^{+}(4, n, \mathbb{F}_q)$ of Eq. (5)], and a ciphertext can be expressed as $(\boldsymbol{c}, g_T^{\zeta} m)$ with $\boldsymbol{c} = (\zeta, \ \omega\vec{x}, \ 0^{2n}, \ \eta\vec{x})_{\mathbb{B}}$ as shown in Remark 9. The proposed NIPE scheme in Sect. 6.3 employs two bases, $\mathbb{B}_0$ and $\mathbb{B}_1$, generated by $X_0 \in GL(5, \mathbb{F}_q)$ and $X_1 \in GL(4n, \mathbb{F}_q)$, and a ciphertext can be expressed as $(\boldsymbol{c}_0, \boldsymbol{c}_1, g_T^{\zeta} m)$ with $\boldsymbol{c}_0 :=(-\omega, \ 0, \ \zeta, \ 0, \ \eta_0)_{\mathbb{B}_0}$ and $\boldsymbol{c}_1 = (\omega\vec{x}, \ 0^{2n}, \ \eta_1\vec{x})_{\mathbb{B}_1}$. Hence, the ciphertext and secret key of the ZIPE scheme are shorter than those of the NIPE scheme (see Table 1 in Sect. 11). It is due to the difference of the decryption tricks in the ZIPE and NIPE schemes. Similarly to the fact on $\mathcal{L}(4, n, \mathbb{F}_q)$ (for the security of the NIPE scheme) shown in Sect. 6.1, it is crucial for the security of the ZIPE scheme that $\mathcal{L}^{+}(4, n, \mathbb{F}_q)$ is a subgroup of $GL(4n+1, \mathbb{F}_q)$ (Lemma 3), and its security proof is made in the essentially same manner as explained in Sect. 6.1.

Theorem 3 *The proposed ZIPE scheme is adaptively payload-hiding against chosen plaintext attacks under the DLIN assumption. For any machine $\mathcal{A}$, there exist probabilistic machines $\mathcal{E}_1$ and $\mathcal{E}_2$, whose running times are essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{ZIPE,PH}}(\lambda) \le \mathsf{Adv}_{\mathcal{E}_1}^{\mathsf{DLIN}}(\lambda) + \sum_{h=1}^{\nu} \mathsf{Adv}_{\mathcal{E}_{2\text{-}h}}^{\mathsf{DLIN}}(\lambda) + \epsilon$, where $\mathcal{E}_{2\text{-}h}(\cdot) :=\mathcal{E}_2(h, \cdot)$, ν is the maximum number of $\mathcal{A}$'s key queries, and $\epsilon :=(11\nu+6)/q$.*

Proof To prove Theorem 3, we consider the following $(\nu+3)$ games. In Game 0, a part framed by a box indicates coefficients to be changed in a subsequent game. In the other games, a part framed by a box indicates coefficients that were changed in a game from the previous game.

Game 0 Original game. That is, the reply to a key query for $\vec{v}$ is

$$\boldsymbol{k}^* :=\Big(1, \ \delta\vec{v}, \ \boxed{0^n}, \ \vec{\varphi}, \ 0^n \Big)_{\mathbb{B}^*},$$

where $\delta \xleftarrow{\mathsf{U}} \mathbb{F}_q, \varphi \xleftarrow{\mathsf{U}} \mathbb{F}_q^n$ and $\vec{v} :=(v_1, \ldots, v_n) \in \mathbb{F}_q^n$ with $v_n \neq 0$. The challenge ciphertext for challenge plaintexts $(m^{(0)}, m^{(1)})$ and $\vec{x}$, $(\vec{x}, \boldsymbol{c}_0, \{C_{1,j}, C_{2,j}\}_{j=1,\ldots,4}, c_3)$, which is identified with $(\vec{x}, \boldsymbol{c}, c_3)$ in Remark 9, is

$$\boldsymbol{c} :=\Big(\boxed{\zeta}, \ \omega\vec{x}, \ \boxed{0^n}, \ 0^n, \ \eta\vec{x} \Big)_{\mathbb{B}}, \quad c_3 :=g_T^{\zeta} m,$$

where $b \xleftarrow{\mathsf{U}} \{0,1\}$; $\omega, \zeta, \eta \xleftarrow{\mathsf{U}} \mathbb{F}_q$ and $\vec{x} :=(x_1, \ldots, x_n) \in \mathbb{F}_q^n$ with $x_l \neq 0$ for some $l \in \{1, \ldots, n-1\}$.

Game 1 Same as Game 0 except that the challenge ciphertext for challenge plaintexts $(m^{(0)}, m^{(1)})$ and $\vec{x}$ is

$$\boldsymbol{c} := \left(\zeta,\ \omega\vec{x},\ \boxed{\vec{r}},\ 0^n,\ \eta\vec{x} \right)_{\mathbb{B}}, \quad c_3 := g_T^{\zeta} m,$$

where $\vec{r} \xleftarrow{\mathsf{U}} \mathsf{span}\langle \vec{x}, \vec{e}_n \rangle$, and all the other variables are generated as in Game 0.
Game 2-$\boldsymbol{h}$($\boldsymbol{h}$= 1, ..., $\boldsymbol{\nu}$) Game 2-0 is Game 1. Game 2-h is the same as Game 2-($h-1$) except that a part of the reply to the h-th key query for $\vec{v}$, $\boldsymbol{k}^*$, is

$$\boldsymbol{k}^* := (\ 1,\ \delta\vec{v},\ \boxed{\vec{w}},\ \vec{\varphi},\ 0^n\ )_{\mathbb{B}^*},$$

where $\vec{w} \xleftarrow{\mathsf{U}} \mathbb{F}_q^n$ and all the other variables are generated as in Game 2-($h-1$).
Game 3 Same as Game 2-ν except that $\boldsymbol{c}$ and c_3 of the challenge ciphertext are

$$\boldsymbol{c} := \left(\boxed{\zeta'},\ \omega\vec{x},\ \vec{r},\ 0^n,\ \eta\vec{x} \right)_{\mathbb{B}}, \quad c_3 := g_T^{\zeta} m^{(b)},$$

where $\zeta' \xleftarrow{\mathsf{U}} \mathbb{F}_q$ (i.e., independent from $\zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$), and all the other variables are generated as in Game 2-ν.

Let $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}h)}(\lambda)$ $(h = 1, \ldots, \nu)$ and $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)$ be the advantage of $\mathcal{A}$ in Game 0, 1, 2-h and 3, respectively. $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$ is equivalent to $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{ZIPE,PH}}(\lambda)$ and $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda) = 0$. We can evaluate the gaps between pairs of $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}h)}(\lambda)$ for $h = 1, \ldots, \nu$ using (variants of) Problems 1 and 2 as in the proof of Theorem 1. The following Lemma 13 gives a gap evaluation between $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}\nu\text{-}2)}(\lambda)$ and $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)$, which requires a detailed proof for our ZIPE with constant-size ciphertexts (see Appendix "Proof of Lemma 13 in Sect. 8" for the proof). Combining the gap evaluations, we obtain Theorem 3. □

Lemma 13 *For any machine $\mathcal{A}$, for any security parameter λ, $|\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}\nu)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)| \le 1/q$.*

9 ZIPE scheme with constant-size secret-keys

9.1 Dual orthonormal basis generator

We describe random dual orthonormal basis generator $\mathcal{G}_{\mathsf{ob}}^{\mathsf{ZIPE,SK}}$ below, which is used as a subroutine in the proposed ZIPE scheme, where $\mathcal{G}_{\mathsf{ob}}^{\mathsf{ZIPE,CT}}$ is defined in Sect. 7.1. Since the definition is employed for the scheme with $w = 5$ in Sect. 10, we describe $\mathcal{G}_{\mathsf{ob}}^{\mathsf{ZIPE,SK}}$ for general w. (We use only the cases with $w = 4, 5$).

```
$\mathcal{G}_{\mathsf{ob}}^{\mathsf{ZIPE,SK}}(1^\lambda, w, n)$ :
  $(\mathsf{param}_n, \{D_{0,0}, D_{0,j}, D_{i,0,l}, D_{i,j}, D'_{i,j,l}\}_{i,j=1,\ldots,w;l=1,\ldots,n}, \mathbb{D}^*) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}^{\mathsf{ZIPE,CT}}(1^\lambda, w, n)$,
  $\mathbb{B} := \mathbb{D}^*,\ B^*_{0,0} := D_{0,0},\ B^*_{0,j} := D_{0,j},\ B^*_{i,0,l} := D_{i,0,l},\ B^*_{i,j} := D_{i,j},\ B'^*_{i,j,l} := D'_{i,j,l}$
        for $i, j = 1, \ldots, w;\ l = 1, \ldots, n$,
  return $(\mathsf{param}_n, \mathbb{B}, \{B^*_{0,0}, B^*_{0,j}, B^*_{i,0,l}, B^*_{i,j}, B'^*_{i,j,l}\}_{i,j=1,\ldots,w;l=1,\ldots,n})$.
```

Remark 11 $\{B^*_{0,0}, B^*_{0,j}, B^*_{i,0,l}, B^*_{i,j}, B'^*_{i,j,l}\}_{i,j=1,\ldots,w;l=1,\ldots,n}$ is identified with basis $\mathbb{B}^* :=(\boldsymbol{b}^*_0, \ldots, \boldsymbol{b}^*_{wn})$ dual to $\mathbb{B}$ as in Remark 6.

9.2 Construction and security

In the description of the scheme, we assume that input vector, $\vec{v} :=(v_1, \ldots, v_n)$, has an index l $(1 \le l \le n-1)$ with $v_l \neq 0$, and that input vector, $\vec{x} :=(x_1, \ldots, x_n)$, satisfies $x_n \neq 0$. The plaintext space is $\mathbb{G}_T$.

$\mathsf{Setup}(1^\lambda,\ n) : (\mathsf{param}_n, \mathbb{B},\ \{B^*_{0,0}, B^*_{0,j}, B^*_{i,0,l}, B^*_{i,j}, B'^*_{i,j,l}\}_{i,j=1,\ldots,4;l=1,\ldots,n}) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{ZIPE,SK}}_{\mathsf{ob}}(1^\lambda, 4, n)$,

$\widehat{\mathbb{B}} :=(\boldsymbol{b}_0, \ldots, \boldsymbol{b}_n, \boldsymbol{b}_{3n+1}, \ldots, \boldsymbol{b}_{4n})$,

return $\mathsf{pk} :=(1^\lambda, \mathsf{param}_n, \widehat{\mathbb{B}})$, $\mathsf{sk} :=\{B^*_{0,0}, B^*_{0,j}, B^*_{i,0,l}, B^*_{i,j}, B'^*_{i,j,l}\}_{i=1,3;\,j=1,\ldots,4;l=1,\ldots,n}$.

$\mathsf{KeyGen}(\mathsf{pk},\ \mathsf{sk},\ \vec{v}) :\quad \delta, \varphi \xleftarrow{\mathsf{U}} \mathbb{F}_q,\quad K^*_0 :=B^*_{0,0} + \sum_{l=1}^{n} v_l(\delta B^*_{1,0,l} + \varphi B^*_{3,0,l})$,

$K^*_{1,j} :=\delta B^*_{1,j} + \varphi B^*_{3,j},\quad K^*_{2,j} :=B^*_{0,j} + \sum_{l=1}^{n} v_l(\delta B'^*_{1,j,l} + \varphi B'^*_{3,j,l})$ for $j = 1, \ldots, 4$,

return $\mathsf{sk}_{\vec{v}} :=(\vec{v}, K^*_0, \{K^*_{1,j}, K^*_{2,j}\}_{j=1,\ldots,4})$.

$\mathsf{Enc}(\mathsf{pk},\ m,\ \vec{x}) : \omega, \zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q, \vec{\eta} \xleftarrow{\mathsf{U}} \mathbb{F}_q^n, \boldsymbol{c} :=(\ \zeta,\ \overbrace{\omega\vec{x}}^{n},\ \overbrace{0^n}^{n},\ \overbrace{0^n}^{n},\ \overbrace{\vec{\eta}}^{n}\ )_{\mathbb{B}},\quad c_3 :=g_T^{\zeta} m$,

return $\mathsf{ct}_{\vec{x}} :=(\boldsymbol{c}, c_3)$.

$\mathsf{Dec}(\mathsf{pk},\ \mathsf{sk}_{\vec{v}} :=(\vec{v}, K^*_0, \{K^*_{1,j}, K^*_{2,j}\}_{j=1,\ldots,4}),\ \mathsf{ct}_{\vec{x}} :=(\boldsymbol{c}, c_3)) :$

Parse $\boldsymbol{c}$ as a $(4n+1)$-tuple $(C_0, \ldots, C_{4n}) \in \mathbb{G}^{4n+1}$,

$D_j := \sum_{l=1}^{n-1} v_l C_{(j-1)n+l}$ for $j = 1, \ldots, 4$,

$F :=e(C_0, K^*_0) \cdot \prod_{j=1}^{4} \left(e(D_j, K^*_{1,j}) \cdot e(C_{jn}, K^*_{2,j}) \right)$, return $m' :=c_3/F$.

Remark 12 A part of output of $\mathsf{Setup}(1^\lambda, n)$, $\{B^*_{0,0}, B^*_{0,j}, B^*_{i,0,l}, B^*_{i,j}, B'^*_{i,j,l}\}_{i=1,3;\,j=1,\ldots,4;l=1,\ldots,n}$, can be identified with $\widehat{\mathbb{B}}^* :=(\boldsymbol{b}^*_0, \ldots, \boldsymbol{b}^*_n, \boldsymbol{b}^*_{2n+1}, \ldots, \boldsymbol{b}^*_{3n})$, while $\mathbb{B}^* :=(\boldsymbol{b}^*_0, \ldots, \boldsymbol{b}^*_{4n})$ is identified with $\{B^*_{0,0}, B^*_{0,j}, B^*_{i,0,l}, B^*_{i,j}, B'^*_{i,j,l}\}_{i=1,\ldots,4;\,j=1,\ldots,4;l=1,\ldots,n}$ in Remark 11. Decryption Dec can be alternatively described as:

$\mathsf{Dec}'(\mathsf{pk},\ \mathsf{sk}_{\vec{v}} :=(\vec{v}, K^*_0, \{K^*_{1,j}, K^*_{2,j}\}_{j=1,\ldots,4}),\ \mathsf{ct}_{\vec{x}} :=(\boldsymbol{c}, c_3)) :$

$$\boldsymbol{k}^* := \big(\ K^*_0,\ \overbrace{v_1 K^*_{1,1}, \ldots, v_{n-1} K^*_{1,1}, K^*_{2,1}}^{n},\ \ \ldots,\ \overbrace{v_1 K^*_{1,4}, \ldots, v_{n-1} K^*_{1,4}, K^*_{2,4}}^{n}\ \big),$$

that is, $\boldsymbol{k}^* = (1,\ \overbrace{\delta\vec{v}}^{n},\ \overbrace{0^n}^{n},\ \overbrace{\varphi\vec{v}}^{n}\ \overbrace{0^n}^{n}\ )_{\mathbb{B}^*}$, $\quad F :=e(\boldsymbol{c}, \boldsymbol{k}^*)$,

return $m' :=c_3/F$.

[Correctness] Using the alternate decryption Dec', $F = e(\boldsymbol{c}, \boldsymbol{k}) = g_T^{\zeta + \omega\delta\vec{x}\cdot\vec{v}} = g_T^{\zeta}$ if $\vec{x}\cdot\vec{v} = 0$.

Theorem 4 *The proposed ZIPE scheme is adaptively weakly-attribute-hiding against chosen plaintext attacks under the DLIN assumption. For any machine $\mathcal{A}$, there exist probabilistic machines $\mathcal{E}_1$ and $\mathcal{E}_2$, whose running times are essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $\mathsf{Adv}^{\mathsf{ZIPE,wAH}}_{\mathcal{A}}(\lambda) \le \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_1}(\lambda) + \sum_{h=1}^{\nu} \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_{2\text{-}h}}(\lambda) + \epsilon$, where $\mathcal{E}_{2\text{-}h}(\cdot) :=\mathcal{E}_2(h, \cdot)$, ν is the maximum number of $\mathcal{A}$'s key queries, and $\epsilon :=(11\nu + 6)/q$.*

Proof To prove Theorem 4, we consider the following $(\nu + 3)$ games. In Game 0, a part framed by a box indicates coefficients to be changed in a subsequent game. In the other games, a part framed by a box indicates coefficients that were changed in a game from the previous game.

Game 0 Original game. That is, the reply to a key query for $\vec{v}$ is

$$\boldsymbol{k}^* := (\, 1,\ \delta\vec{v},\ \boxed{0^n},\ \varphi\vec{v},\ 0^n \,)_{\mathbb{B}^*},$$

where $\delta, \varphi \xleftarrow{\mathsf{U}} \mathbb{F}_q$ and $\vec{v} := (v_1, \ldots, v_n) \in \mathbb{F}_q^n$ with $v_n \neq 0$. The challenge ciphertext for challenge plaintexts $(m^{(0)}, m^{(1)})$ and $\vec{x}$, $(\vec{x}, \boldsymbol{c}_0, \{C_{1,j}, C_{2,j}\}_{j=1,\ldots,4}, c_3)$, which is identified with $(\vec{x}, \boldsymbol{c}, c_3)$ in Remark 9, is

$$\boldsymbol{c} := (\, \boxed{\zeta},\ \boxed{\omega\vec{x}},\ \boxed{0^n},\ 0^n,\ \vec{\eta}\,)_{\mathbb{B}}, \quad c_3 := g_T^{\zeta} m,$$

where $b \xleftarrow{\mathsf{U}} \{0, 1\}$; $\omega, \zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $\vec{\eta} \xleftarrow{\mathsf{U}} \mathbb{F}_q^n$ and $\vec{x} := (x_1, \ldots, x_n) \in \mathbb{F}_q^n$ with $x_l \neq 0$ for some $l \in \{1, \ldots, n-1\}$.

Game 1 Same as Game 0 except that the challenge ciphertext for challenge plaintexts $(m^{(0)}, m^{(1)})$ and $\vec{x}$ is

$$\boldsymbol{c} := (\, \zeta,\ \omega\vec{x},\ \boxed{\vec{r}},\ 0^n,\ \vec{\eta}\,)_{\mathbb{B}}, \quad c_3 := g_T^{\zeta} m,$$

where $\vec{r} \xleftarrow{\mathsf{U}} \mathbb{F}_q^n$, and all the other variables are generated as in Game 0.

Game 2-*h*(*h*= 1, . . . , *ν*) Game 2-0 is Game 1. Game 2-h is the same as Game 2-$(h-1)$ except that a part of the reply to the h-th key query for $\vec{v}$, $\boldsymbol{k}^*$, is

$$\boldsymbol{k}^* := (\, 1,\ \delta\vec{v},\ \boxed{\vec{w}},\ \varphi\vec{v},\ 0^n \,)_{\mathbb{B}^*},$$

where $\vec{w} \xleftarrow{\mathsf{U}} \mathsf{span}\langle \vec{v}, \vec{e}_n \rangle$ and all the other variables are generated as in Game 2-$(h-1)$.

Game 3 Same as Game 2-ν except that $\boldsymbol{c}$ and c_3 of the challenge ciphertext are

$$\boldsymbol{c} := (\, \boxed{\zeta'},\ \boxed{\vec{x}'},\ \vec{r},\ 0^n,\ \vec{\eta}\,)_{\mathbb{B}}, \quad c_3 := g_T^{\zeta} m^{(b)},$$

where $\zeta' \xleftarrow{\mathsf{U}} \mathbb{F}_q$ (i.e., independent from $\zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$), $\vec{x}' \xleftarrow{\mathsf{U}} \mathbb{F}_q^n$ (i.e., independent from $\vec{x} \xleftarrow{\mathsf{U}} \mathbb{F}_q^n$), and all the other variables are generated as in Game 2-ν.

Let $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}h)}(\lambda)$ $(h = 1, \ldots, \nu)$ and $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)$ be the advantage of $\mathcal{A}$ in Game 0, 1, 2-h and 3, respectively. $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$ is equivalent to $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{ZIPE,wAH}}(\lambda)$ and $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda) = 0$. We can evaluate the gaps between pairs of $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}h)}(\lambda)$ for $h = 1, \ldots, \nu$ using (variants of) Problems 1 and 2 as in the proof of Theorem 1. The following Lemma 14 gives a gap evaluation between $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}\nu)}(\lambda)$ and $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)$, which requires a detailed proof for our ZIPE with constant-size secret-keys (see Appendix "Proof of Lemma 14 in Sect. 9" for the proof). Combining the gap evaluations, we obtain Theorem 4. □

Lemma 14 *For any machine $\mathcal{A}$, for any security parameter λ,* $|\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}\nu)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)| \leq 1/q$.

10 Fully-attribute-hiding ZIPE scheme with constant-size secret-keys

By applying our technique to the fully-attribute-hiding ZIPE scheme in [27], we obtain a fully-attribute-hiding ZIPE scheme with short secret-keys.

