J Cryptol (2019) 32:1491–1573
https://doi.org/10.1007/s00145-018-9305-0

Journal of
CRYPTOLOGY

Fully Secure Functional Encryption with a Large Class of Relations from the Decisional Linear Assumption*

Tatsuaki Okamoto
NTT, Musashino-shi, Japan
okamoto.tatsuaki@lab.ntt.co.jp

Katsuyuki Takashima
Mitsubishi Electric, Kamakura-shi, Japan
Takashima.Katsuyuki@aj.MitsubishiElectric.co.jp

Communicated by Jonathan Katz.

Received 3 July 2017 / Revised 18 July 2018
Online publication 4 December 2018

Abstract. This paper presents a fully secure (adaptively secure) practical functional encryption scheme for a large class of relations, that are specified by non-monotone access structures combined with inner-product relations. The security is proven under a standard assumption, the decisional linear assumption, in the standard model. Our scheme is constructed on the concept of dual pairing vector spaces and a hierarchical reduction technique on this concept is employed for the security proof. The proposed functional encryption scheme covers, as special cases, (1) key-policy, ciphertext-policy and unified-policy attribute-based encryption with non-monotone access structures, (2) (hierarchical) attribute-hiding functional encryption with inner-product relations and functional encryption with nonzero inner-product relations and (3) spatial encryption and a more general class of encryption than spatial encryption.

Keywords. Functional encryption, Attribute-based encryption, Inner-product predicate encryption, Adaptive security, Decisional linear assumption, Dual pairing vector spaces.

1. Introduction

1.1. *Background*

Although numerous encryption systems have been developed over several thousand years, any traditional encryption system before the 1970's had a great restriction on the relation between a ciphertext encrypted by an encryption key and the decryption key such that these keys should be equivalent. The innovative notion of public key cryptosystems

*The extended abstract of a preliminary version [37] was presented at Advances in Cryptology—CRYPTO 2010.

in the 1970's relaxed this restriction, where these keys differ and the encryption key can be published, but the decryption key is firmly related to the encryption key for the unique decryption of a ciphertext to its plaintext.

Recently, a new innovative class of encryption systems, *functional encryption* (FE), has been introduced [14,15,28,41,44], where a secret (decryption) key, sk_f, is associated with a function f, an input x (to f) is encrypted to a ciphertext $\mathsf{Enc}(\mathsf{pk}, x)$ using system (master) public key pk, and the ciphertext is decrypted by the secret to $f(x)$.

This notion provides more sophisticated and flexible relations between decryption keys and ciphertexts such that a secret key, sk_Ψ, is associated with a parameter, Ψ, and message m is encrypted to a ciphertext $\mathsf{Enc}(\mathsf{pk}, (m, \Upsilon))$ using system public key pk along with another parameter Υ. Ciphertext $\mathsf{Enc}(\mathsf{pk}, (m, \Upsilon))$ can be decrypted by secret sk_Ψ if and only if a relation (predicate) $R(\Psi, \Upsilon)$ holds. Here, $x := (m, \Upsilon)$ is an input to encryption of FE and the function $f_{R,\Psi}$ (with secret key sk_Ψ) of $x := (m, \Upsilon)$ is m if and only if a relation $R(\Psi, \Upsilon)$ holds. Such a concept of FE has various applications in the areas of access control for databases, mail services, and contents distribution [5,12,15,28,30,42–45,48].

When R is the simplest relation or equality relation, i.e., $R(\Psi, \Upsilon)$ holds iff $\Psi = \Upsilon$, it is *identity-based encryption* (IBE) [6–8,10,16,21,24,25].

As a more general class of FE, *attribute-based encryption* (ABE) schemes have been proposed [5,12,15,28,30,42–45,48], where either one of the parameters for encryption and secret key is a tuple of attributes, and the other is a policy on attributes. Here each attribute is an element of a finite field or ring. For example, a policy Ψ is an access structure $\hat{M}$ along with a tuple of attributes $(v_1, \ldots, v_\iota)$ for a secret key, and a tuple of attributes, $\Upsilon := (x_1, \ldots, x_\iota)$, for encryption. Here, some elements of the tuples may be empty. $R(\Psi, \Upsilon)$ holds iff the truth-value vector of $(\mathsf{T}(x_1 = v_1), \ldots, \mathsf{T}(x_\iota = v_\iota))$ is accepted by $\hat{M}$, where $\mathsf{T}(\cdot)$ is a predicate such that $\mathsf{T}(\psi) := 1$ if ψ is true, and $\mathsf{T}(\psi) := 0$ if ψ is false (For example, $\mathsf{T}(x = v) := 1$ if $x = v$, and $\mathsf{T}(x = v) := 0$ if $x \neq v$). A monotone general access structure can express any monotone formula over atomic terms of $\mathsf{T}(x_1 = v_1), \ldots, \mathsf{T}(x_\iota = v_\iota)