10.1 Construction and security

In the description of the scheme, we assume that input vector, $\vec{v} :=(v_1, \ldots, v_n)$, has an index l $(1 \le l \le n-1)$ with $v_l \neq 0$, and that input vector, $\vec{x} :=(x_1, \ldots, x_n)$, satisfies $x_n \neq 0$. The plaintext space is $\mathbb{G}_T$.

$\mathsf{Setup}(1^\lambda,\ n) : (\mathsf{param}_n, \mathbb{B}, \{B^*_{0,0}, B^*_{0,j}, B^*_{i,0,l}, B^*_{i,j}, B'^*_{i,j,l}\}_{i,j=1,\ldots,5;l=1,\ldots,n}) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{ZIPE,SK}}_{\mathsf{ob}}$
$(1^\lambda, 5, n)$,
$\widehat{\mathbb{B}} :=(\boldsymbol{b}_0, \ldots, \boldsymbol{b}_n, \boldsymbol{b}_{4n+1}, \ldots, \boldsymbol{b}_{5n})$,
return$\mathsf{pk} :=(1^\lambda, \mathsf{param}_n, \widehat{\mathbb{B}})$, $\mathsf{sk} :=\{B^*_{0,0}, B^*_{0,j}, B^*_{i,0,l}, B^*_{i,j}, B'^*_{i,j,l}\}_{i=1,4;\,j=1,\ldots,5;l=1,\ldots,n}$.

$\mathsf{KeyGen}(\mathsf{pk},\ \mathsf{sk},\ \vec{v}) :\quad \delta, \varphi \xleftarrow{\mathsf{U}} \mathbb{F}_q,\quad K^*_0 :=B^*_{0,0} + \sum_{l=1}^{n} v_l(\delta B^*_{1,0,l} + \varphi B^*_{4,0,l})$,
$K^*_{1,j} :=\delta B^*_{1,j} + \varphi B^*_{4,j},\quad K^*_{2,j} :=B^*_{0,j} + \sum_{l=1}^{n} v_l(\delta B'^*_{1,j,l} + \varphi B'^*_{4,j,l})$ for $j = 1, \ldots, 5$,
return $\mathsf{sk}_{\vec{v}} :=(\vec{v}, K^*_0, \{K^*_{1,j}, K^*_{2,j}\}_{j=1,\ldots,5})$.

$\mathsf{Enc}(\mathsf{pk},\ m,\ \vec{x}) :\quad \omega, \zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q,\ \vec{\eta} \xleftarrow{\mathsf{U}} \mathbb{F}_q^n,\quad \boldsymbol{c} :=(\ \zeta,\ \overbrace{\omega\vec{x}}^{n},\ \overbrace{0^{2n}}^{2n},\ \overbrace{0^n}^{n},\ \overbrace{\vec{\eta}}^{n}\ )_{\mathbb{B}}$,
$c_3 :=g_T^{\zeta} m$,
return $\mathsf{ct}_{\vec{x}} :=(\boldsymbol{c}, c_3)$.

$\mathsf{Dec}(\mathsf{pk},\ \mathsf{sk}_{\vec{v}} :=(\vec{v}, K^*_0, \{K^*_{1,j}, K^*_{2,j}\}_{j=1,\ldots,5}),\ \mathsf{ct}_{\vec{x}} :=(\boldsymbol{c}, c_3)) :$
Parse $\boldsymbol{c}$ as a $(5n+1)$-tuple $(C_0, \ldots, C_{5n}) \in \mathbb{G}^{5n+1}$,
$D_j := \sum_{l=1}^{n-1} v_l C_{(j-1)n+l}$ for $j = 1, \ldots, 5$,
$F :=e(C_0, K^*_0) \cdot \prod_{j=1}^{5} \left(e(D_j, K^*_{1,j}) \cdot e(C_{jn}, K^*_{2,j}) \right)$, return $m' :=c_3/F$.

Remark 13 A part of output of $\mathsf{Setup}(1^\lambda, n)$, $\{B^*_{0,0}, B^*_{0,j}, B^*_{i,0,l}, B^*_{i,j}, B'^*_{i,j,l}\}_{i=1,4;\,j=1,\ldots,5;l=1,\ldots,n}$, can be identified with $\widehat{\mathbb{B}}^* :=(\boldsymbol{b}^*_0, \ldots, \boldsymbol{b}^*_n, \boldsymbol{b}^*_{3n+1}, \ldots, \boldsymbol{b}^*_{4n})$, while $\mathbb{B}^* :=(\boldsymbol{b}^*_0, \ldots, \boldsymbol{b}^*_{5n})$ is identified with $\{B^*_{0,0}, B^*_{0,j}, B^*_{i,0,l}, B^*_{i,j}, B'^*_{i,j,l}\}_{i=1,\ldots,5;\,j=1,\ldots,5;l=1,\ldots,n}$ in Remark 11. Decryption Dec can be alternatively described as:

$\mathsf{Dec}'(\mathsf{pk},\ \mathsf{sk}_{\vec{v}} :=(\vec{v}, K^*_0, \{K^*_{1,j}, K^*_{2,j}\}_{j=1,\ldots,5}),\ \mathsf{ct}_{\vec{x}} :=(\boldsymbol{c}, c_3)) :$

$$\boldsymbol{k}^* := \Big(\ K^*_0,\ \overbrace{v_1 K^*_{1,1}, \ldots, v_{n-1} K^*_{1,1}, K^*_{2,1}}^{n},\quad \ldots,\ \overbrace{v_1 K^*_{1,5}, \ldots, v_{n-1} K^*_{1,5}, K^*_{2,5}}^{n}\ \Big),$$

that is, $\boldsymbol{k}^* = (1,\ \overbrace{\delta\vec{v}}^{n},\ \overbrace{0^{2n}}^{2n},\ \overbrace{\varphi\vec{v}}^{n}\ \overbrace{0^n}^{n}\ )_{\mathbb{B}^*},\quad F :=e(\boldsymbol{c}, \boldsymbol{k}^*)$,
return $m' :=c_3/F$.

[Correctness] Using the alternate decryption Dec', $F = e(\boldsymbol{c}, \boldsymbol{k}) = g_T^{\zeta+\omega\delta\vec{x}\cdot\vec{v}} = g_T^{\zeta}$ if $\vec{x}\cdot\vec{v} = 0$.

Theorem 5 *The proposed ZIPE scheme is adaptively fully-attribute-hiding against chosen plaintext attacks under the DLIN assumption.*

For any machine $\mathcal{A}$, there exist probabilistic machines $\mathcal{E}_{0\text{-}1}, \mathcal{E}_{0\text{-}2}, \mathcal{E}_{1\text{-}1}, \mathcal{E}_{1\text{-}2\text{-}1}$ and $\mathcal{E}_{1\text{-}2\text{-}2}$, whose running times are essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $\mathsf{Adv}^{\mathsf{ZIPE,AH}}_{\mathcal{A}}(\lambda) \leq \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_{0\text{-}1}}(\lambda) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_{1\text{-}1}}(\lambda) + \sum_{h=1}^{\nu} \Big(\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_{0\text{-}2\text{-}h}}(\lambda) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_{1\text{-}2\text{-}h\text{-}1}}(\lambda) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_{1\text{-}2\text{-}h\text{-}2}}(\lambda)\Big) + \epsilon$, where $\mathcal{E}_{0\text{-}2\text{-}h}(\cdot) := \mathcal{E}_{0\text{-}2}(h, \cdot)$, $\mathcal{E}_{1\text{-}2\text{-}h\text{-}1}(\cdot) := \mathcal{E}_{1\text{-}2\text{-}1}(h, \cdot)$, $\mathcal{E}_{1\text{-}2\text{-}h\text{-}2}(\cdot) := \mathcal{E}_{1\text{-}2\text{-}2}(h, \cdot)$, ν is the maximum number of $\mathcal{A}$'s key queries and $\epsilon := (29\nu + 17)/q$.

Proof Similarly to the proof of Theorem 1 in [27], the proof of Theorem 5 is reduced to that of Lemma 15.

First, we execute a preliminary game transformation from Game 0 (original security game in Definition 6) to Game 0', which is the same as Game 0 except that flip a coin $t \xleftarrow{\mathsf{U}} \{0, 1\}$ before setup, and the game is aborted in the challenge step if $t \neq s$. We define that $\mathcal{A}$ wins with probability 1/2 when the game is aborted (and the advantage in Game 0' is $\mathsf{Pr}[\mathcal{A} \text{ wins }] - 1/2$ as well). Since t is independent from s, the game is aborted with probability 1/2. Hence, the advantage in Game 0' is a half of that in Game 0, i.e., $\mathsf{Adv}^{\mathsf{IPE,AH,0'}}_{\mathcal{A}}(\lambda) = 1/2 \cdot \mathsf{Adv}^{\mathsf{IPE,AH}}_{\mathcal{A}}(\lambda)$. Moreover, $\Pr[\mathcal{A} \text{ wins}] = 1/2 \cdot (\Pr[\mathcal{A} \text{ wins} \mid t = 0] + \Pr[\mathcal{A} \text{ wins} \mid t = 1])$ in Game 0' since t is uniformly and independently generated.

As for the conditional probability with $t = 0$, it holds that, for any adversary $\mathcal{A}$, there exist probabilistic machines $\mathcal{E}_1$ and $\mathcal{E}_2$, whose running times are essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, in Game 0', $\Pr[\mathcal{A} \text{ wins} \mid t = 0] - 1/2 \leq \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_1}(\lambda) + \sum_{h=1}^{\nu} \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_{2\text{-}h}}(\lambda) + \epsilon$, where $\mathcal{E}_{2\text{-}h}(\cdot) := \mathcal{E}_2(h, \cdot)$ and ν is the maximum number of $\mathcal{A}$'s key queries and $\epsilon := (6\nu + 5)/q$. This is obtained in the same manner as the weakly attribute-hiding security of the OT10 IPE in the full version of [25]: Since the difference between our IPE and the OT10 IPE is only the dimension of the hidden subspaces, i.e., the former has $2n$ and the latter has n, the weakly attribute-hiding security of the OT10 IPE implies the security with $t = 0$ of our IPE.

As for the conditional probability with $t = 1$, i.e., $\Pr[\mathcal{A} \text{ wins} \mid t = 1]$, Lemma 15 holds. Therefore, $\mathsf{Adv}^{\mathsf{ZIPE,AH}}_{\mathcal{A}}(\lambda) = 2 \cdot \mathsf{Adv}^{\mathsf{ZIPE,AH,0'}}_{\mathcal{A}}(\lambda) = \Pr[\mathcal{A} \text{ wins} \mid t = 0] + \Pr[\mathcal{A} \text{ wins} \mid t = 1] - 1 = (\Pr[\mathcal{A} \text{ wins} \mid t = 0] - 1/2) + (\Pr[\mathcal{A} \text{ wins} \mid t = 1] - 1/2) \leq \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_{0\text{-}1}}(\lambda) + \sum_{h=1}^{\nu} \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_{0\text{-}2\text{-}h}}(\lambda) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_{1\text{-}1}}(\lambda) + \sum_{h=1}^{\nu} \Big(\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_{1\text{-}2\text{-}h\text{-}1}}(\lambda) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_{1\text{-}2\text{-}h\text{-}2}}(\lambda)\Big) + \epsilon$, where $\epsilon := (29\nu + 17)/q$. □

Lemma 15 *For any machine $\mathcal{A}$, there exist probabilistic machines $\mathcal{E}_1, \mathcal{E}_{2\text{-}1}$ and $\mathcal{E}_{2\text{-}2}$, whose running times are essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, in Game 0' (described in the proof of Theorem 5), $\Pr[\mathcal{A} \text{ wins} \mid t = 1] - \frac{1}{2} \leq \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_1}(\lambda) + \sum_{h=1}^{\nu} \Big(\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_{2\text{-}h\text{-}1}}(\lambda) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_{2\text{-}h\text{-}2}}(\lambda)\Big) + \epsilon$, where $\mathcal{E}_{2\text{-}h\text{-}1}(\cdot) := \mathcal{E}_{2\text{-}1}(h, \cdot)$, $\mathcal{E}_{2\text{-}h\text{-}2}(\cdot) := \mathcal{E}_{2\text{-}2}(h, \cdot)$, ν is the maximum number of $\mathcal{A}$'s key queries and $\epsilon := (23\nu + 12)/q$.*

Proof To prove Lemma 15, we consider the following $4\nu + 3$ games when $t = 1$. In Game 0', a part framed by a box indicates coefficients to be changed in a subsequent game. In the other games, a part framed by a box indicates coefficients which were changed in a game from the previous game.

Game 0' Same as Game 0 except that flip a coin $t \xleftarrow{\mathsf{U}} \{0, 1\}$ before setup, and the game is aborted in the challenge step if $t \neq s$. In order to prove Lemma 15, we consider the case with $t = 1$.

The reply to a key query for $\vec{v}$ is:

$$\boldsymbol{k}^* :=(\, 1,\ \delta\vec{v},\ \boxed{0^n},\ \boxed{0^n},\ \varphi\vec{v},\ 0^n\,)_{\mathbb{B}^*},$$

where $\delta, \varphi \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q$. The challenge ciphertext for challenge plaintext $m :=m^{(0)} = m^{(1)}$ and vectors $(\vec{x}^{(0)}, \vec{x}^{(1)})$ is:

$$\boldsymbol{c} :=(\, \zeta,\ \boxed{\omega\vec{x}^{(b)}},\ \boxed{0^n},\ \boxed{0^n},\ 0^n,\ \vec{\eta}\,)_{\mathbb{B}},\quad c_3 :=g_T^{\zeta} m,$$

where $b \overset{\mathsf{U}}{\leftarrow} \{0, 1\}$ and $\zeta, \omega \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q$ and $\vec{\eta} \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q^n$. Here, we note that c_3 is independent from bit b.

Game 1 Game 1 is the same as Game 0' except that $\boldsymbol{c}_1$ of the challenge ciphertext for (challenge plaintext $m :=m^{(0)} = m^{(1)}$ and) vectors $(\vec{x}^{(0)}, \vec{x}^{(1)})$ is:

$$\boldsymbol{c}_1 :=(\, \zeta,\ \omega\vec{x}^{(b)},\ \boxed{\omega'\vec{x}^{(b)}},\ 0^n,\ 0^n,\ \vec{\eta}\,)_{\mathbb{B}},$$

where $\omega' \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q$ and all the other variables are generated as in Game 0'.

Game 2-*h*-1(*h*= 1, . . . , *ν*) Game 2-0-4 is Game 1. Game 2-h-1 is the same as Game 2-$(h-1)$-4 except that $\boldsymbol{c}_1$ of the challenge ciphertext for (challenge plaintext $m :=m^{(0)} = m^{(1)}$ and) vectors $(\vec{x}^{(0)}, \vec{x}^{(1)})$ is:

$$\boldsymbol{c}_1 :=(\, \zeta,\ \omega\vec{x}^{(b)},\ \boxed{\omega'\vec{x}^{(b)}},\ \boxed{\omega''_0\vec{x}^{(0)} + \omega''_1\vec{x}^{(1)}},\ 0^n,\ \vec{\eta}\,)_{\mathbb{B}},$$

where $\omega', \omega''_0, \omega''_1 \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q$ and all the other variables are generated as in Game 2-$(h-1)$-4.

Game 2-*h*-2(*h*= 1, . . . , *ν*) Game 2-h-2 is the same as Game 2-h-1 except that the reply to the h-th key query for $\vec{v}$ is:

$$\boldsymbol{k}^* :=(\, 1,\ \sigma\vec{v},\ \boxed{\sigma'\vec{v}},\ 0^n,\ \varphi\vec{v},\ 0^n\,)_{\mathbb{B}^*},$$

where $\sigma' \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q$ and all the other variables are generated as in Game 2-h-1.

Game 2-*h*-3(*h*= 1, . . . , *ν*) Game 2-h-3 is the same as Game 2-h-2 except that $\boldsymbol{c}_1$ of the challenge ciphertext for (challenge plaintexts $m :=m^{(0)} = m^{(1)}$ and) vectors $(\vec{x}^{(0)}, \vec{x}^{(1)})$ is:

$$\boldsymbol{c}_1 :=(\, \zeta,\ \omega\vec{x}^{(b)},\ \boxed{\omega'_0\vec{x}^{(0)} + \omega'_1\vec{x}^{(1)}},\ \omega''_0\vec{x}^{(0)} + \omega''_1\vec{x}^{(1)},\ 0^n,\ \vec{\eta}\,)_{\mathbb{B}},$$

where $\omega'_0, \omega'_1 \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q$ and all the other variables are generated as in Game 2-h-2.

Game 2-*h*-4(*h*= 1, . . . , *ν*) Game 2-h-4 is the same as Game 2-h-3 except that the reply to the h-th key query for $\vec{v}$ is:

$$\boldsymbol{k}^* :=(\, 1,\ \sigma\vec{v},\ \boxed{0^n},\ \boxed{\sigma''\vec{v}},\ \varphi\vec{v},\ 0^n\,)_{\mathbb{B}^*},$$

where $\sigma'' \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q$ and all the other variables are generated as in Game 2-h-3.

Game 3 Game 3 is the same as Game 2-ν-4 except that $\boldsymbol{c}_1$ of the challenge ciphertext for (challenge plaintexts $m :=m^{(0)} = m^{(1)}$ and) vectors $(\vec{x}^{(0)}, \vec{x}^{(1)})$ is:

$$\boldsymbol{c}_1 :=(\, \zeta,\ \boxed{\omega_0\vec{x}^{(0)} + \omega_1\vec{x}^{(1)}},\ \omega'_0\vec{x}^{(0)} + \omega'_1\vec{x}^{(1)},\ \omega''_0\vec{x}^{(0)} + \omega''_1\vec{x}^{(1)},\ 0^n,\ \vec{\eta}\,)_{\mathbb{B}},$$

where $\omega_0, \omega_1 \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q$ and all the other variables are generated as in Game 2-ν-4. Here, we note that $\boldsymbol{c}_1$ is independent from bit $b \overset{\mathsf{U}}{\leftarrow} \{0, 1\}$.

Let $\mathsf{Adv}_{\mathcal{A}}^{(0')}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}h\text{-}1)}(\lambda), \ldots, \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}h\text{-}4)}(\lambda)$ and $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)$ be the advantage of $\mathcal{A}$ in Game $0'$, 1, 2-h-$1, \ldots, 2$-h-4 and 3 when $t = 1$, respectively. $\mathsf{Adv}_{\mathcal{A}}^{(0')}(\lambda)$ is equivalent to the left-hand side of Eq. (15) and $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda) = 0$.

We can evaluate the gaps between pairs of neighboring games, $\mathsf{Adv}_{\mathcal{A}}^{(0')}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda), \ldots,$ $\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}\nu\text{-}4)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)$, similarly to [27]. This completes the proof of Lemma 15. □

11 Comparison

Table 1 compares the proposed ZIPE and NIPE schemes (ZIPE with short ciphertexts in Sect. 8, NIPE with short ciphertexts in Sect. 6, ZIPE with short secret-keys in Sect. 9, NIPE with short secret-keys in Sect. 7, and fully-attribute-hiding ZIPE with short secret-keys in Sect. 10) with the ZIPE and NIPE schemes in [4] that are secure under standard assumptions.

12 Hierarchical ZIPE scheme with constant-size ciphertexts

The proposed hierarchical ZIPE (HIPE) scheme with short ciphertexts is constructed by using two vector spaces, 5-dimensional $\mathbb{V}_0$ and $4n$-dimensional $\mathbb{V}_1$, where hierarchical vector $(\vec{v}_1, \ldots, \vec{v}_\ell)$ (resp. $(\vec{x}_1, \ldots, \vec{x}_{\ell'})$) of secret-key (resp. ciphertext) is embedded in an element in $\mathbb{V}_1$. The delegation mechanism is based on the payload hiding HIPE scheme given in Appendix H.3 in the full version of [25].

12.1 Dual orthonormal basis generator

We describe random dual orthonormal basis generator $\mathcal{G}_{\mathsf{ob}}^{\mathsf{HIPE,CT}}$ below, which is used as a subroutine in the proposed hierarchical ZIPE scheme.

$$
\begin{aligned}
&\mathcal{G}_{\mathsf{ob}}^{\mathsf{HIPE,CT}}(1^\lambda, 4, \vec{n} := (d; n_1, \ldots, n_d)) :\ n := \textstyle\sum_{t=1}^{d} n_t,\\
&\quad \mathsf{param}_{\mathbb{G}} := (q, \mathbb{G}, \mathbb{G}_T, G, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{bpg}}(1^\lambda), \quad N_0 := 5,\ N_1 := 4n,\\
&\quad \mathsf{param}_{\mathbb{V}_t} := (q, \mathbb{V}_t, \mathbb{G}_T, \mathbb{A}_t, e) := \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, N_t, \mathsf{param}_{\mathbb{G}}) \ \text{ for } t = 0, 1,\\
&\quad \psi \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times,\ g_T := e(G, G)^\psi,\ \mathsf{param}_{\vec{n}} := (\vec{n},\ \{\mathsf{param}_{\mathbb{V}_t}\}_{t=0,1},\ g_T),\\
&\quad X_0 := (\chi_{0,i,j})_{i,j=1,\ldots,5} \xleftarrow{\mathsf{U}} GL(N_0, \mathbb{F}_q),\ X_1 \xleftarrow{\mathsf{U}} \widetilde{\mathcal{L}}(4, n, \mathbb{F}_q),\ \text{hereafter},\\
&\quad \{\mu_{i,j}, \mu'_{i,j,l}\}_{i,j=1,\ldots 4; l=1,\ldots,n} \text{ denotes non-zero entries of } X_1 \text{ as in Eq. (4)},\\
&\quad \boldsymbol{b}_{0,i} := (\chi_{0,i,1}, \ldots, \chi_{0,i,5})_{\mathbb{A}} = \textstyle\sum_{j=1}^{5} \chi_{0,i,j} \boldsymbol{a}_j \text{ for } i = 1, \ldots, 5,\ \mathbb{B}_0 := (\boldsymbol{b}_{0,1}, \ldots, \boldsymbol{b}_{0,5}),\\
&\quad B_{i,j} := \mu_{i,j} G,\ B'_{i,j,l} := \mu'_{i,j,l} G \ \text{ for } i, j = 1, \ldots, 4; l = 1, \ldots, n,\\
&\quad \text{for } t = 0, 1,\ (\vartheta_{t,i,j})_{i,j=1,\ldots,N_t} := \psi \cdot (X_t^{\mathrm{T}})^{-1},\\
&\quad \boldsymbol{b}^*_{t,i} := (\vartheta_{t,i,1}, \ldots, \vartheta_{t,i,N_t})_{\mathbb{A}} = \textstyle\sum_{j=1}^{N_t} \vartheta_{t,i,j} \boldsymbol{a}_j \text{ for } i = 1, \ldots, N_t,\ \mathbb{B}^*_t := (\boldsymbol{b}^*_{t,1}, \ldots, \boldsymbol{b}^*_{t,N_t}),\\
&\quad \text{return } (\mathsf{param}_{\vec{n}}, \mathbb{B}_0, \mathbb{B}^*_0, \{B_{i,j}, B'_{i,j,l}\}_{i,j=1,\ldots,4; l=1,\ldots,n}, \mathbb{B}^*_1).
\end{aligned}
$$

Table 1 Comparison with IPE schemes in [4], where $|\mathbb{G}|$, $|\mathbb{G}_T|$, $|\mathbb{F}_q|$, P and M represent size of an element of $\mathbb{G}$, that of $\mathbb{G}_T$, that of $\mathbb{F}_q$, pairing operation, and scalar multiplication on $\mathbb{G}$, respectively

	AL10 [4] ZIPE with short CT	AL10 [4] NIPE with short CT	Proposed ZIPE with short CT	Proposed NIPE with short CT	Proposed ZIPE with short SK	Proposed NIPE with short SK	Proposed fully-AH ZIPE with short SK
Security	Adaptive PH	Co-selective PH	Adaptive PH	Adaptive PH	Adaptive weakly-AH	Adaptive PH	Adaptive fully-AH
Assump.	DLIN and DBDH	DLIN and DBDH	DLIN	DLIN	DLIN	DLIN	DLIN
IP rel.	Zero	Non-zero	Zero	Non-zero	Zero	Non-zero	Zero
PK size	$(n+11)\vert\mathbb{G}\vert+\vert\mathbb{G}_T\vert$	$(n+11)\vert\mathbb{G}\vert+\vert\mathbb{G}_T\vert$	$(10n+13)\vert\mathbb{G}\vert+\vert\mathbb{G}_T\vert$	$(8n+23)\vert\mathbb{G}\vert+\vert\mathbb{G}_T\vert$	$(10n+13)\vert\mathbb{G}\vert+\vert\mathbb{G}_T\vert$	$(8n+23)\vert\mathbb{G}\vert+\vert\mathbb{G}_T\vert$	$(12n+16)\vert\mathbb{G}\vert+\vert\mathbb{G}_T\vert$
SK size	$(n+6)\vert\mathbb{G}\vert+(n-1)\vert\mathbb{F}_q\vert$	$(n+6)\vert\mathbb{G}\vert$	$(4n+1)\vert\mathbb{G}\vert$	$(4n+5)\vert\mathbb{G}\vert$	$9\vert\mathbb{G}\vert$	$13\vert\mathbb{G}\vert$	$11\vert\mathbb{G}\vert$
CT size	$9\vert\mathbb{G}\vert+\vert\mathbb{G}_T\vert+\vert\mathbb{F}_q\vert$	$9\vert\mathbb{G}\vert+\vert\mathbb{G}_T\vert$	$9\vert\mathbb{G}\vert+\vert\mathbb{G}_T\vert$	$13\vert\mathbb{G}\vert+\vert\mathbb{G}_T\vert$	$(4n+1)\vert\mathbb{G}\vert+\vert\mathbb{G}_T\vert$	$(4n+5)\vert\mathbb{G}\vert+\vert\mathbb{G}_T\vert$	$(5n+1)\vert\mathbb{G}\vert+\vert\mathbb{G}_T\vert$
Dec time	$9\text{P}+n\text{M}$	$9\text{P}+n\text{M}$	$9\text{P}+4(n-1)\text{M}$	$13\text{P}+4(n-1)\text{M}$	$9\text{P}+4(n-1)\text{M}$	$13\text{P}+4(n-1)\text{M}$	$11\text{P}+5(n-1)\text{M}$

CT, SK, PH, AH, IP and DBDH stand for ciphertexts, secret-keys, payload-hiding, attribute-hiding, inner-product and decisional bilinear Diffie–Hellman, respectively

Remark 14 Let

$$\begin{pmatrix} \boldsymbol{b}_{1,(i-1)n+1} \\ \vdots \\ \boldsymbol{b}_{1,in} \end{pmatrix} :=\begin{pmatrix} B'_{i,1,1} & & & & & B'_{i,4,1} & & & \\ B'_{i,1,2} & B_{i,1} & & & \cdots & B'_{i,4,2} & B_{i,4} & & \\ \vdots & & \ddots & & & \vdots & & \ddots & \\ B'_{i,1,n} & & & B_{i,1} & & B'_{i,4,n} & & & B_{i,4} \end{pmatrix}$$

$$\text{for } i = 1, \dots, 4,$$

$$\mathbb{B}_1 :=(\boldsymbol{b}_{1,1}, \dots, \boldsymbol{b}_{1,4n}),$$

where a blank element in the matrix denotes $0 \in \mathbb{G}$. $\mathbb{B}_1$ is the dual orthonormal basis of $\mathbb{B}_1^*$, i.e., $e(\boldsymbol{b}_{1,i}, \boldsymbol{b}^*_{1,i}) = g_T$ and $e(\boldsymbol{b}_{1,i}, \boldsymbol{b}^*_{1,j}) = 1$ for $1 \le i \ne j \le 4n$.

12.2 Construction and security

In the description of the scheme, we assume that input vector, $\vec{x}_t :=(x_{t,1}, \dots, x_{t,n_t})$, has an index $(t, l) \ne (1, 1)$ with $x_{t,l} \ne 0$, and that level-1 input vector, $\vec{v}_1 :=(v_{1,1}, \dots, v_{1,n_1})$, satisfies $v_{1,1} \ne 0$. The plaintext space is $\mathbb{G}_T$.

```
Setup$(1^\lambda,\ \vec{n} :=(d; n_1, \dots, n_d))$ : $n := \sum_{t=1}^{d} n_t$,
  $(\mathsf{param}_{\vec{n}}, \mathbb{B}_0, \mathbb{B}_0^*, \{B_{i,j}, B'_{i,j,l}\}_{i,j=1,\dots,4;l=1,\dots,n}, \mathbb{B}_1^*) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}^{\mathsf{HIPE,CT}}(1^\lambda, 4, \vec{n})$,
 $\widehat{\mathbb{B}}_0 :=(\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5})$, $\widehat{\mathbb{B}}_0^* :=(\boldsymbol{b}^*_{0,1}, \boldsymbol{b}^*_{0,4})$, $\widehat{\mathbb{B}}_1^* :=(\boldsymbol{b}^*_{1,1}, \dots, \boldsymbol{b}^*_{1,n}, \boldsymbol{b}^*_{1,2n+1}, \dots, \boldsymbol{b}^*_{1,3n})$,
 return pk $:=(1^\lambda, \mathsf{param}_{\vec{n}}, \widehat{\mathbb{B}}_0, \{B_{i,j}, B'_{i,j,l}\}_{i=1,4;j=1,\dots,4;l=1,\dots,n}, \{\widehat{\mathbb{B}}_t^*\}_{t=0,1})$, sk $:=\boldsymbol{b}^*_{0,3}$.
KeyGen(pk, sk, $(\vec{v}_1, \dots, \vec{v}_\ell) \in \mathbb{F}_q^{n_1} \times \cdots \times \mathbb{F}_q^{n_\ell}$) :
  $s_t, \theta_t, \varphi_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q$ for $t = 1, \dots, \ell$, $s_0 := \sum_{t=1}^{\ell} s_t$, $\vec{\varphi}_1 \xleftarrow{\mathsf{U}} \mathbb{F}_q^n$,
  $\boldsymbol{k}^*_{\ell,0} :=(\ -s_0,\ 0,\ 1,\ \varphi_0,\ 0\ )_{\mathbb{B}_0^*}$,
  $\boldsymbol{k}^*_{\ell,1} :=(\ \overbrace{s_1\vec{e}_{1,1} + \theta_1\vec{v}_1,\ \dots,\ s_\ell\vec{e}_{\ell,1} + \theta_\ell\vec{v}_\ell,\ 0^{n_{\ell+1}},\ \dots,\ 0^{n_d},}^{n}\ \ 0^n,\ \ \vec{\varphi}_1,\ \ 0^n\ \ )_{\mathbb{B}_1^*}$,
  return sk$_\ell$ $:=((\vec{v}_1, \dots, \vec{v}_\ell), \boldsymbol{k}^*_{\ell,0}, \boldsymbol{k}^*_{\ell,1})$.
Enc(pk, $m \in \mathbb{G}_T, (\vec{x}_1, \dots, \vec{x}_\ell) \in \mathbb{F}_q^{n_1} \times \cdots \times \mathbb{F}_q^{n_\ell}$) :
  $\omega, \eta_0, \eta_1 \xleftarrow{\mathsf{U}} \mathbb{F}_q, \boldsymbol{c}_0 :=(\omega, 0, \zeta, 0, \eta_0)_{\mathbb{B}_0}$, $\vec{x} :=(x_l)_{l=1,\dots,n} :=(\vec{x}_1, \dots, \vec{x}_\ell, 0^{n_{\ell+1}}, \dots,$
  $0^{n_d}) \in \mathbb{F}_q^n$,
  $C_{1,j} :=\omega B_{1,j} + \eta_1 B_{4,j}$, $C_{2,j} := \sum_{l=1}^{n} x_l(\omega B'_{1,j,l} + \eta_1 B'_{4,j,l})$ for $j = 1, \dots, 4$,
  $c_3 :=g_T^{\zeta} m$, return ct $:=((\vec{x}_1, \dots, \vec{x}_\ell), \boldsymbol{c}_0, \{C_{1,j}, C_{2,j}\}_{j=1,\dots,4}, c_3)$.
Dec(pk, sk$_\ell$ $:=((\vec{v}_1, \dots, \vec{v}_\ell), \boldsymbol{k}^*_{\ell,0}, \boldsymbol{k}^*_{\ell,1})$, ct $:=((\vec{x}_1, \dots, \vec{x}_{\ell'}), \boldsymbol{c}_0, \{C_{1,j}, C_{2,j}\}_{j=1,\dots,4}, c_3)$):
  if $\ell \le \ell'$,   parse $\boldsymbol{k}^*_1$ as a $4n$-tuple $(K^*_1, \dots, K^*_{4n}) \in \mathbb{G}^{4n}$,
   $\vec{x} :=(x_1, \dots, x_n) :=(\vec{x}_1, \dots, \vec{x}_{\ell'}, 0^{n_{\ell'+1}}, \dots, 0^{n_d}) \in \mathbb{F}_q^n$,
   $D^*_j := \sum_{l=2}^{n} x_l\, K^*_{(j-1)n+l}$ for $j = 1, \dots, 4$,
   $F :=e(\boldsymbol{c}_0, \boldsymbol{k}^*_0) \cdot \prod_{j=1}^{4} \left(e(C_{1,j}, D^*_j) \cdot e(C_{2,j}, K^*_{(j-1)n+1})\right)$,  return $m' :=c_3/F$,
  else,  return $\perp$.
Delegate$_\ell$(pk, sk$_\ell$, $\vec{v}_{\ell+1}$) :
```

$$s_{\mathsf{del},t}, \theta_{\mathsf{del},t}, \varphi_{\mathsf{del},0} \xleftarrow{\mathsf{U}} \mathbb{F}_q \ \text{ for } t = 1, \ldots, \ell + 1, \ \ s_{\mathsf{del},0} := \textstyle\sum_{t=1}^{\ell+1} s_{\mathsf{del},t}, \ \ \vec{\varphi}_{\mathsf{del},1} \xleftarrow{\mathsf{U}} \mathbb{F}_q^n,$$

$$\boldsymbol{k}^*_{\mathsf{del},0} := (\ -s_{\mathsf{del},0},\ 0,\ 0,\ \varphi_{\mathsf{del},0},\ 0\ )_{\mathbb{B}_0^*},$$

$$\boldsymbol{k}^*_{\mathsf{del},1} := (\overbrace{\ s_{\mathsf{del},1}\vec{e}_{1,1} + \theta_1\vec{v}_1,\ \ldots,\ s_{\mathsf{del},\ell+1}\vec{e}_{\ell+1,1} + \theta_{\mathsf{del},\ell+1}\vec{v}_{\ell+1},\ 0^{n_{\ell+2}},\ \ldots,\ 0^{n_d}}^{n}, \\ 0^n,\ \ \vec{\varphi}_{\mathsf{del},1},\ \ 0^n\ )_{\mathbb{B}_1^*},$$

$$\boldsymbol{k}^*_{\ell+1,\iota} := \boldsymbol{k}^*_{\ell,\iota} + \boldsymbol{k}^*_{\mathsf{del},\iota} \ \text{ for } \iota = 0, 1,$$

return $\mathsf{sk}_{\ell+1} := ((\vec{v}_1, \ldots, \vec{v}_{\ell+1}), \boldsymbol{k}^*_{\ell+1,0}, \boldsymbol{k}^*_{\ell+1,1})$.

Remark 15 A part of output of $\mathsf{Setup}(1^\lambda, \vec{n})$, $\{B_{i,j},\ B'_{i,j,l}\}_{i=1,4;\,j=1,\ldots,4;\,l=1,\ldots,n}$, can be identified with $\widehat{\mathbb{B}}_1 := (\boldsymbol{b}_{1,1}, \ldots, \boldsymbol{b}_{1,n}, \boldsymbol{b}_{1,3n+1}, \ldots, \boldsymbol{b}_{1,4n})$ through the form of Eq. (6), while $\mathbb{B}_1 := (\boldsymbol{b}_{1,1}, \ldots, \boldsymbol{b}_{1,4n})$ is identified with $\{B_{i,j},\ B'_{i,j,l}\}_{i,j=1,\ldots,4;\ l=1,\ldots,n}$ by Eq. (6). Decryption Dec can be alternatively described as:

$\mathsf{Dec}'(\mathsf{pk}, \mathsf{sk}_\ell := ((\vec{v}_1, \ldots, \vec{v}_\ell), \boldsymbol{k}^*_{\ell,0}, \boldsymbol{k}^*_{\ell,1}), \mathsf{ct} := ((\vec{x}_1, \ldots, \vec{x}_{\ell'}), \boldsymbol{c}_0, \{C_{1,j}, C_{2,j}\}_{j=1,\ldots,4}, c_3))$:

$\vec{x} := (x_1, \ldots, x_n) := (\vec{x}_1, \ldots, \vec{x}_{\ell'}, 0^{n_{\ell'+1}}, \ldots, 0^{n_d}) \in \mathbb{F}_q^n$,

$$\boldsymbol{c}_1 := \big(\ \overbrace{C_{2,1}, x_2C_{1,1}, \ldots, x_nC_{1,1}}^{n},\ \ \ldots,\ \ \overbrace{C_{2,4}, x_2C_{1,4}, \ldots, x_nC_{1,4}}^{n}\ \big),$$

$$\text{that is, } \boldsymbol{c}_1 = (\ \overbrace{\omega\vec{x},}^{n}\ \ \overbrace{0^n,}^{n}\ \ \overbrace{0^n,}^{n}\ \ \overbrace{\eta_1\vec{x}}^{n}\ )_{\mathbb{B}_1},\ \ F := e(\boldsymbol{c}_0, \boldsymbol{k}^*_0) \cdot e(\boldsymbol{c}_1, \boldsymbol{k}^*_1),$$

return $m' := c_3/F$.

[Correctness] Using the alternate decryption Dec', $F = e(\boldsymbol{c}_0, \boldsymbol{k}^*_0) \cdot e(\boldsymbol{c}_1, \boldsymbol{k}^*_1) = g_T^{-\omega s_0+\zeta} g_T^{\omega \sum_{t=1}^{\ell} s_t} = g_T^{\zeta}$ if $\ell \le \ell'$ and $\vec{x}_t \cdot \vec{v}_t = 0$ for $t = 1, \ldots, \ell$.

The definition of *adaptively payload-hiding* security and the advantage $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{HIPE,PH}}(\lambda)$ of adversary $\mathcal{A}$ can be obtained through a straightforward extension of that of HIBE, e.g., [15], with replacing ID-matching by vector-orthogonality.

Theorem 6 *The proposed HIPE scheme is adaptively payload-hiding against chosen plaintext attacks under the DLIN assumption.*

For any machine $\mathcal{A}$, there exist probabilistic machines $\mathcal{E}_1$ and $\mathcal{E}_2$, whose running times are essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{HIPE,PH}}(\lambda) \le \mathsf{Adv}_{\mathcal{E}_1}^{\mathsf{DLIN}}(\lambda) + \sum_{h=1}^{\nu} \mathsf{Adv}_{\mathcal{E}_{2\text{-}h}}^{\mathsf{DLIN}}(\lambda) + \epsilon$, where $\mathcal{E}_{2\text{-}h}(\cdot) := \mathcal{E}_2(h, \cdot)$, ν is the maximum number of adversary $\mathcal{A}$'s key queries, and $\epsilon = (11\nu + 6)/q$.

Theorem 6 is proven similarly to Theorem 3.

13 Concluding remarks

The technique with using special type matrices shown in this paper can reduce the size of ciphertexts or secret-keys of adaptively secure FE schemes in [25] from $O(dn)$ to $O(d)$, where d is the number of sub-universes of attributes, and n is the maximal length of attribute vectors. A key-policy attribute-based encryption (ABE) system with constant-size ciphertext [5] is selectively secure in the standard model. Therefore, it is an interesting open problem to realize an *adaptively secure and constant-size ciphertext* ABE scheme.

Acknowledgments The authors would like to thank Sherman S.M. Chow for his invaluable comments and suggestions on our preliminary manuscript. We also appreciate anonymous reviewers of CANS 2011 for their valuable comments.

Appendix: Proofs of Lemmas

Proofs of Lemmas 2 and 3 in Sect. 5

For a positive integer x, let $[x] :=\{1, \dots, x\}$.
Lemma 2 $\mathcal{L}(w, n, \mathbb{F}_q)$ and $\widetilde{\mathcal{L}}(w, n, \mathbb{F}_q)$ are subgroups of $GL(wn, \mathbb{F}_q)$.

Proof Below, we will show that $\mathcal{L}(w, n, \mathbb{F}_q)$ is a subgroup of $GL(wn, \mathbb{F}_q)$. For $\widetilde{\mathcal{L}}(w, n, \mathbb{F}_q)$, the lemma is proven in the same manner as for $\mathcal{L}(w, n, \mathbb{F}_q)$.

Based on the block partition on $X \in \mathbb{F}_q^{wn\times wn}$ with submatrices $X_{i,j} \in \mathbb{F}_q^{n\times n}$, i.e., $X :=(X_{i,j})_{i,j\in[w]} := \begin{pmatrix} X_{1,1} & \cdots & X_{1,w} \\ \vdots & & \vdots \\ X_{w,1} & \cdots & X_{w,w} \end{pmatrix}$, we will define a permutation matrix Π. Since $X_{i,j} \in \mathbb{F}_q^{n\times n}$, each row of X is indexed by a pair (i, k) with $i \in [w]$; $k \in [n]$, which is corresponding to the $((i-1)n + k)$-th row. The swapping of the index pair $(i, k) \mapsto (k, i)$ leads to a permutation π on the set $[wn]$ as,

$$\begin{array}{cccc} \pi : & [wn] & \rightarrow & [wn] \\ & \cup\!\!\!| & & \cup\!\!\!| \\ & (i-1)n + k & \mapsto & (k-1)w + i \end{array} \tag{13}$$

with $i \in [w]$; $k \in [n]$. We denote the corresponding permutation matrix by Π, i.e., the left multiplication by Π is equivalent to the permutation π on rows (of X). $\Pi^{-1} = \Pi^{\mathrm{T}}$ since Π is a permutation matrix, and we see that the right multiplication by Π^{-1} is equivalent to the permutation π on columns (of X).

Let the conjugate set $\mathcal{P}(w, n, \mathbb{F}_q) :=\Pi \cdot \mathcal{L}(w, n, \mathbb{F}_q) \cdot \Pi^{-1}$. Since the rows and columns are permuted by π, for $X :=(X_{i,j})_{i,j\in[w]} \in \mathcal{L}(w, n, \mathbb{F}_q)$ with $X_{i,j} := \begin{pmatrix} \mu_{i,j} & & \mu'_{i,j,1} \\ & \ddots & \vdots \\ & \mu_{i,j} & \mu'_{i,j,n-1} \\ & & \mu'_{i,j,n} \end{pmatrix}$, $Y :=\Pi \cdot X \cdot \Pi^{-1}$ is given as $Y = \begin{pmatrix} Y_0 & & Y_1 \\ & \ddots & \vdots \\ & Y_0 & Y_{n-1} \\ & & Y_n \end{pmatrix}$, where $Y_0 := \begin{pmatrix} \mu_{1,1} & \cdots & \mu_{1,w} \\ \vdots & & \vdots \\ \mu_{w,1} & \cdots & \mu_{w,w} \end{pmatrix}$ and $Y_i := \begin{pmatrix} \mu'_{1,1,i} & \cdots & \mu'_{1,w,i} \\ \vdots & & \vdots \\ \mu'_{w,1,i} & \cdots & \mu'_{w,w,i} \end{pmatrix}$. Therefore, since

$\mathcal{L}(w, n, \mathbb{F}_q) \subset GL(wn, \mathbb{F}_q)$,

$$\mathcal{P}(w, n, \mathbb{F}_q) = \left\{ Y := \begin{pmatrix} Y_0 & & & Y_1 \\ & \ddots & & \vdots \\ & & Y_0 & Y_{n-1} \\ & & & Y_n \end{pmatrix} \middle| \begin{array}{l} Y_0, Y_n \in GL(w, \mathbb{F}_q), \\ Y_1, \ldots, Y_{n-1} \in \mathbb{F}_q^{w \times w}, \\ \text{a blank element in the} \\ \text{matrix denotes } 0 \in \mathbb{F}_q \end{array} \right\}. \tag{14}$$

We see that $\mathcal{P}(w, n, \mathbb{F}_q)$ is a subgroup of $GL(wn, \mathbb{F}_q)$. So, $\mathcal{L}(w, n, \mathbb{F}_q) = \Pi^{-1} \cdot \mathcal{P}(w, n, \mathbb{F}_q) \cdot \Pi$ is also a subgroup of $GL(wn, \mathbb{F}_q)$. This completes the proof of Lemma 2. □

Lemma 3 $\mathcal{L}^+(w, n, \mathbb{F}_q)$ is a subgroup of $GL(wn + 1, \mathbb{F}_q)$.

Proof For the proof, we define an injective group homomorphism,

$$\begin{array}{ccc} \iota : GL(wn+1, \mathbb{F}_q) & \hookrightarrow & GL((w+1)n, \mathbb{F}_q) \\ \rotatebox{90}{$\in$} & & \rotatebox{90}{$\in$} \\ X & \mapsto & \begin{pmatrix} I_{n-1} & 0 \\ 0 & X \end{pmatrix}. \end{array}$$

We will show the following claim.

Claim 1 $\iota(\mathcal{L}^+(w, n, \mathbb{F}_q)) = \mathcal{L}(w+1, n, \mathbb{F}_q) \cap \iota(GL((w+1)n, \mathbb{F}_q))$.
This equality is on the bottom-right corner of the following diagram,

$$\begin{array}{cccc} \iota : GL(wn+1, \mathbb{F}_q) & \hookrightarrow & GL((w+1)n, \mathbb{F}_q) & \\ \cup & & \cup & \\ \mathcal{L}^+(w, n, \mathbb{F}_q) & \cong & \iota(\mathcal{L}^+(w, n, \mathbb{F}_q)) & = \mathcal{L}(w+1, n, \mathbb{F}_q) \cap \iota(GL((w+1)n, \mathbb{F}_q)). \end{array}$$

Proof of Claim 1 Since $X \in \mathcal{L}(w+1, n, \mathbb{F}_q) \cap \iota(GL((w+1)n, \mathbb{F}_q))$ is given as $(X_{i,j})_{i,j \in [w+1]} := \begin{pmatrix} X_{1,1} & \cdots & X_{1,w+1} \\ \vdots & & \vdots \\ X_{w+1,1} & \cdots & X_{w+1,w+1} \end{pmatrix}$, $X_{1,1} = \begin{pmatrix} 1 & & & \\ & \ddots & & \\ & & 1 & \\ & & & \mu'_{1,1,n} \end{pmatrix}$, $X_{i,1} = \begin{pmatrix} & \mu'_{i,1,1} \\ & \vdots \\ & \\ & \mu'_{i,1,n} \end{pmatrix}$ for $i = 2, \ldots, w+1$, and $X_{1,j} = \begin{pmatrix} & & \\ & & \\ & & \mu'_{1,j,n} \end{pmatrix}$ for $j = 2, \ldots, w+1$, where a blank element in the submatrices denotes $0 \in \mathbb{F}_q$. That is,

$$X := \begin{pmatrix} I_{n-1} & & & & \\ & \mu'_{1,1,n} & \mu'_{1,2,n}\vec{e}_n & \cdots & \mu'_{1,w+1,n}\vec{e}_n \\ & \vec{\mu}'^{\mathrm{T}}_{2,1} & X_{2,2} & \cdots & X_{2,w+1} \\ & \vdots & \vdots & & \vdots \\ & \vec{\mu}'^{\mathrm{T}}_{w+1,1} & X_{w+1,2} & \cdots & X_{w+1,w+1} \end{pmatrix},$$

where $\vec{\mu}'_{i,1} := (\mu'_{i,1,1}, \ldots, \mu'_{i,1,n})$. This shows that $\iota(\mathcal{L}^+(w, n, \mathbb{F}_q)) = \mathcal{L}(w+1, n, \mathbb{F}_q) \cap \iota(GL((w+1)n, \mathbb{F}_q))$, i.e., Claim 1 holds. □

Since $\mathcal{L}(w+1, n, \mathbb{F}_q)$ (and $\iota(GL((w+1)n, \mathbb{F}_q))$) are subgroups of $GL((w+1)n, \mathbb{F}_q)$ (Lemma 2), from Claim 1, $\iota(\mathcal{L}^+(w, n, \mathbb{F}_q))$ is a subgroup of $GL((w+1)n, \mathbb{F}_q)$. Therefore, since ι is an injective group homomorphism, $\mathcal{L}^+(w, n, \mathbb{F}_q)$ is also a subgroup of $GL(wn + 1, \mathbb{F}_q)$. This completes the proof of Lemma 3. □

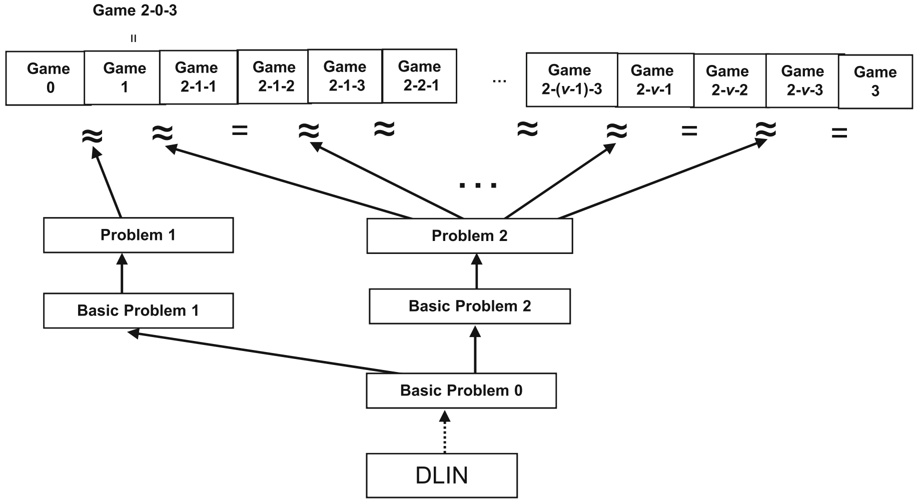

Fig. 1 Structure of reductions for Theorem 1

Proofs of Lemmas 4–12 in Sect. 6

Preliminaries

Figure 1 shows the structure of security reduction for Theorem 1, where the security of the scheme is hierarchically reduced to the intractability of the DLIN problem. Basic Problems 0, 1, 2 are defined below. The reduction steps indicated by arrows will be shown below, and the step given by dotted arrow can be shown in the same manner as that in (the full version of) [25].

For the proofs of Lemmas 4 and 5, we give the following intermediate problem, Basic Problem 0 (Definition 10) and Lemma 16. (In [25], an additional element $\delta\xi G$ is included in an output of Basic Problem 0 for a shorter dimension $3n+1$ than $4n$. Here, it is not necessary.)

Definition 10 (*Basic Problem 0*) Basic Problem 0 is to guess $\beta \in \{0, 1\}$, given $(\mathsf{param}_{\mathsf{BP0}}, \widehat{\mathbb{B}}, \mathbb{B}^*, \boldsymbol{y}^*_\beta, \boldsymbol{f}, \kappa G, \xi G) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{BP0}}_\beta(1^\lambda)$, where

$$\begin{aligned}
\mathcal{G}^{\mathsf{BP0}}_\beta(1^\lambda):\quad & \mathsf{param}_{\mathbb{G}} :=(q, \mathbb{G}, \mathbb{G}_T, G, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{bpg}}(1^\lambda),\\
& \mathsf{param}_{\mathbb{V}} :=(q, \mathbb{V}, \mathbb{G}_T, \mathbb{A}, e) :=\mathcal{G}_{\mathsf{dpvs}}(1^\lambda, 3, \mathsf{param}_{\mathbb{G}}),\\
& X := \begin{pmatrix} \vec{\chi}_1 \\ \vec{\chi}_2 \\ \vec{\chi}_3 \end{pmatrix} :=(\chi_{i,j})_{i,j} \xleftarrow{\mathsf{U}} GL(3, \mathbb{F}_q),\ (\vartheta_{i,j})_{i,j} := \begin{pmatrix} \vec{\vartheta}_1 \\ \vec{\vartheta}_2 \\ \vec{\vartheta}_3 \end{pmatrix} :=(X^{\mathrm{T}})^{-1},\ \ \kappa, \xi \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times,\\
& \boldsymbol{b}_i :=\kappa(\vec{\chi}_i)_{\mathbb{A}} = \kappa \textstyle\sum_{j=1}^3 \chi_{i,j}\boldsymbol{a}_j \ \text{ for } i = 1, 3,\ \ \widehat{\mathbb{B}} :=(\boldsymbol{b}_1, \boldsymbol{b}_3),\\
& \boldsymbol{b}^*_i :=\xi(\vec{\vartheta}_i)_{\mathbb{A}} = \xi \textstyle\sum_{j=1}^3 \vartheta_{i,j}\boldsymbol{a}_{t,j} \ \text{ for } i = 1, 2, 3,\ \ \mathbb{B}^* :=(\boldsymbol{b}^*_1, \boldsymbol{b}^*_2, \boldsymbol{b}^*_3),\\
& g_T :=e(G, G)^{\kappa\xi},\ \ \mathsf{param}_{\mathsf{BP0}} :=(\mathsf{param}_{\mathbb{V}},\ g_T),\ \ \delta, \sigma, \omega \xleftarrow{\mathsf{U}} \mathbb{F}_q,\ \ \rho, \tau \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times,\\
& \boldsymbol{y}^*_0 :=(\delta, 0, \sigma)_{\mathbb{B}^*},\ \ \boldsymbol{y}^*_1 :=(\delta, \rho, \sigma)_{\mathbb{B}^*},\ \ \boldsymbol{f} :=(\omega, \tau, 0)_{\mathbb{B}},\\
& \text{return } (\mathsf{param}_{\mathsf{BP0}}, \widehat{\mathbb{B}}, \mathbb{B}^*, \boldsymbol{y}^*_\beta, \boldsymbol{f}, \kappa G, \xi G).
\end{aligned}$$

for $\beta \xleftarrow{\mathsf{U}} \{0,1\}$. For a probabilistic machine $\mathcal{D}$, we define the advantage of $\mathcal{D}$ for Basic Problem 0, $\mathsf{Adv}_{\mathcal{D}}^{\mathsf{BP0}}(\lambda)$, is similarly defined as in Definition 8.

Lemma 16 *For any machine $\mathcal{D}$, there is a probabilistic machine $\mathcal{E}$, whose running time is essentially the same as that of $\mathcal{E}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{D}}^{\mathsf{BP0}}(\lambda) \le \mathsf{Adv}_{\mathcal{E}}^{\mathsf{DLIN}}(\lambda) + 5/q$.*

Proof We note that dual bases $(\mathbb{B}, \mathbb{B}^*)$ in Basic Problem 0 are generated by a general linear matrix $X \xleftarrow{\mathsf{U}} GL(3, \mathbb{F}_q)$, so Lemma 16 is proven in a similar manner to the security proof of Basic Problem 0 in [25]. □

The following Remark 16 is for the proofs of Lemmas of 17 and 19.

Remark 16 For matrix $W := (\chi_{i,j})_{i,j=1,\ldots,N} \in \mathbb{F}_q^{N\times N}$ and element $\boldsymbol{v}$ in N-dimensional $\mathbb{V}$, $W(\boldsymbol{v})$ denotes $\sum_{i=1,j=1}^{N,N} \chi_{i,j}\phi_{i,j}(\boldsymbol{v})$ using canonical maps $\{\phi_{i,j}\}$ (Definition 2). Similarly, for matrix $(\vartheta_{i,j}) \,:\, =(W^{-1})^{\mathrm{T}}$, $(W^{-1})^{\mathrm{T}}(\boldsymbol{v}) : = \sum_{i=1,j=1}^{N,N} \vartheta_{i,j}\phi_{i,j}(\boldsymbol{v})$. It holds that $e(W(\boldsymbol{x}), (W^{-1})^{\mathrm{T}}(\boldsymbol{y})) = e(\boldsymbol{x}, \boldsymbol{y})$ for any $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{V}$.

Proof of Lemma 4

Lemma 4 *For any machine $\mathcal{B}$, there exists a probabilistic machine $\mathcal{E}$, whose running times are essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{P1}}(\lambda) \le \mathsf{Adv}_{\mathcal{E}}^{\mathsf{DLIN}}(\lambda) + 5/q$.*

Proof At the top level, the proof of Lemma 4 is similar to the security proof of Problem 1 in [25]. The main difference is that special form matrices Eq. (3) are used for generating master public and secret keys in our schemes. One key fact for the security reduction is that $\mathcal{L}(4, n, \mathbb{F}_q)$ is a subgroup of $GL(4n, \mathbb{F}_q)$ (Lemma 2).

For the proof of Lemma 4, we give the following intermediate problem, Basic Problems 1 (Definition 11). From Lemmas 16, 17 and 18, we obtain Lemma 4. □

Based on Remark 4, hereafter, we consider the output of $\mathcal{G}_{\beta}^{\mathsf{P1}}(1^{\lambda}, n)$ is expressed as $(\mathsf{param}_n, \mathbb{B}_0, \widehat{\mathbb{B}}_0^*, \boldsymbol{e}_{\beta,0}, \mathbb{B}_1, \widehat{\mathbb{B}}_1^*, \{\boldsymbol{e}_{\beta,1,i}\}_{i=1,\ldots,n})$ and also we give the output of Basic Problem 1 as such a vector form over bases $\{\mathbb{B}_t\}_{t=0,1}$.

Definition 11 (*Basic Problem 1*) Basic Problem 1 is to guess $\beta \in \{0,1\}$, given $(\mathsf{param}_n, \{\mathbb{B}_t, \widehat{\mathbb{B}}_t^*\}_{t=0,1}, \boldsymbol{f}_{\beta,0}, \{\boldsymbol{f}_{\beta,1,i}\}_{i=1,\ldots,n}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\beta}^{\mathsf{BP1}}(1^{\lambda}, n)$, where

$$
\begin{aligned}
&\mathcal{G}_{\beta}^{\mathsf{BP1}}(1^{\lambda}, n): \quad (\mathsf{param}_n, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0,1}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}^{\mathsf{NIPE,CT}}(1^{\lambda}, 4, n),\\
&\quad \widehat{\mathbb{B}}_0^* : = (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \ldots, \boldsymbol{b}_{0,5}), \quad \widehat{\mathbb{B}}_1^* : = (\boldsymbol{b}_{1,1}, \ldots, \boldsymbol{b}_{1,n}, \boldsymbol{b}_{1,2n+1}, \ldots, \boldsymbol{b}_{1,4n}),\\
&\quad \omega, \gamma_0, \gamma_1 \xleftarrow{\mathsf{U}} \mathbb{F}_q, \quad \tau \xleftarrow{\mathsf{U}} \mathbb{F}_q^{\times}, \quad \boldsymbol{f}_{0,0} : = (\omega, 0, 0, 0, \gamma_0)_{\mathbb{B}_0}, \quad \boldsymbol{f}_{1,0} : = (\omega, \tau, 0, 0, \gamma_0)_{\mathbb{B}_0},\\
&\quad \text{for } i = 1, \ldots, n;\\
&\quad\quad \vec{e}_i : = (0^{i-1}, 1, 0^{n-i}) \in \mathbb{F}_q^n,\\
&\quad\quad \begin{array}{llllll}
\boldsymbol{f}_{0,1,i} : = (& \overbrace{\omega\vec{e}_i,}^{n} & \overbrace{0^n,}^{n} & \overbrace{0^n,}^{n} & \overbrace{\gamma_1\vec{e}_i}^{n} &)_{\mathbb{B}_1},\\
\boldsymbol{f}_{1,1,i} : = (& \omega\vec{e}_i, & \tau\vec{e}_i, & 0^n, & \gamma_1\vec{e}_i &)_{\mathbb{B}_1},
\end{array}\\
&\quad \text{return } (\mathsf{param}_n, \{\mathbb{B}_t, \widehat{\mathbb{B}}_t^*\}_{t=0,1}, \boldsymbol{f}_{\beta,0}, \{\boldsymbol{f}_{\beta,1,i}\}_{i=1,\ldots,n}).
\end{aligned}
$$

for $\beta \xleftarrow{\mathsf{U}} \{0,1\}$. For a probabilistic machine $\mathcal{C}$, we define the advantage of $\mathcal{C}$ for Basic Problem 1, $\mathsf{Adv}_{\mathcal{C}}^{\mathsf{BP1}}(\lambda)$, as in Definition 8.

Lemma 17 *For any machine $\mathcal{C}$, there is a probabilistic machine $\mathcal{D}$, whose running time is essentially the same as that of $\mathcal{C}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{C}}^{\mathsf{BP1}}(\lambda) \leq \mathsf{Adv}_{\mathcal{D}}^{\mathsf{BP0}}(\lambda)$.*

Proof $\mathcal{D}$ is given a Basic Problem 0 instance

$$(\mathsf{param}_{\mathsf{BP0}}, \widehat{\mathbb{B}}, \mathbb{B}^*, \boldsymbol{y}^*_\beta, \boldsymbol{f}, \kappa G, \xi G).$$

By using $\mathsf{param}_{\mathbb{G}} := (q, \mathbb{G}, \mathbb{G}_T, G, e)$ underlying $\mathsf{param}_{\mathsf{BP0}}$, $\mathcal{D}$ calculates

$$\begin{aligned}
&\mathsf{param}_0 := (q, \mathbb{V}_0, \mathbb{G}_T, \mathbb{A}_0, e) := \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, 5, \mathsf{param}_{\mathbb{G}}),\\
&\mathsf{param}_1 := (q, \mathbb{V}_1, \mathbb{G}_T, \mathbb{A}_1, e) := \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, 4n, \mathsf{param}_{\mathbb{G}}),\\
&\mathsf{param}_n := (\{\mathsf{param}_t\}_{t=0,1}, g_T),
\end{aligned}$$

where g_T is contained in $\mathsf{param}_{\mathsf{BP0}}$.

$\mathcal{D}$ generates random linear transformation defined by matrices $W_0 \xleftarrow{\mathsf{U}} GL(5, \mathbb{F}_q)$ on $\mathbb{V}_0$ and $W_1 \xleftarrow{\mathsf{U}} \mathcal{P}(4, n, \mathbb{F}_q)$ on $\mathbb{V}_1$ as in Remark 16, where $\mathcal{P}(4, n, \mathbb{F}_q)$ is given in Eq. (14). Then $\mathcal{D}$ sets

$$\begin{aligned}
&\boldsymbol{d}_{0,\iota} := W_0(\boldsymbol{b}^*_\iota, 0, 0) \quad \text{for } \iota = 1, 2, \quad \boldsymbol{d}_{0,3} := W_0(0, 0, 0, \xi G, 0),\\
&\boldsymbol{d}_{0,4} := W_0(0, 0, 0, 0, \xi G), \quad \boldsymbol{d}_{0,5} := W_0(\boldsymbol{b}^*_3, 0, 0),\\
&\boldsymbol{d}^*_{0,\iota} := (W_0^{-1})^{\mathrm{T}}(\boldsymbol{b}_\iota, 0, 0) \quad \text{for } \iota = 1, 2, \quad \boldsymbol{d}^*_{0,3} := (W_0^{-1})^{\mathrm{T}}(0, 0, 0, \kappa G, 0),\\
&\boldsymbol{d}^*_{0,4} := (W_0^{-1})^{\mathrm{T}}(0, 0, 0, 0, \kappa G), \quad \boldsymbol{d}^*_{0,5} := (W_0^{-1})^{\mathrm{T}}(\boldsymbol{b}_3, 0, 0),\\
&\boldsymbol{g}_{\beta,0} := W_0(\boldsymbol{y}^*_\beta, 0, 0) + \eta \boldsymbol{d}_{0,5} \text{ where } \eta \xleftarrow{\mathsf{U}} \mathbb{F}_q,\\
&\text{for } i = 1, \ldots, n,\\
&\quad \boldsymbol{p}_{1,4(i-1)+\iota} := W_1(0^{4(i-1)}, \boldsymbol{b}^*_\iota, 0, 0^{4(n-i)}) \text{ for } \iota = 1, 2,\\
&\quad \boldsymbol{p}_{1,4(i-1)+3} := W_1(0^{4(i-1)}, 0^3, \xi G, 0^{4(n-i)}), \quad \boldsymbol{p}_{1,4i} := W_1(0^{4(i-1)}, \boldsymbol{b}^*_3, 0, 0^{4(n-i)}),\\
&\quad \boldsymbol{p}^*_{1,4(i-1)+\iota} := (W_1^{-1})^{\mathrm{T}}(0^{4(i-1)}, \boldsymbol{b}_\iota, 0, 0^{4(n-i)}) \text{ for } \iota = 1, 2,\\
&\quad \boldsymbol{p}^*_{1,4(i-1)+3} := (W_1^{-1})^{\mathrm{T}}(0^{4(i-1)}, 0^3, \kappa G, 0^{4(n-i)}), \quad \boldsymbol{p}^*_{1,4i} := W_1(0^{4(i-1)}, \boldsymbol{b}_3, 0, 0^{4(n-i)}),\\
&\quad \boldsymbol{g}_{\beta,1,i} := W_1(0^{4(i-1)}, \boldsymbol{y}^*_\beta, 0, 0^{4(n-i)}),
\end{aligned}$$

where $(0^{4(i-1)}, \boldsymbol{v}, 0, 0^{4(n-i)}) := (0^{4(i-1)}, \widetilde{G}_1, \widetilde{G}_2, \widetilde{G}_3, 0, 0^{4(n-i)})$ for any $\boldsymbol{v} := (\widetilde{G}_1, \widetilde{G}_2, \widetilde{G}_3) \in \mathbb{V} = \mathbb{G}^3$. Then, $\mathbb{D}_0 := (\boldsymbol{d}_{0,i})_{i=1,\ldots,5}$ and $\mathbb{D}^*_0 := (\boldsymbol{d}^*_{0,i})_{i=1,\ldots,5}$, $\mathbb{P}_1 := (\boldsymbol{p}_{1,i})_{i=1,\ldots,4n}$ and $\mathbb{P}^*_1 := (\boldsymbol{p}^*_{1,i})_{i=1,\ldots,4n}$ are dual orthonormal bases.

Moreover, we see that the distribution of $\mathbb{D}_1$ is equivalent to that of bases generated by using random special type matrix $Y \xleftarrow{\mathsf{U}} \mathcal{P}(4, n, \mathbb{F}_q)$. For the permutation π given in Eq. (13) and the associated matrix Π, the left multiplication by Π gives the permutation π of the basis vectors $\{\boldsymbol{p}_{1,i}\}_{i=1,\ldots,4n}$ and the right multiplication by Π^{-1} gives the permutation π of the coordinates of vectors in $\mathbb{G}^{4n}$. Therefore, by the conjugate action of the matrix Π, we obtain a basis $\mathbb{D}_1 := (\boldsymbol{d}_{1,\iota})_{\iota=1,\ldots,4n}$, whose distribution is equivalent to that of bases generated by using random special type matrix $X \xleftarrow{\mathsf{U}} \mathcal{L}(4, n, \mathbb{F}_q) = \Pi^{-1} \cdot \mathcal{P}(4, n, \mathbb{F}_q) \cdot \Pi$, and its dual $\mathbb{D}^*_1 := (\boldsymbol{d}^*_{1,\iota})_{\iota=1,\ldots,4n}$.

$\mathcal{D}$ can compute $\mathbb{D}_0, \mathbb{D}_1, \widehat{\mathbb{D}}_0^* : = (\boldsymbol{d}_{0,1}^*, \boldsymbol{d}_{0,3}^*, \ldots, \boldsymbol{d}_{0,5}^*)$, $\widehat{\mathbb{D}}_1^* : = (\boldsymbol{d}_{1,1}^*, \ldots, \boldsymbol{d}_{1,n}^*, \boldsymbol{d}_{1,2n+1}^*, \ldots, \boldsymbol{d}_{1,4n}^*)$ from $\widehat{\mathbb{B}} : = (\boldsymbol{b}_1, \boldsymbol{b}_3), \mathbb{B}^*, \kappa G$, and ξG. $\mathcal{D}$ then gives $(\mathsf{param}_n, \{\mathbb{D}_t, \widehat{\mathbb{D}}_t^*\}_{t=0,1}, \boldsymbol{g}_{\beta,0}, \{\boldsymbol{g}_{\beta,1,i}\}_{i=1,\ldots,n})$ to $\mathcal{C}$, and outputs $\beta' \in \{0, 1\}$ if $\mathcal{C}$ outputs β'. $\boldsymbol{g}_{\beta,0}$ is expressed over basis $\mathbb{D}_0$ as

$$\begin{aligned} \boldsymbol{g}_{0,0} &= W_0(\boldsymbol{y}_0^*, 0, 0) + \eta \boldsymbol{d}_{0,5} = (\delta, 0, 0, 0, \sigma_0)_{\mathbb{D}_0}, \\ \boldsymbol{g}_{1,0} &= W_0(\boldsymbol{y}_1^*, 0, 0) + \eta \boldsymbol{d}_{0,5} = (\delta, \rho, 0, 0, \sigma_0)_{\mathbb{D}_0}, \end{aligned}$$

with $\sigma_0 : = \sigma + \eta$, and $\boldsymbol{g}_{\beta,1,i}$ $(i = 1, \ldots, n)$ are expressed over bases $\mathbb{P}_1$ and $\mathbb{D}_1$ as

$$\begin{aligned} \boldsymbol{g}_{0,1,i} &= W_1(0^{4(i-1)}, \boldsymbol{y}_0^*, 0, 0^{4(n-i)}) = (0^{4(i-1)}, \delta, 0, 0, \sigma, 0^{4(n-i)})_{\mathbb{P}_1} \\ &= (\overbrace{\delta \vec{e}_i}^{n}, \overbrace{0^n}^{n}, \overbrace{0^n}^{n}, \overbrace{\sigma \vec{e}_i}^{n})_{\mathbb{D}_1}, \\ \boldsymbol{g}_{1,1,i} &= W_1(0^{4(i-1)}, \boldsymbol{y}_1^*, 0, 0^{4(n-i)}) = (0^{4(i-1)}, \delta, \rho, 0, \sigma, 0^{4(n-i)})_{\mathbb{P}_1} \\ &= (\overbrace{\delta \vec{e}_i}^{n}, \overbrace{\rho \vec{e}_i}^{n}, \overbrace{0^n}^{n}, \overbrace{\sigma \vec{e}_i}^{n})_{\mathbb{D}_1}, \end{aligned}$$

where δ, ρ, σ, and σ_0 are distributed uniformly in $\mathbb{F}_q$. Therefore, the distribution of $(\mathsf{param}_n, \{\mathbb{D}_t, \widehat{\mathbb{D}}_t^*\}_{t=0,1}, \boldsymbol{g}_{\beta,0}, \{\boldsymbol{g}_{\beta,1,i}\}_{i=1,\ldots,n})$ is exactly the same as $\left\{ \varrho \;\middle|\; \varrho \xleftarrow{\mathsf{R}} \mathcal{G}_\beta^{\mathsf{BP1}}(1^\lambda, n) \right\}$. □

Lemma 18 *For any machine $\mathcal{B}$, there is a probabilistic machine $\mathcal{C}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{P1}}(\lambda) = \mathsf{Adv}_{\mathcal{C}}^{\mathsf{BP1}}(\lambda)$.*

Proof Given a Basic Problem 1 instance

$$(\mathsf{param}_n, \{\mathbb{B}_t, \widehat{\mathbb{B}}_t^*\}_{t=0,1}, \boldsymbol{f}_{\beta,0}, \{\boldsymbol{f}_{\beta,1,i}\}_{i=1,\ldots,n}),$$

$\mathcal{C}$ generates $u, u'_n \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times, u'_1, \ldots, u'_{n-1} \xleftarrow{\mathsf{U}} \mathbb{F}_q$ and

$$U : = \begin{pmatrix} u & & & u'_1 \\ & \ddots & & \vdots \\ & & u & u'_{n-1} \\ & & & u'_n \end{pmatrix}, \quad Z : = (U^{-1})^{\mathrm{T}} : = \begin{pmatrix} u^{-1} & & & \\ & \ddots & & \\ & & u^{-1} & \\ -(u'_n)^{-1} u'_1 & \ldots & -(u'_n)^{-1} u'_{n-1} & u_n'^{-1} \end{pmatrix},$$

$(\boldsymbol{d}_{1,n+1}, \ldots, \boldsymbol{d}_{1,2n})^{\mathrm{T}} : = Z \cdot (\boldsymbol{b}_{1,n+1}, \ldots, \boldsymbol{b}_{1,2n})^{\mathrm{T}}$ and $(\boldsymbol{d}_{1,n+1}^*, \ldots, \boldsymbol{d}_{1,2n}^*)^{\mathrm{T}} : = U \cdot (\boldsymbol{b}_{1,n+1}^*, \ldots, \boldsymbol{b}_{1,2n}^*)^{\mathrm{T}}$. We set

$$\begin{aligned} \mathbb{D}_1 &: = (\boldsymbol{b}_{1,1}, \ldots, \boldsymbol{b}_{1,n}, \boldsymbol{d}_{1,n+1}, \ldots, \boldsymbol{d}_{1,2n}, \boldsymbol{b}_{1,2n+1}, \ldots, \boldsymbol{b}_{1,4n}), \\ \mathbb{D}_1^* &: = (\boldsymbol{b}_{1,1}^*, \ldots, \boldsymbol{b}_{1,n}^*, \boldsymbol{d}_{1,n+1}^*, \ldots, \boldsymbol{d}_{1,2n}^*, \boldsymbol{b}_{1,2n+1}^*, \ldots, \boldsymbol{b}_{1,4n}^*). \end{aligned}$$

We then easily verify that $\mathbb{D}_1$ and $\mathbb{D}_1^*$ are dual orthonormal, and are distributed the same as the original bases, $\mathbb{B}_1$ and $\mathbb{B}_1^*$. We note that $\mathcal{C}$ cannot calculate above $\boldsymbol{d}_{1,i}^*$ for $i = n + 1, \ldots, 2n$ (from $\widehat{\mathbb{B}}_1^*$) and $\mathbb{D}_1^*$ is consistent with $\widehat{\mathbb{B}}_1^*$. $\mathcal{C}$ gives $(\mathsf{param}_n, \mathbb{B}_0, \widehat{\mathbb{B}}_0^*, \mathbb{D}_1, \widehat{\mathbb{B}}_1^*, \boldsymbol{f}_{\beta,0}, \{\boldsymbol{f}_{\beta,1,i}\}_{i=1,\ldots,n})$ to $\mathcal{B}$, and outputs $\beta' \in \{0, 1\}$ if $\mathcal{B}$ outputs β'.

Then, with respect to $\mathbb{D}_1, \mathbb{D}_1^*$ (instead of $\mathbb{B}_1, \mathbb{B}_1^*$), the above answer to $\mathcal{B}$ has the same distribution as the Problem 1 instance, i.e., the above instance has the same distribution as the one given by generator $\mathcal{G}_\beta^{\mathsf{P1}}(1^\lambda, n)$. □

Proof of Lemma 5

Lemma 5 *For any machine $\mathcal{B}$, there exists a probabilistic machine $\mathcal{E}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}}(\lambda) \le \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}}(\lambda) + 5/q$.*

Proof Similarly to Lemma 4, we employ the fact that $\mathcal{L}(4, n, \mathbb{F}_q)$ is a subgroup of $GL(4n, \mathbb{F}_q)$ (Lemma 2) in the proof. For the proof of Lemma 5, we give an intermediate problem, Basic Problem 2 below (Definition 12). From Lemmas 16, 19 and 20, we obtain Lemma 5. □

Based on Remark 5, hereafter, we consider the output of $\mathcal{G}^{\mathsf{P2}}_{\beta}(1^{\lambda}, n)$ is expressed as $(\mathsf{param}_n, \widehat{\mathbb{B}}_0, \mathbb{B}^*_0, \boldsymbol{h}^*_{\beta,0}, \boldsymbol{e}_0, \widehat{\mathbb{B}}_1, \mathbb{B}^*_1, \{\boldsymbol{h}^*_{\beta,1,i}, \boldsymbol{e}_{1,i}\}_{i=1,\ldots,n})$ and also we give the output of Basic Problem 2 as such a vector form over bases $\{\mathbb{B}_t, \mathbb{B}^*_t\}_{t=0,1}$.

Definition 12 (*Basic Problem 2*) Basic Problem 2 is to guess $\beta \in \{0, 1\}$, given $(\mathsf{param}_n, \{\widehat{\mathbb{B}}_t, \mathbb{B}^*_t\}_{t=0,1}, \boldsymbol{y}^*_{\beta,0}, \boldsymbol{f}_0, \{\boldsymbol{y}^*_{\beta,1,i}, \boldsymbol{f}_{1,i}\}_{i=1,\ldots,n}) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{BP2}}_{\beta}(1^{\lambda}, n)$, where

$$\begin{aligned}
&\mathcal{G}^{\mathsf{BP2}}_{\beta}(1^{\lambda}, n): \quad (\mathsf{param}_n, \{\mathbb{B}_t, \mathbb{B}^*_t\}_{t=0,1}) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{NIPE,CT}}_{\mathsf{ob}}(1^{\lambda}, 4, n),\\
&\quad \widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \ldots, \boldsymbol{b}_{0,5}), \quad \widehat{\mathbb{B}}_1 := (\boldsymbol{b}_{1,1}, \ldots, \boldsymbol{b}_{1,n}, \boldsymbol{b}_{1,2n+1}, \ldots, \boldsymbol{b}_{1,4n}),\\
&\quad \delta, \delta_0, \omega \xleftarrow{\mathsf{U}} \mathbb{F}_q, \quad \rho, \tau \xleftarrow{\mathsf{U}} \mathbb{F}_q^{\times},\\
&\quad \boldsymbol{y}^*_{0,0} := (\delta, 0, 0, \delta_0, 0)_{\mathbb{B}^*_0}, \quad \boldsymbol{y}^*_{1,0} := (\delta, \rho, 0, \delta_0, 0)_{\mathbb{B}^*_0}, \quad \boldsymbol{f}_0 := (\omega, \tau, 0, 0, 0)_{\mathbb{B}_0},\\
&\quad \text{for } i = 1, \ldots, n;\\
&\quad\ \vec{e}_i := (0^{i-1}, 1, 0^{n-i}) \in \mathbb{F}_q^n,\\
&\quad \boldsymbol{y}^*_{0,1,i} := (\overbrace{\delta\vec{e}_i,}^{n}\ \overbrace{0^n,}^{n}\ \overbrace{\delta_0\vec{e}_i,}^{n}\ \overbrace{0^n}^{n})_{\mathbb{B}^*_1}\\
&\quad \boldsymbol{y}^*_{1,1,i} := (\delta\vec{e}_i,\ \rho\vec{e}_i,\ \delta_0\vec{e}_i,\ 0^n)_{\mathbb{B}^*_1}\\
&\quad \boldsymbol{f}_{1,i} := (\omega\vec{e}_i,\ \tau\vec{e}_i,\ 0^n,\ 0^n)_{\mathbb{B}_1},\\
&\quad \text{return } (\mathsf{param}_n, \{\widehat{\mathbb{B}}_t, \mathbb{B}^*_t\}_{t=0,1}, \boldsymbol{y}^*_{\beta,0}, \boldsymbol{f}_0, \{\boldsymbol{y}^*_{\beta,1,i}, \boldsymbol{f}_{1,i}\}_{i=1,\ldots,n}).
\end{aligned}$$

for $\beta \xleftarrow{\mathsf{U}} \{0, 1\}$. For a probabilistic machine $\mathcal{C}$, we define the advantage of $\mathcal{C}$ for Basic Problem 2, $\mathsf{Adv}^{\mathsf{BP2}}_{\mathcal{C}}(\lambda)$, as in Definition 8.

Lemma 19 *For any machine $\mathcal{C}$, there is a probabilistic machine $\mathcal{D}$, whose running time is essentially the same as that of $\mathcal{C}$, such that for any security parameter λ, $\mathsf{Adv}^{\mathsf{BP2}}_{\mathcal{C}}(\lambda) \le \mathsf{Adv}^{\mathsf{BP0}}_{\mathcal{D}}(\lambda)$.*

Proof $\mathcal{D}$ is given a Basic Problem 0 instance

$$(\mathsf{param}_{\mathsf{BP0}}, \widehat{\mathbb{B}}, \mathbb{B}^*, \boldsymbol{y}^*_{\beta}, \boldsymbol{f}, \kappa G, \xi G).$$

By using $\mathsf{param}_{\mathbb{G}} := (q, \mathbb{G}, \mathbb{G}_T, G, e)$ underlying $\mathsf{param}_{\mathsf{BP0}}$, $\mathcal{D}$ calculates

$$\begin{aligned}
\mathsf{param}_0 &:= (q, \mathbb{V}_0, \mathbb{G}_T, \mathbb{A}_0, e) := \mathcal{G}_{\mathsf{dpvs}}(1^{\lambda}, 5, \mathsf{param}_{\mathbb{G}}),\\
\mathsf{param}_1 &:= (q, \mathbb{V}_1, \mathbb{G}_T, \mathbb{A}_t, e) := \mathcal{G}_{\mathsf{dpvs}}(1^{\lambda}, 4n, \mathsf{param}_{\mathbb{G}}),\\
\mathsf{param}_n &:= (\{\mathsf{param}_t\}_{t=0,1}, g_T),
\end{aligned}$$

where g_T is contained in $\mathsf{param}_{\mathsf{BP0}}$.

$\mathcal{D}$ generates random linear transformations defined by matrices $W_0 \xleftarrow{\mathsf{U}} GL(5, \mathbb{F}_q)$ on $\mathbb{V}_0$ and $W_1 \xleftarrow{\mathsf{U}} \mathcal{P}(4, n, \mathbb{F}_q)$ on $\mathbb{V}_1$ as in Remark 16, where $\mathcal{P}(4, n, \mathbb{F}_q)$ is given in Eq. (14). Then $\mathcal{D}$ sets

$$\begin{aligned}
&\boldsymbol{d}_{0,\iota} :=W_0(\boldsymbol{b}_\iota, 0, 0) \quad \text{for } \iota = 1, 2, \quad \boldsymbol{d}_{0,3} :=W_0(0, 0, 0, \kappa G, 0),\\
&\boldsymbol{d}_{0,4} :=W_0(\boldsymbol{b}_3, 0, 0), \quad \boldsymbol{d}_{0,5} :=W_0(0, 0, 0, 0, \kappa G),\\
&\boldsymbol{d}^*_{0,\iota} :=(W_0^{-1})^{\mathrm{T}}(\boldsymbol{b}^*_\iota, 0, 0) \quad \text{for } \iota = 1, 2, \quad \boldsymbol{d}^*_{0,3} :=(W_0^{-1})^{\mathrm{T}}(0, 0, 0, \xi G, 0),\\
&\boldsymbol{d}^*_{0,4} :=(W_0^{-1})^{\mathrm{T}}(\boldsymbol{b}^*_3, 0, 0) \quad \boldsymbol{d}^*_{0,5} :=(W_0^{-1})^{\mathrm{T}}(0, 0, 0, 0, \xi G),\\
&\boldsymbol{q}^*_{\beta,0} :=(W_0^{-1})^{\mathrm{T}}(\boldsymbol{y}^*_\beta, 0, 0), \qquad \boldsymbol{g}_0 :=W_0(\boldsymbol{f}, 0, 0),\\
&\text{for } i = 1, \ldots, n,\\
&\quad \boldsymbol{p}_{1,4(i-1)+\iota} :=W_1(0^{4(i-1)}, \boldsymbol{b}_\iota, 0, 0^{4(n-i)}) \ \text{ for } \iota = 1, 2, 3,\\
&\quad \boldsymbol{p}_{1,4i} :=W_t(0^{4(i-1)}, 0^3, \kappa G, 0^{4(n-i)}),\\
&\quad \boldsymbol{p}^*_{1,4(i-1)+\iota} :=(W_1^{-1})^{\mathrm{T}}(0^{4(i-1)}, \boldsymbol{b}^*_\iota, 0, 0^{4(n-i)}) \ \text{ for } \iota = 1, 2, 3,\\
&\quad \boldsymbol{p}^*_{1,4i} :=(W_1^{-1})^{\mathrm{T}}(0^{4(i-1)}, 0^3, \xi G, 0^{4(n-i)}),\\
&\quad \boldsymbol{q}^*_{\beta,1,i} :=(W_1^{-1})^{\mathrm{T}}(0^{4(i-1)}, \boldsymbol{y}^*_\beta, 0, 0^{4(n-i)}) + \textstyle\sum_{j=1}^n \eta_{i,j} \boldsymbol{p}^*_{1,4(j-1)+3}\\
&\qquad\qquad\qquad \text{where } \vec{\eta}_i :=(\eta_{i,1}, \ldots, \eta_{i,n}) \xleftarrow{\mathsf{U}} \mathbb{F}_q^n,\\
&\quad \boldsymbol{g}_{1,i} :=W_1(0^{4(i-1)}, \boldsymbol{f}, 0, 0^{4(n-i)})
\end{aligned}$$

where $(0^{4(i-1)}, \boldsymbol{v}, 0, 0^{4(n-i)}) :=(0^{4(i-1)}, \widetilde{G}_1, \widetilde{G}_2, \widetilde{G}_3, 0, 0^{4(n-i)})$ for any $\boldsymbol{v} :=(\widetilde{G}_1, \widetilde{G}_2, \widetilde{G}_3) \in \mathbb{V} = \mathbb{G}^3$. Then, $\mathbb{D}_0 :=(\boldsymbol{d}_{0,i})_{i=1,\ldots,5}$ and $\mathbb{D}^*_0 :=(\boldsymbol{d}^*_{0,i})_{i=1,\ldots,5}$, $\mathbb{P}_1 :=(\boldsymbol{p}_{1,i})_{i=1,\ldots,4n}$ and $\mathbb{P}^*_1 :=(\boldsymbol{p}^*_{1,i})_{i=1,\ldots,4n}$ are dual orthonormal bases.

Moreover, we see that the distribution of $\mathbb{P}_1$ is equivalent to that of bases generated by using random special type matrix $Y \xleftarrow{\mathsf{U}} \mathcal{P}(4, n, \mathbb{F}_q)$. For the permutation π given in Eq. (13) and the associated matrix Π, the left multiplication by Π gives the permutation π of the basis vectors $\{\boldsymbol{p}_{1,i}\}_{i=1,\ldots,4n}$ and the right multiplication by Π^{-1} gives the permutation π of the coordinates of vectors in $\mathbb{G}^{4n}$. Therefore, by the conjugate action of the matrix Π, we obtain a basis $\mathbb{D}_1 :=(\boldsymbol{d}_{1,\iota})_{\iota=1....,4n}$, whose distribution is equivalent to that of bases generated by using random special type matrix $X \xleftarrow{\mathsf{U}} \mathcal{L}(4, n, \mathbb{F}_q) = \Pi^{-1} \cdot \mathcal{P}(4, n, \mathbb{F}_q) \cdot \Pi$, and its dual $\mathbb{D}^*_1 :=(\boldsymbol{d}^*_{1,\iota})_{\iota=1....,4n}$.

$\mathcal{D}$ can compute $\widehat{\mathbb{D}}_0 :=(\boldsymbol{d}_{0,1}, \boldsymbol{d}_{0,3}, \ldots, \boldsymbol{d}_{0,5})$, $\widehat{\mathbb{D}}_1 :=(\boldsymbol{d}_{1,1}, \ldots, \boldsymbol{d}_{1,n}, \boldsymbol{d}_{1,2n+1}, \ldots, \boldsymbol{d}_{1,4n})$, $\mathbb{D}^*_0, \mathbb{D}^*_1$ from $\widehat{\mathbb{B}} :=(\boldsymbol{b}_1, \boldsymbol{b}_3)$, $\mathbb{B}^*$, κG, and ξG. $\mathcal{D}$ then gives $(\mathsf{param}_n, \{\widehat{\mathbb{D}}_t, \mathbb{D}^*_t\}_{t=0,1}, \boldsymbol{q}^*_{\beta,0}, \boldsymbol{g}_0, \{\boldsymbol{q}^*_{\beta,1,i}, \boldsymbol{g}_{1,i}\}_{i=1,\ldots,n})$ to $\mathcal{C}$, and outputs $\beta' \in \{0, 1\}$ if $\mathcal{C}$ outputs β'.

$\boldsymbol{q}^*_{\beta,0}, \boldsymbol{g}_0$ are expressed over bases $(\mathbb{D}_0, \mathbb{D}^*_0)$ as

$$\begin{aligned}
&\boldsymbol{q}^*_{0,0} = (W_0^{-1})^{\mathrm{T}}(\boldsymbol{y}^*_0, 0, 0) = (\delta, 0, 0, \sigma, 0)_{\mathbb{D}^*_0}, \quad \boldsymbol{q}^*_{1,0} = (W_0^{-1})^{\mathrm{T}}(\boldsymbol{y}^*_1, 0, 0) = (\delta, \rho, 0, \sigma, 0)_{\mathbb{D}^*_0},\\
&\boldsymbol{g}_0 = W_0(\boldsymbol{f}, 0, 0) = (\omega, \tau, 0, 0, 0)_{\mathbb{D}_0},
\end{aligned}$$

and $\boldsymbol{q}^*_{\beta,1,i}, \boldsymbol{g}_{1,i}$ $(i = 1, \ldots, n)$ are expressed over bases $(\mathbb{P}_1, \mathbb{P}^*_1)$ and $(\mathbb{D}_1, \mathbb{D}^*_1)$ as

$$\begin{aligned}
\boldsymbol{q}^*_{0,1,i} &= (W_1^{-1})^{\mathrm{T}}(0^{4(i-1)}, \boldsymbol{y}^*_0, 0, 0^{4(n-i)}) + \textstyle\sum_{j=1}^n \eta_{i,j} \boldsymbol{p}^*_{1,4(j-1)+3}\\
&= (0^{4(i-1)}, \delta, 0, \sigma, 0, 0^{4(n-i)})_{\mathbb{P}^*_1} + \textstyle\sum_{j=1}^n \eta_{i,j} \boldsymbol{p}^*_{1,4(j-1)+3} = (\overbrace{\delta\vec{e}_i}^{n}, \overbrace{0^n}^{n}, \overbrace{\vec{\varphi}_i}^{n}, \overbrace{0^n}^{n})_{\mathbb{D}^*_1},\\
\boldsymbol{q}^*_{1,1,i} &= (W_1^{-1})^{\mathrm{T}}(0^{4(i-1)}, \boldsymbol{y}^*_1, 0, 0^{4(n-i)}) + \textstyle\sum_{j=1}^n \eta_{i,j} \boldsymbol{p}^*_{1,4(j-1)+3}
\end{aligned}$$

$$
\begin{aligned}
&= (0^{4(i-1)}, \delta, \rho, \sigma, 0, 0^{4(n-i)})_{\mathbb{P}_1^*} + \textstyle\sum_{j=1}^{n} \eta_{i,j}\, \boldsymbol{p}^*_{1,4(j-1)+3} = (\overbrace{\delta\vec{e}_i}^{n}, \overbrace{\rho\vec{e}_i}^{n}, \overbrace{\vec{\varphi}_i}^{n}, \overbrace{0^n}^{n})_{\mathbb{D}_1^*}, \\
\boldsymbol{g}_{1,i} &= W_1(0^{4(i-1)}, \boldsymbol{f}, 0, 0^{4(n-i)}) = (0^{4(i-1)}, \omega, \tau, 0, 0, 0^{4(n-i)})_{\mathbb{P}_1} \\
&= (\overbrace{\omega\vec{e}_i}^{n}, \overbrace{\tau\vec{e}_i}^{n}, \overbrace{0^n}^{n}, \overbrace{0^n}^{n})_{\mathbb{D}_1},
\end{aligned}
$$

where $\vec{\varphi}_i := \sigma\vec{e}_i + \vec{\eta}_i$, and $\delta, \rho, \sigma, \omega, \tau \in \mathbb{F}_q$, and $\vec{\varphi}_i \in \mathbb{F}_q^n$ are uniformly and independently distributed. Therefore, the distribution of $(\mathsf{param}_n, \{\widehat{\mathbb{D}}_t, \mathbb{D}_t^*\}_{t=0,1}, \boldsymbol{q}^*_{\beta,0}, \boldsymbol{g}_0, \{\boldsymbol{q}^*_{\beta,1,i}, \boldsymbol{g}_{1,i}\}_{i=1,\ldots,n})$ is exactly the same as $\left\{ \varrho \;\middle|\; \varrho \xleftarrow{\mathsf{R}} \mathcal{G}_\beta^{\mathsf{BP2}}(1^\lambda, n) \right\}$. □

Lemma 20 *For any machine $\mathcal{B}$, there is a probabilistic machine $\mathcal{C}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{P2}}(\lambda) = \mathsf{Adv}_{\mathcal{C}}^{\mathsf{BP2}}(\lambda)$.*

Proof Given a Basic Problem 2 instance

$$(\mathsf{param}_n, \{\widehat{\mathbb{B}}_t, \mathbb{B}_t^*\}_{t=0,1}, \boldsymbol{y}^*_{\beta,0}, \boldsymbol{f}_0, \{\boldsymbol{y}^*_{\beta,1,i}, \boldsymbol{f}_{1,i}\}_{i=1,\ldots,n}),$$

$\mathcal{C}$ generates $u, u'_n \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times, u'_1, \ldots, u'_{n-1} \xleftarrow{\mathsf{U}} \mathbb{F}_q$ and

$$
U := \begin{pmatrix} u & & & u'_1 \\ & \ddots & & \vdots \\ & & u & u'_{n-1} \\ & & & u'_n \end{pmatrix}, \; Z := (U^{-1})^{\mathrm{T}} := \begin{pmatrix} u^{-1} & & & \\ & \ddots & & \\ & & u^{-1} & \\ -(u'_n)^{-1}u'_1 & \ldots & -(u'_n)^{-1}u'_{n-1} & u_n'^{-1} \end{pmatrix},
$$

$(\boldsymbol{d}_{1,n+1}, \ldots, \boldsymbol{d}_{1,2n})^{\mathrm{T}} := Z \cdot (\boldsymbol{b}_{1,n+1}, \ldots, \boldsymbol{b}_{1,2n})^{\mathrm{T}}$ and $(\boldsymbol{d}^*_{1,n+1}, \ldots, \boldsymbol{d}^*_{1,2n})^{\mathrm{T}} := U \cdot (\boldsymbol{b}^*_{1,n+1}, \ldots, \boldsymbol{b}^*_{1,2n})^{\mathrm{T}}$. We set

$$
\begin{aligned}
\mathbb{D}_1 &:= (\boldsymbol{b}_{1,1}, \ldots, \boldsymbol{b}_{1,n}, \boldsymbol{d}_{1,n+1}, \ldots, \boldsymbol{d}_{1,2n}, \boldsymbol{b}_{1,2n+1}, \ldots, \boldsymbol{b}_{1,4n}), \\
\mathbb{D}_1^* &:= (\boldsymbol{b}^*_{1,1}, \ldots, \boldsymbol{b}^*_{1,n}, \boldsymbol{d}^*_{1,n+1}, \ldots, \boldsymbol{d}^*_{1,2n}, \boldsymbol{b}^*_{1,2n+1}, \ldots, \boldsymbol{b}^*_{1,4n}).
\end{aligned}
$$

We then easily verify that $\mathbb{D}_1$ and $\mathbb{D}_1^*$ are dual orthonormal, and are distributed the same as the original bases, $\mathbb{B}_1$ and $\mathbb{B}_1^*$. We note that $\mathcal{C}$ cannot calculate above $\boldsymbol{d}_{1,i}$ for $i = n+1, \ldots, 2n$ (from $\widehat{\mathbb{B}}_1$) and $\mathbb{D}_1$ is consistent with $\widehat{\mathbb{B}}_1$. $\mathcal{C}$ gives $(\mathsf{param}_n, \widehat{\mathbb{B}}_0, \mathbb{B}_0^*, \widehat{\mathbb{B}}_1, \mathbb{D}_1^*, \boldsymbol{y}^*_{\beta,0}, \boldsymbol{f}_0, \{\boldsymbol{y}^*_{\beta,1,i}, \boldsymbol{f}_{1,i}\}_{i=1,\ldots,n})$ to $\mathcal{B}$, and outputs $\beta' \in \{0, 1\}$ if $\mathcal{B}$ outputs β'.

Then, with respect to $\mathbb{D}_1, \mathbb{D}_1^*$ (instead of $\mathbb{B}_1, \mathbb{B}_1^*$), the above answer to $\mathcal{B}$ has the same distribution as the Problem 2 instance, i.e., the above instance has the same distribution as the one given by generator $\mathcal{G}_\beta^{\mathsf{P2}}(1^\lambda, n)$. □

Next is a key lemma for applying the proof techniques in [25] to our NIPE (and ZIPE) schemes, where *limited randomness* is used in public parameter, e.g., $\{B_{i,j}, B'_{i,j,l}\}_{i=1,4;\, j=1,\ldots,4;\, l=1,\ldots,n}$, in the NIPE scheme in Sect. 6.

Proof of Lemma 6

Lemma 6 *Let $\vec{e}_n := (0, \ldots, 0, 1) \in \mathbb{F}_q^n$. For all $\vec{x} \in \mathbb{F}_q^n \setminus \mathsf{span}\langle\vec{e}_n\rangle$ and $\pi \in \mathbb{F}_q$, let $W_{\vec{x},\pi} := \{(\vec{r}, \vec{w}) \in (\mathsf{span}\langle\vec{x}, \vec{e}_n\rangle \setminus \mathsf{span}\langle\vec{e}_n\rangle) \times (\mathbb{F}_q^n \setminus \mathsf{span}\langle\vec{e}_n\rangle^\perp) \mid \vec{r} \cdot \vec{w} = \pi\}$.*

For all $(\vec{x},\vec{v}) \in \left(\mathbb{F}_q^n \setminus \mathsf{span}\langle\vec{e}_n\rangle\right) \times \left(\mathbb{F}_q^n \setminus \mathsf{span}\langle\vec{e}_n\rangle^{\perp}\right)$, *for all* $(\vec{r},\vec{w}) \in W_{\vec{x},(\vec{x}\cdot\vec{v})}$, $\Pr\left[\,\vec{x}U=\vec{r} \wedge \ \vec{v}Z=\vec{w}\,\right] = 1/\sharp\, W_{\vec{x},(\vec{x}\cdot\vec{v})}$, *where* $U \xleftarrow{\mathsf{U}} \mathcal{H}(n,\mathbb{F}_q) \cap GL(n,\mathbb{F}_q)$ *and* $Z := (U^{-1})^{\mathrm{T}}$.

Proof Let
$$\begin{pmatrix} u & & & u'_1 \\ & \ddots & & \vdots \\ & & u & u'_{n-1} \\ & & & u'_n \end{pmatrix} := U, \quad \begin{pmatrix} u^{-1} & & & \\ & \ddots & & \\ & & u^{-1} & \\ -(uu'_n)^{-1}u'_1 & \dots & -(uu'_n)^{-1}u'_{n-1} & (u'_n)^{-1} \end{pmatrix}$$
$:=(U^{-1})^{\mathrm{T}} := Z$, and $\vec{u}' := (u'_1,\dots,u'_n)$. For $\vec{x} := (x_1,\dots,x_n)$ and $\vec{v} := (v_1,\dots,v_n)$ with $v_n \neq 0$, let

$$\begin{aligned} \vec{r} &:= \vec{x}U = (ux_1,\dots,ux_{n-1},\vec{x}\cdot\vec{u}') = (ux_1,\dots,ux_{n-1},p), \quad \text{and} \\ \vec{w} &:= \vec{v}Z = (u^{-1}v_1 - u'_1(uu'_n)^{-1}v_n,\dots,u^{-1}v_{n-1} - u'_{n-1}(uu'_n)^{-1}v_n,(u'_n)^{-1}v_n) \\ &= (u'_n)^{-1}v_n \cdot \left(u^{-1}\left(u'_n(v_1v_n^{-1}) - u'_1\right),\dots,u^{-1}\left(u'_n(v_{n-1}v_n^{-1}) - u'_{n-1}\right),\ 1\right) \\ &= (u'_n)^{-1}v_n \cdot (\widetilde{u}_1,\dots,\widetilde{u}_{n-1},1), \end{aligned}$$

where $\widetilde{u}_j := u^{-1}\left(u'_n(v_jv_n^{-1}) - u'_j\right)$ for $j=1,\dots,n-1$ and $p := \vec{x}\cdot\vec{u}'$. Then,

$$\vec{x}\cdot\vec{v} = \left(u'_n\right)^{-1} v_n \left(\textstyle\sum_{j=1}^{n-1}(ux_j)\widetilde{u}_j + p\right) = \vec{r}\cdot\vec{w}. \tag{15}$$

Case that $\boldsymbol{\vec{x}\cdot\vec{v}\neq 0}$ Since $\vec{x}\cdot\vec{v}\neq 0$, u and $\vec{u}'$ can be generated as: $(u,\widetilde{u}_1,\dots,\widetilde{u}_{n-1},p) \xleftarrow{\mathsf{U}} \{(u,(\widetilde{u}_j)_{j=1,\dots,n-1},p) \in \mathbb{F}_q^{\times} \times \mathbb{F}_q^n \mid \sum_{j=1}^{n-1}(ux_j)\widetilde{u}_j + p \neq 0\}$, $u'_n := v_n(\sum_{j=1}^{n-1}(ux_j)\widetilde{u}_j + p)/(\vec{x}\cdot\vec{v})$, and $u'_j := u'_n(v_jv_n^{-1}) - u\widetilde{u}_j$ for $j=1,\dots,n-1$. We note that the condition $\sum_{j=1}^{n-1}(ux_j)\widetilde{u}_j + p \neq 0$ among $u,\widetilde{u}_j$ $(j=1,\dots,n-1)$ and p is equivalent to the condition $u'_n \neq 0$.

Since $(u,\widetilde{u}_1,\dots,\widetilde{u}_{n-1},p) \xleftarrow{\mathsf{U}} \{(u,(\widetilde{u}_j)_{j=1,\dots,n-1},p) \in \mathbb{F}_q^{\times} \times \mathbb{F}_q^n \mid \sum_{j=1}^{n-1}(ux_j)\widetilde{u}_j + p \neq 0\}$ and $u'_n := v_n(\sum_{j=1}^{n-1}(ux_j)\widetilde{u}_j + p)/(\vec{x}\cdot\vec{v})$, the pair of $\vec{r} := (ux_1,\dots,ux_{n-1},p)$ and $\vec{w} := \left(u'_n\right)^{-1} v_n \cdot (\widetilde{u}_1,\dots,\widetilde{u}_{n-1},1)$ is uniformly distributed in $W_{\vec{x},(\vec{x}\cdot\vec{v})}$.

Case that $\boldsymbol{\vec{x}\cdot\vec{v}=0}$ Since $\vec{x}\cdot\vec{v}=0$, Eq. (15) is given as $\sum_{j=1}^{n-1}(ux_j)\widetilde{u}_j + p = 0$. Since $\vec{x} \notin \mathsf{span}\langle\vec{e}_n\rangle$, there exists an index $j_0 \in \{1,\dots,n-1\}$ such that $x_{j_0} \neq 0$. Using the index j_0, u and $\vec{u}'$ can be generated as: $u \xleftarrow{\mathsf{U}} \mathbb{F}_q^{\times}$, $\widetilde{u}_j \xleftarrow{\mathsf{U}} \mathbb{F}_q$ $(j=1,\dots,j_0-1,j_0+1,\dots,n-1)$, $p \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $u'_{j_0} := (-\sum_{j=1,\dots,j_0-1,j_0+1,n-1} x_ju'_j - u^{-1}p)/x_{j_0}$, $u'_n \xleftarrow{\mathsf{U}} \mathbb{F}_q^{\times}$ and $u'_j := u'_n(v_jv_n^{-1}) - u\widetilde{u}_j$ for $j=1,\dots,n-1$.

Since $(u,\widetilde{u}_1,\dots,\widetilde{u}_{n-1},p) \xleftarrow{\mathsf{U}} \{(u,(\widetilde{u}_j)_{j=1,\dots,n-1},p) \in \mathbb{F}_q^{\times} \times \mathbb{F}_q^n \mid \sum_{j=1}^{n-1}(ux_j)\widetilde{u}_j + p = 0\}$ and $u'_n \xleftarrow{\mathsf{U}} \mathbb{F}_q^{\times}$, the pair of $\vec{r} := (ux_1,\dots,ux_{n-1},p)$ and $\vec{w} := \left(u'_n\right)^{-1} v_n \cdot (\widetilde{u}_1,\dots,\widetilde{u}_{n-1},1)$ is uniformly distributed in $W_{\vec{x},0}$. □

Proof of Lemma 7

Lemma 7 *For any machine $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_1$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ,* $|\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)| \leq \mathsf{Adv}_{\mathcal{B}_1}^{\mathsf{P1}}(\lambda)$.

Proof Lemma 7 is proven by the same manner as the proof of Lemma 4 in [25].

In order to prove Lemma 7, we construct a probabilistic machine $\mathcal{B}_1$ against Problem 1 using an adversary $\mathcal{A}$ in a security game (Game 0 or 1) as a black box as follows:

1. $\mathcal{B}_1$ is given a Problem 1 instance, $(\mathsf{param}_n, \mathbb{B}_0, \widehat{\mathbb{B}}^*_0, \boldsymbol{e}_{\beta,0}, \{B_{i,j}, B'_{i,j,l}\}_{i,j=1,\ldots,4;l=1,\ldots,n}, \widehat{\mathbb{B}}^*_1, \{E_{\beta,j},\ E'_{\beta,j,l}\}_{j=1,\ldots,4;l=1,\ldots,n})$, which is identified with $(\mathsf{param}_n, \mathbb{B}_0, \widehat{\mathbb{B}}^*_0, \boldsymbol{e}_{\beta,0}, \mathbb{B}_1, \widehat{\mathbb{B}}^*_1, \{\boldsymbol{e}_{\beta,1,l}\}_{l=1,\ldots,n})$ (Remark 4).
2. $\mathcal{B}_1$ plays a role of the challenger in the security game against adversary $\mathcal{A}$.
3. At the first step of the game, $\mathcal{B}_1$ provides $\mathcal{A}$ a public key $\mathsf{pk} :=(1^\lambda, \mathsf{param}_n, \{\widehat{\mathbb{B}}_t\}_{t=0,1})$ of Game 0 (and 1), where $\widehat{\mathbb{B}}_0 :=(\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5})$ and $\widehat{\mathbb{B}}_1 :=(\boldsymbol{b}_{1,1}, \ldots, \boldsymbol{b}_{1,n}, \boldsymbol{b}_{1,3n+1}, \ldots, \boldsymbol{b}_{1,4n})$, which are obtained from the Problem 1 instance.
4. When a key query is issued for vector $\vec{v}$, $\mathcal{B}_1$ answers normal key $(\boldsymbol{k}^*_0, \boldsymbol{k}^*_1)$ with Eq. (7), which is computed using $\{\widehat{\mathbb{B}}^*_t\}_{t=0,1}$ of the Problem 1 instance.
5. When $\mathcal{B}_1$ receives an encryption query with challenge plaintexts $(m^{(0)}, m^{(1)})$ and vector $\vec{x} :=(x_1, \ldots, x_n)$ from $\mathcal{A}$, $\mathcal{B}_1$ computes the challenge ciphertext $(\vec{x}, \boldsymbol{c}_0, \{C_{1,j}, C_{2,j}\}_{j=1,\ldots,4}, c_3)$ which is identified with $(\vec{x}, \boldsymbol{c}_0, \boldsymbol{c}_1, c_3)$ in Remark 3 such that $\boldsymbol{c}_0 := -\boldsymbol{e}_{\beta,0} + \zeta \boldsymbol{b}_{0,3}$, $\boldsymbol{c}_1 := \sum_{l=1}^{n} x_l \boldsymbol{e}_{\beta,1,l}$, $c_3 := g_T^{\zeta} m^{(b)}$, where $b \xleftarrow{\mathsf{U}} \{0,1\}$, $\zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$, and $(\boldsymbol{e}_{\beta,0}, \boldsymbol{b}_{0,3}, \{\boldsymbol{e}_{\beta,1,l}\}_{l=1,\ldots,n})$ is a part of the Problem 1 instance.
6. When a key query is issued by $\mathcal{A}$ after the encryption query, $\mathcal{B}_1$ executes the same procedure as that of step 4.
7. $\mathcal{A}$ finally outputs bit b'. If $b = b'$, $\mathcal{B}_1$ outputs $\beta' :=1$. Otherwise, $\mathcal{B}_1$ outputs $\beta' :=0$.

Claim 2 *The distribution of the view of adversary $\mathcal{A}$ in the above-mentioned game simulated by $\mathcal{B}_1$ given a Problem 1 instance with $\beta \in \{0,1\}$ is the same as that in Game 0 (resp. Game 1) if $\beta = 0$ (resp. $\beta = 1$).*

Proof Since the public key pk and secret keys $\mathsf{sk}_{\vec{v}}$ answered by $\mathcal{A}$ are distributed as in Game 0 and 1, we consider the distribution of challenge ciphertext $\mathsf{ct}_{\vec{x}} :=(\vec{x}, \boldsymbol{c}_0, \{C_{1,j}, C_{2,j}\}_{j=1,\ldots,4}, c_3)$ which is equivalent to $(\vec{x}, \boldsymbol{c}_0, \boldsymbol{c}_1, c_3)$ under the identification Eq. (6).

When $\beta = 0$, ciphertext $\mathsf{ct}_{\vec{x}}$ generated in step 5 is

$$\boldsymbol{c}_0 = -\boldsymbol{e}_{0,0} + \zeta \boldsymbol{b}_{0,3} = (-\omega,\ 0,\ \zeta,\ 0,\ -\eta_0)_{\mathbb{B}_0}, \quad c_3 := g_T^{\zeta} m^{(b)},$$
$$\boldsymbol{c}_1 = \textstyle\sum_{l=1}^{n} x_l \boldsymbol{e}_{0,1,l} = (\omega\vec{x},\ 0^n,\ 0^n,\ \eta_1\vec{x})_{\mathbb{B}_1},$$

where variables $\omega, \zeta, \eta_0, \eta_1 \in \mathbb{F}_q$ are uniformly and independently distributed. Therefore, generated $\mathsf{ct}_{\vec{x}}$ and $\mathsf{sk}_{\vec{v}}$ have the same distribution as in Game 0.

When $\beta = 1$, ciphertext $\mathsf{ct}_{\vec{x}}$ generated in step 5 is

$$\boldsymbol{c}_0 = -\boldsymbol{e}_{1,0} + \zeta \boldsymbol{b}_{0,3} = (-\omega,\ -\tau,\ \zeta,\ 0,\ -\eta_0)_{\mathbb{B}_0}, \quad c_3 := g_T^{\zeta} m^{(b)},$$
$$\boldsymbol{c}_1 = \textstyle\sum_{l=1}^{n} x_l \boldsymbol{e}_{1,1,l} = (\omega\vec{x},\ \tau\vec{x},\ 0^n,\ \eta_1\vec{x})_{\mathbb{B}_1},$$

where variables $\omega, \tau, \zeta, \eta_0, \eta_1 \in \mathbb{F}_q$ are uniformly and independently distributed. Therefore, generated $\mathsf{ct}_{\vec{x}}$ and $\mathsf{sk}_{\vec{v}}$ have the same distribution as in Game 1. □

This completes the proof of Lemma 7. □

Proof of Lemma 8

Lemma 8 *For any machine $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_{2\text{-}1}$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $|\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}(h-1)\text{-}3)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}h\text{-}1)}(\lambda)| \le \mathsf{Adv}_{\mathcal{B}_{2\text{-}h\text{-}1}}^{\mathsf{P2}}(\lambda)$, where $\mathcal{B}_{2\text{-}h\text{-}1}(\cdot) :=\mathcal{B}_{2\text{-}1}(h, \cdot)$.*

Proof Lemma 8 is proven by the same manner as the proof of Lemma 5 in [25].

In order to prove Lemma 8, we construct a probabilistic machine $\mathcal{B}_{2\text{-}1}$ against Problem 2 using an adversary $\mathcal{A}$ in a security game (Game 2-$(h-1)$-3 or 2-h-1) as a black box as follows:

1. $\mathcal{B}_{2\text{-}1}$ is given an integer h and a Problem 2 instance, $(\mathsf{param}_n, \widehat{\mathbb{B}}_0, \mathbb{B}_0^*, \boldsymbol{h}^*_{\beta,0}, \boldsymbol{e}_0, \{B_{i,j}, B'_{i,j,l}\}_{i=1,3,4; j=1,\ldots,4; l=1,\ldots,n}, \mathbb{B}_1^*, \{\boldsymbol{h}^*_{\beta,1,l}, E_j, E'_{j,l}\}_{j=1,\ldots,4; l=1,\ldots,n})$, which is identified with $(\mathsf{param}_n, \widehat{\mathbb{B}}_0, \mathbb{B}_0^*, \boldsymbol{h}^*_{\beta,0}, \boldsymbol{e}_0, \widehat{\mathbb{B}}_1, \mathbb{B}_1^*, \{\boldsymbol{h}^*_{\beta,1,l}, \boldsymbol{e}_{1,l}\}_{l=1,\ldots,n})$ (Remark 5).
2. $\mathcal{B}_{2\text{-}1}$ plays a role of the challenger in the security game against adversary $\mathcal{A}$.
3. At the first step of the game, $\mathcal{B}_{2\text{-}1}$ provides $\mathcal{A}$ a public key $\mathsf{pk} := (1^\lambda, \mathsf{param}_n, \{\widehat{\mathbb{B}}'_t\}_{t=0,1})$ of Game 2-$(h-1)$-3 (and 2-h-1), where $\widehat{\mathbb{B}}'_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5})$ and $\widehat{\mathbb{B}}'_1 := (\boldsymbol{b}_{1,1}, \ldots, \boldsymbol{b}_{1,n}, \boldsymbol{b}_{1,3n+1}, \ldots, \boldsymbol{b}_{1,4n})$.
4. When the ι-th key query is issued for $\vec{v} := (v_1, \ldots, v_n)$, $\mathcal{B}_{2\text{-}1}$ answers as follows:
   (a) When $1 \le \iota \le h-1$, $\mathcal{B}_{2\text{-}1}$ answers semi-functional keys of the form Eq. (12), which is computed using $(\mathbb{B}_0^*, \mathbb{B}_1^*)$ of the Problem 2 instance.
   (b) When $\iota = h$, $\mathcal{B}_{2\text{-}1}$ calculates $(\boldsymbol{k}_0^*, \boldsymbol{k}_1^*)$ using $(\boldsymbol{h}^*_{\beta,0}, \{\boldsymbol{h}^*_{\beta,1,l}\}_{l=1,\ldots,n})$ of the Problem 2 instance as follows: $\boldsymbol{k}_0^* := \boldsymbol{h}^*_{\beta,0} + \boldsymbol{b}^*_{0,3}$, $\boldsymbol{k}_1^* := \sum_{l=1}^n v_l \boldsymbol{h}^*_{\beta,1,l}$, where $(\boldsymbol{h}^*_{\beta,0}, \boldsymbol{b}^*_{0,3}, \{\boldsymbol{h}^*_{\beta,1,l}\}_{l=1,\ldots,n})$ is a part of the Problem 2 instance.
   (c) When $\iota \ge h+1$, $\mathcal{B}_{2\text{-}1}$ answers normal keys of the form Eq. (7), which is computed using $(\mathbb{B}_0^*, \mathbb{B}_1^*)$ of the Problem 2 instance.
5. When $\mathcal{B}_{2\text{-}1}$ receives an encryption query with challenge plaintexts $(m^{(0)}, m^{(1)})$ and vector $\vec{x} := (x_1, \ldots, x_n)$ from $\mathcal{A}$, $\mathcal{B}_1$ computes the challenge ciphertext $(\vec{x}, \boldsymbol{c}_0, \{C_{1,j}, C_{2,j}\}_{j=1,\ldots,4}, c_3)$ which is identified with $(\vec{x}, \boldsymbol{c}_0, \boldsymbol{c}_1, c_3)$ in Remark 3 such that $\boldsymbol{c}_0 := -\boldsymbol{e}_0 + \zeta \boldsymbol{b}_{0,3} + \eta_0 \boldsymbol{b}_{0,5}$, $\boldsymbol{c}_1 := \sum_{l=1}^n x_l(\boldsymbol{e}_{1,l} + \eta_1 \boldsymbol{b}_{1,3n+l})$, $c_3 := g_T^\zeta m^{(b)}$, where $b \overset{\mathsf{U}}{\leftarrow} \{0,1\}$, $\zeta, \eta_0, \eta_1 \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q$, and $(\boldsymbol{e}_0, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5}, \{\boldsymbol{e}_{1,l}, \boldsymbol{b}_{1,3n+l}\}_{l=1,\ldots,n})$ is a part of the Problem 2 instance.
6. When a key query is issued by $\mathcal{A}$ after the encryption query, $\mathcal{B}_{2\text{-}1}$ executes the same procedure as that of step 4.
7. $\mathcal{A}$ finally outputs bit b'. If $b = b'$, $\mathcal{B}_{2\text{-}1}$ outputs $\beta' := 1$. Otherwise, $\mathcal{B}_{2\text{-}1}$ outputs $\beta' := 0$.

Claim 3 *The distribution of the view of adversary $\mathcal{A}$ in the above-mentioned game simulated by $\mathcal{B}_{2\text{-}1}$ given a Problem 2 instance with $\beta \in \{0,1\}$ is the same as that in Game 2-$(h-1)$-3 (resp. Game 2-h-1) if $\beta = 0$ (resp. $\beta = 1$).*

Proof We consider the joint distribution of $\mathsf{ct}_{\vec{x}}$ and $\mathsf{sk}_{\vec{v}}$. We see that the distribution of challenge ciphertext $\mathsf{ct}_{\vec{x}} := (\vec{x}, \boldsymbol{c}_0, \{C_{1,j}, C_{2,j}\}_{j=1,\ldots,4}, c_3)$ is the same as that in Game 2-$(h-1)$-3 (and Game 2-h-1) similarly to the proof of Claim 2 for the case with $\beta = 1$.

When $\beta = 0$, the h-th secret key $\mathsf{sk}_{\vec{v}} := (\vec{v}, \boldsymbol{k}_0^*, \boldsymbol{k}_1^*)$ generated in case (b) of step 4 or 6 is $\boldsymbol{k}_0^* = \boldsymbol{h}^*_{0,0} + \boldsymbol{b}^*_{0,3} = (\delta, 0, 1, \varphi_0, 0)_{\mathbb{B}_0^*}$, $\boldsymbol{k}_1^* = \sum_{l=1}^n v_l \boldsymbol{h}^*_{0,1,l} = (\,\delta\vec{v},\ 0^n,\ \vec{\varphi}'_1,\ 0^n\,)_{\mathbb{B}_1^*}$, where, variables $\delta, \varphi_0 \in \mathbb{F}_q$, $\vec{\varphi}'_1 := \sum_{l=1}^n v_l \vec{\varphi}_l \in \mathbb{F}_q^n$ are uniformly and independently distributed. Therefore, generated $\mathsf{ct}_{\vec{x}}$ and $\mathsf{sk}_{\vec{v}}$ have the same joint distribution as in Game 2-$(h-1)$-3.

When $\beta = 1$, the h-th secret key $\mathsf{sk}_{\vec{v}} := (\vec{v}, \boldsymbol{k}_0^*, \boldsymbol{k}_1^*)$ generated in case (b) of step 4 or 6 is $\boldsymbol{k}_0^* = \boldsymbol{h}^*_{1,0} + \boldsymbol{b}^*_{0,3} = (\delta, \rho, 1, \varphi_0, 0)_{\mathbb{B}_0^*}$, $\boldsymbol{k}_1^* = \sum_{l=1}^n v_l \boldsymbol{h}^*_{1,1,l} = (\,\delta\vec{v},\ \rho\vec{v}Z,\ \vec{\varphi}'_1,\ 0^n\,)_{\mathbb{B}_1^*}$, where,

$$Z := \begin{pmatrix} z & & & \\ & \ddots & & \\ & & z & \\ z'_1 & \ldots & z'_{n-1} & z'_n \end{pmatrix} := \begin{pmatrix} u^{-1} & & & \\ & \ddots & & \\ & & u^{-1} & \\ -(uu'_n)^{-1}u'_1 & \ldots & -(uu'_n)^{-1}u'_{n-1} & (u'_n)^{-1} \end{pmatrix} := (U^{-1})^{\mathrm{T}}$$

for $U \xleftarrow{\mathsf{U}} \mathcal{H}(n, \mathbb{F}_q) \cap GL(n, \mathbb{F}_q)$ used for challenge ciphertext $\mathsf{ct}_{\vec{x}}$, variables $\delta, \varphi_0 \in \mathbb{F}_q, \vec{\varphi}'_1 := \sum_{l=1}^{n} v_l \vec{\varphi}_l \in \mathbb{F}_q^n$ are uniformly and independently distributed. Therefore, generated $\mathsf{ct}_{\vec{x}}$ and $\mathsf{sk}_{\vec{v}}$ have the same joint distribution as in Game 2-h-1. □

This completes the proof of Lemma 8. □

Proof of Lemma 9

Lemma 9 *For any machine $\mathcal{A}$, for any security parameter* λ, $|\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}h\text{-}1)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}h\text{-}2)}(\lambda)| \leq 1/q$.

Proof We consider joint distribution of the h-th answered key $(\vec{v}, \boldsymbol{k}_0^*, \boldsymbol{k}_1^*)$ and the challenge ciphertext $(\vec{x}, \boldsymbol{c}_0, \boldsymbol{c}_1)$ in Game 2-h-1.

$$\begin{aligned} &\boldsymbol{k}_0^* := (\ \delta,\ \rho,\ 1,\ \varphi_0,\ 0\ )_{\mathbb{B}_0^*}, && \boldsymbol{k}_1^* := (\ \delta\vec{v},\ \rho\vec{v}Z,\ \vec{\varphi}_1,\ 0^n\ )_{\mathbb{B}_1^*}, \\ &\boldsymbol{c}_0 := (\ -\omega,\ -\tau,\ \zeta,\ 0,\ \eta_0\ )_{\mathbb{B}_0}, && \boldsymbol{c}_1 := (\ \omega\vec{x},\ \tau\vec{x}U,\ 0^n,\ \eta_1\vec{x}\ )_{\mathbb{B}_1}, \end{aligned}$$

where $\delta, \rho, \varphi_0, \omega, \tau, \zeta, \eta_0, \eta_1 \xleftarrow{\mathsf{U}} \mathbb{F}_q, \vec{\varphi}_1 \xleftarrow{\mathsf{U}} \mathbb{F}_q^n$, $U \xleftarrow{\mathsf{U}} \mathcal{H}(n, \mathbb{F}_q) \cap GL(n, \mathbb{F}_q)$ and $Z := (U^{-1})^{\mathrm{T}}$.

By the security definition, it holds that $\vec{x} \cdot \vec{v} = 0$. From Lemma 6, $(\tau\vec{x}U, \rho\vec{v}Z)$ is uniformly distributed in $W_{\tau\vec{x},0}$. In particular, if $\tau \neq 0$, it is uniformly distributed in $W_{\vec{x},0}$. That is, coefficient $-\tau$ in $\boldsymbol{k}_0^*$ is independent from all the other variables except with negligible probability $1/q$, and the joint distribution is equivalent to that in Game 2-h-2 except with negligible probability $1/q$. □

Proof of Lemma 10

Lemma 10 *For any machine $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_{2\text{-}2}$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter* λ, $|\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}h\text{-}2)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(2\text{-}h\text{-}3)}(\lambda)| \leq \mathsf{Adv}_{\mathcal{B}_{2\text{-}h\text{-}2}}^{\mathsf{P2}}(\lambda)$, *where* $\mathcal{B}_{2\text{-}h\text{-}2}(\cdot) := \mathcal{B}_{2\text{-}2}(h, \cdot)$.

Proof Lemma 10 is proven by the similar manner to the proof of Lemma 8. □

Proof of Lemma 11

Lemma 11 *For any machine $\mathcal{A}$, for any security parameter* λ, $|\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}\nu\text{-}3)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)| \leq 1/q$.

Proof Lemma 11 is proven by the same manner as the proof of Lemma 7 in [25]. □

Proof of Lemma 12

Lemma 12 *For any machine $\mathcal{A}$, for any security parameter* λ, $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda) = 0$.

Proof The value of b is independent from the adversary's view in Game 3. Hence, $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda) = 0$. □

Proof of Lemma 13 in Sect. 8

Lemma 13 *For any machine $\mathcal{A}$, for any security parameter λ,* $|\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}\nu)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)| \leq 1/q$.

Proof To prove Lemma 13, we will show distribution $(\mathsf{param}_{\mathbb{V}}, \widehat{\mathbb{B}}, \{\boldsymbol{k}^{(j)*}\}_{j=1,\ldots,\nu}, \boldsymbol{c}, c_3)$ in Game 2-ν and that in Game 3 are equivalent (see Remark 9). For that purpose, we define new bases $\mathbb{D}$ of $\mathbb{V}$ and $\mathbb{D}^*$ of $\mathbb{V}^*$ as follows:

We generate random $\theta \xleftarrow{\mathsf{U}} \mathbb{F}_q$, and set

$$\boldsymbol{d}_{2n} := \boldsymbol{b}_{2n} - \theta \boldsymbol{b}_0, \quad \boldsymbol{d}_0^* := \boldsymbol{b}_0^* + \theta \boldsymbol{b}_{2n}^*,$$
$$\mathbb{D} := (\boldsymbol{b}_0, \ldots, \boldsymbol{b}_{2n-1}, \boldsymbol{d}_{2n}, \boldsymbol{b}_{2n+1}, \ldots, \boldsymbol{b}_{4n}), \quad \mathbb{D}^* := (\boldsymbol{d}_0^*, \boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_{4n}^*).$$

We then easily verify that $\mathbb{D}$ and $\mathbb{D}^*$ are dual orthonormal, and are distributed the same as the original bases, $\mathbb{B}$ and $\mathbb{B}^*$.

Keys and challenge ciphertext $(\{\boldsymbol{k}^{(j)*}\}_{j=1,\ldots,\nu}, \boldsymbol{c}, c_3)$ in Game 2-ν are expressed over bases $(\mathbb{B}, \mathbb{B}^*)$ and $(\mathbb{D}, \mathbb{D}^*)$ as

$$\boldsymbol{k}^{(j)*} = (\ 1,\ \delta^{(j)}\vec{v}^{(j)},\ \vec{w}^{(j)},\ \varphi^{(j)}\vec{v}^{(j)},\ 0^n\ )_{\mathbb{B}^*} = (\ 1,\ \delta^{(j)}\vec{v}^{(j)},\ \vec{\gamma}^{(j)},\ \varphi^{(j)}\vec{v}^{(j)},\ 0^n\ )_{\mathbb{D}^*}$$
$$\boldsymbol{c} = (\ \zeta,\ \omega\vec{x},\ \vec{r},\ 0^n,\ \vec{\eta}\ )_{\mathbb{B}} = (\ \zeta',\ \omega\vec{x},\ \vec{r},\ 0^n,\ \vec{\eta}\ )_{\mathbb{B}}$$
$$c_3 := g_T^{\zeta} m^{(b)}.$$

where

$$\vec{r} := p_0\vec{x} + p_1\vec{e}_n \text{ with } p_0, p_1 \xleftarrow{\mathsf{U}} \mathbb{F}_q, \quad \vec{\gamma}^{(j)} := \vec{w}^{(j)} - \theta\vec{e}_n, \quad \zeta' := \zeta + p_1\theta.$$

$\vec{\gamma}^{(j)}$ and ζ' are uniformly, independently distributed since $\vec{w}^{(j)} \xleftarrow{\mathsf{U}} \mathbb{F}_q^n$ and $\theta \xleftarrow{\mathsf{U}} \mathbb{F}_q$, except for the case $p_1 = 0$, i.e., except with the probability $1/q$.

In the light of the adversary's view, both $(\mathbb{B}, \mathbb{B}^*)$ and $(\mathbb{D}, \mathbb{D}^*)$ are consistent with public key $\mathsf{pk} := (1^\lambda, \mathsf{param}_{\mathbb{V}}, \widehat{\mathbb{B}})$. Therefore, $\{\boldsymbol{k}^{(j)*}\}_{j=1,\ldots,\nu}$ and $\boldsymbol{c}$ above can be expressed as keys and ciphertext in two ways, in Game 2-ν over bases $(\mathbb{B}, \mathbb{B}^*)$ and in Game 3 over bases $(\mathbb{D}, \mathbb{D}^*)$. Thus, Game 2-$\nu$ can be conceptually changed to Game 3. □

Proof of Lemma 14 in Sect. 9

Lemma 14 *For any machine $\mathcal{A}$, for any security parameter λ,* $|\mathsf{Adv}_{\mathcal{A}}^{(2\text{-}\nu)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)| \leq 1/q$.

Proof To prove Lemma 14, we will show distribution $(\mathsf{param}_{\mathbb{V}}, \widehat{\mathbb{B}}, \{\boldsymbol{k}^{(j)*}\}_{j=1,\ldots,\nu}, \boldsymbol{c}, c_3)$ in Game 2-ν and that in Game 3 are equivalent.

For that purpose, we define new bases $\mathbb{D}$ of $\mathbb{V}$ and $\mathbb{D}^*$ of $\mathbb{V}^*$ as follows:

$$\text{We generate } F := \begin{pmatrix} u & & u_1' \\ & \ddots & \vdots \\ & & u \ u_{n-1}' \\ & & u_n' \end{pmatrix} \xleftarrow{\mathsf{U}} \mathcal{H}(n, \mathbb{F}_q), \theta \xleftarrow{\mathsf{U}} \mathbb{F}_q, \text{ and set}$$

$$\boldsymbol{d}_{n+i} := \boldsymbol{b}_{n+i} - u\boldsymbol{b}_i \ \text{ for } i = 1, \ldots, n-1, \quad \boldsymbol{d}_{2n} := \boldsymbol{b}_{2n} - \theta\boldsymbol{b}_0 - \textstyle\sum_{\iota=1}^{n} u_\iota' \boldsymbol{b}_\iota$$
$$\boldsymbol{d}_0^* := \boldsymbol{b}_0^* + \theta\boldsymbol{b}_{2n}^*, \ \ \boldsymbol{d}_i^* := \boldsymbol{b}_i^* + u\boldsymbol{b}_{n+i}^* + u_i'\boldsymbol{b}_{2n}^* \ \text{ for } i = 1, \ldots, n-1, \ \ \boldsymbol{d}_n^* := \boldsymbol{b}_n^* + u_n'\boldsymbol{b}_{2n}^*$$

Let

$$\begin{aligned}&\vec{\boldsymbol{b}}_1 : = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)^{\mathrm{T}},\ \vec{\boldsymbol{b}}_2 := (\boldsymbol{b}_{n+1}, \ldots, \boldsymbol{b}_{2n})^{\mathrm{T}},\ \vec{\boldsymbol{b}}_1^* := (\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_n^*)^{\mathrm{T}},\ \vec{\boldsymbol{b}}_2^*\\ &\quad := (\boldsymbol{b}_{n+1}^*, \ldots, \boldsymbol{b}_{2n}^*)^{\mathrm{T}},\\ &\vec{\boldsymbol{d}}_2 : = (\boldsymbol{d}_{n+1}, \ldots, \boldsymbol{d}_{2n})^{\mathrm{T}},\ \vec{\boldsymbol{d}}_1^* := (\boldsymbol{d}_1^*, \ldots, \boldsymbol{d}_n^*)^{\mathrm{T}},\ \vec{\theta} := (0, \ldots, 0, \theta) \in \mathbb{F}_q^n.\end{aligned}$$

That is,

$$\begin{pmatrix} \boldsymbol{b}_0 \\ \vec{\boldsymbol{b}}_1 \\ \vec{\boldsymbol{d}}_2 \end{pmatrix} := \begin{pmatrix} 1 & 0 & 0 \\ 0 & I_n & 0_n \\ -\vec{\theta}^{\mathrm{T}} & -F^{\mathrm{T}} & I_n \end{pmatrix} \begin{pmatrix} \boldsymbol{b}_0 \\ \vec{\boldsymbol{b}}_1 \\ \vec{\boldsymbol{b}}_2 \end{pmatrix},$$

$$\begin{pmatrix} \boldsymbol{d}_0^* \\ \vec{\boldsymbol{d}}_1^* \\ \vec{\boldsymbol{b}}_2^* \end{pmatrix} := \begin{pmatrix} 1 & 0 & \vec{\theta} \\ 0 & I_n & F \\ 0 & 0_n & I_n \end{pmatrix} \begin{pmatrix} \boldsymbol{b}_0^* \\ \vec{\boldsymbol{b}}_1^* \\ \vec{\boldsymbol{b}}_2^* \end{pmatrix}.$$

We set

$$\mathbb{D} := (\boldsymbol{b}_0, \ldots, \boldsymbol{b}_n, \boldsymbol{d}_{n+1}, \ldots, \boldsymbol{d}_{2n}, \boldsymbol{b}_{2n+1}, \ldots, \boldsymbol{b}_{4n}), \quad \mathbb{D}^* := (\boldsymbol{d}_0^*, \ldots, \boldsymbol{d}_n^*, \boldsymbol{b}_{n+1}^*, \ldots, \boldsymbol{b}_{4n}^*).$$

We then easily verify that $\mathbb{D}$ and $\mathbb{D}^*$ are dual orthonormal, and are distributed the same as the original bases, $\mathbb{B}$ and $\mathbb{B}^*$.

Keys and challenge ciphertext $(\{\boldsymbol{k}^{(j)*}\}_{j=1,\ldots,\nu}, \boldsymbol{c}, c_3)$ in Game 2-ν are expressed over bases $\mathbb{B}$ and $\mathbb{B}^*$ as

$$\begin{aligned}&\boldsymbol{k}^{(j)*} = (\ 1,\ \delta^{(j)}\vec{v}^{(j)},\ \vec{w}^{(j)},\ \varphi^{(j)}\vec{v}^{(j)},\ 0^n\ )_{\mathbb{B}^*} = (\ 1,\ \delta^{(j)}\vec{v}^{(j)},\ \vec{\gamma}^{(j)},\ \varphi^{(j)}\vec{v}^{(j)},\ 0^n\ )_{\mathbb{D}^*},\\ &\boldsymbol{c} = (\ \zeta,\ \omega\vec{x},\ \vec{r},\ 0^n,\ \vec{\eta}\ )_{\mathbb{B}} = (\ \zeta', \vec{x}',\ \vec{r},\ 0^n,\ \vec{\eta}\ )_{\mathbb{D}}\\ &c_3 : = g_T^{\zeta} m^{(b)},\end{aligned}$$

where

$$\begin{aligned}&\vec{\gamma}^{(j)} : = \vec{w}^{(j)} - (\theta - u\delta^{(j)} v_n^{(j)} + \delta^{(j)} \textstyle\sum_{\iota=1}^n v_\iota^{(j)} u'_\iota)\vec{e}_n - u\delta^{(j)}\vec{v}^{(j)}\\ &\zeta' : = \zeta + \theta r_n,\ \ \vec{x}' := \omega\vec{x} + r_n\vec{u}' + u\vec{r}.\end{aligned}$$

$\vec{\gamma}^{(j)} \in \mathsf{span}\langle\vec{v}^{(j)}, \vec{e}_n\rangle, \zeta' \in \mathbb{F}_q, \vec{x}' \in \mathbb{F}_q^n$ are uniformly, independently distributed since $\vec{w}^{(j)} \overset{\mathsf{U}}{\leftarrow} \mathsf{span}\langle\vec{v}^{(j)}, \vec{e}_n\rangle, \theta \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q, \vec{u}' := (u'_1, \ldots, u'_n) \overset{\mathsf{U}}{\leftarrow} \mathbb{F}_q^n$ except for the case $r_n = 0$, i.e., except with the probability $1/q$.

In the light of the adversary's view, both $(\mathbb{B}, \mathbb{B}^*)$ and $(\mathbb{D}, \mathbb{D}^*)$ are consistent with public key $\mathsf{pk} := (1^\lambda, \mathsf{param}_{\mathbb{V}}, \widehat{\mathbb{B}})$. Therefore, $\{\boldsymbol{k}^{(j)*}\}_{j=1,\ldots,\nu}$ and $\boldsymbol{c}$ above can be expressed as keys and ciphertext in two ways, in Game 2-ν over bases $(\mathbb{B}, \mathbb{B}^*)$ and in Game 3 over bases $(\mathbb{D}, \mathbb{D}^*)$. Thus, Game 2-$\nu$ can be conceptually changed to Game 3. □

References

1. Abdalla M., Kiltz E., Neven G.: Generalized key delegation for hierarchical identity-based encryption. In: Biskup J., Lopez J. (eds.) ESORICS 2007. Lecture Notes in Computer Science, vol. 4734, pp. 139–154. Springer, Berlin (2007).
2. Agrawal S., Gorbunov S., Vaikuntanathan V., Wee H.: Functional encryption: new perspectives and lower bounds. In: CRYPTO 2013, pp. 500–518. Springer, Berlin (2013).
3. Ananth P., Boneh D., Garg S., Sahai A., Zhandry M.: Differing-inputs obfuscation and applications. In: IACR Cryptology ePrint Archive, vol. 2013, p. 689 (2013).

4. Attrapadung N., Libert B.: Functional encryption for inner product: achieving constant-size ciphertexts with adaptive security or support for negation. In: Nguyen P.Q., Pointcheval D. (eds.) PKC 2010. Lecture Notes in Computer Science, vol. 6056, pp. 384–402. Springer, Berlin (2010).
5. Attrapadung N., Libert B., de Panafieu E.: Expressive key-policy attribute-based encryption with constant-size ciphertexts. In: Catalano D., Fazio N., Gennaro R., Nicolosi A. (eds.) PKC 2011. Lecture Notes in Computer Science, vol. 6571, pp. 90–108. Springer, Berlin (2011).
6. Bethencourt J., Sahai A., Waters B.: Ciphertext-policy attribute-based encryption. In: IEEE Symposium on Security and Privacy, pp. 321–334. IEEE Computer Society Press, Washington, DC (2007).
7. Boneh D., Canetti R., Halevi S., Katz J.: Chosen-ciphertext security from identity-based encryption. SIAM J. Comput. **36**(5),1301–1328 (2007).
8. Boneh D., Hamburg M.: Generalized identity based and broadcast encryption schemes. In: Pieprzyk J. (ed.) ASIACRYPT 2008. Lecture Notes in Computer Science, vol. 5350, pp. 455–470. Springer, Berlin (2008).
9. Boneh D., Sahai A., Waters B.: Functional encryption: definitions and challenges. In: TCC 2011. Lecture Notes in Computer Science, vol. 6597, pp. 253–273. Springer, Heidelberg (2011).
10. Boyle E., Chung K.-M., Pass R.: On extractability obfuscation. In: IACR Cryptology ePrint Archive, vol. 2013, p. 650 (2013).
11. Chen J., Wee H.: Dual system groups and its applications—compact HIBE and more. In: IACR Cryptology ePrint Archive, vol. 2014, p. 265 (2014). (A preliminary version appeared at CRYPTO 2013)
12. Delerablée, C.: Identity-based broadcast encryption with constant size ciphertexts and private keys. In: Kurosawa K. (ed.) ASIACRYPT 2007. Lecture Notes in Computer Science, vol. 4833, pp. 200–215. Springer, Berlin (2007).
13. Garg S., Gentry C., Halevi S., Sahai A., Waters, B.: Attribute-based encryption for circuits from multilinear maps. In: CRYPTO 2013, Part II. Lecture Notes in Computer Science, vol. 8043, pp. 479–499. Springer, Heidelberg (2013).
14. Garg S., Gentry C., Halevi S., Zhandry M.: Fully secure attribute based encryption from multilinear maps. In: IACR Cryptology ePrint Archive, vol. 2014, p. 622 (2014).
15. Gentry C., Silverberg A.: Hierarchical id-based cryptography. In: Zheng Y. (ed.) ASIACRYPT 2002. Lecture Notes in Computer Science, vol. 2501, pp. 548–566. Springer, Heidelberg (2002).
16. Gentry C., Waters B.: Adaptive security in broadcast encryption systems (with short ciphertexts). In: Joux A. (ed.) EUROCRYPT 2009. Lecture Notes in Computer Science, vol. 5479, pp. 171–188. Springer, Heidelberg (2009).
17. Gorbunov S., Vaikuntanathan V., Wee H.: Attribute-based encryption for circuits. In: IACR Cryptology ePrint Archive, vol. 2013, p. 337 (2013).
18. Goyal V., Pandey O., Sahai A., Waters B.: Attribute-based encryption for fine-grained access control of encrypted data. In: Juels A., Wright R.N., di Vimercati D.C.S. (eds.) ACM Conference on Computer and Communications Security, pp. 89–98. ACM, New York (2006).
19. Katz J., Sahai A., Waters B.: Predicate encryption supporting disjunctions, polynomial equations, and inner products. In: Smart N.P. (ed.) EUROCRYPT 2008. Lecture Notes in Computer Science, vol. 4965, pp. 146–162. Springer, Heidelberg (2008).
20. Lewko A.B., Okamoto T., Sahai A., Takashima K., Waters B.: Fully secure functional encryption: attribute-based encryption and (hierarchical) inner product encryption. In: Gilbert H. (ed.) EUROCRYPT 2010. Lecture Notes in Computer Science, vol. 6110, pp. 62–91. Springer, Heidelberg (2010). http://eprint.iacr.org/2010/110.
21. Lewko A.B., Sahai A., Waters B.: Revocation systems with very small private keys. In: IEEE Symposium on Security and Privacy, pp. 273–285. IEEE Computer Society Press, Washington, DC (2010).
22. Lewko A.B., Waters B.: New techniques for dual system encryption and fully secure HIBE with short ciphertexts. In: Micciancio D. (ed.) TCC 2010. Lecture Notes in Computer Science, vol. 5978, pp. 455–479. Springer, Heidelberg (2010).
23. Okamoto T., Takashima K.: Homomorphic encryption and signatures from vector decomposition. In: Galbraith S.D., Paterson K.G. (eds.) Pairing 2008. Lecture Notes in Computer Science, vol. 5209, pp. 57–74. Springer, Heidelberg (2008).
24. Okamoto T., Takashima K.: Hierarchical predicate encryption for inner-products. In: Matsui M. (ed.) ASIACRYPT 2009. Lecture Notes in Computer Science, vol. 5912, pp. 214–231. Springer, Heidelberg (2009).
25. Okamoto T., Takashima K.: Fully secure functional encryption with general relations from the decisional linear assumption. In: Rabin T. (ed.) CRYPTO 2010. Lecture Notes in Computer Science, vol. 6223, pp. 191–208. Springer, Heidelberg (2010). http://eprint.iacr.org/2010/563.

26. Okamoto T., Takashima K.: Achieving short ciphertexts or short secret-keys for adaptively secure general inner-product encryption. In: Lin D., Tsudik G., Wang X. (eds.) CANS 2011. Lecture Notes in Computer Science, vol. 7092, pp. 138–159. Springer, Heidelberg (2011).
27. Okamoto T., Takashima K.: Adaptively attribute-hiding (hierarchical) inner product encryption. In: Pointcheval D., Johansson T. (eds.) EUROCRYPT 2012. Lecture Notes in Computer Science, vol. 7237, pp. 591–608. Springer, Heidelberg (2012). http://eprint.iacr.org/2011/543.
28. O'Neill A.: Definitional issues in functional encryption. In: IACR Cryptology ePrint Archive vol. 2010, p. 556 (2010).
29. Sahai A., Waters B.: Fuzzy identity-based encryption. In: Cramer R. (ed.) EUROCRYPT 2005. Lecture Notes in Computer Science, vol. 3494, pp. 457–473. Springer, Heidelberg (2005).
30. Sakai R., Furukawa J.: Identity-based broadcast encryption. In: IACR Cryptology ePrint Archive, vol. 2007, p. 217 (2007).
31. Waters B.: Dual system encryption: realizing fully secure IBE and HIBE under simple assumptions. In: Halevi S. (ed.) CRYPTO 2009. Lecture Notes in Computer Science, vol. 5677, pp. 619–636. Springer, Heidelberg (2009).
32. Waters B.: A punctured programming approach to adaptively secure functional encryption. In: IACR Cryptology ePrint Archive, vol. 2014, p. 588 (2014).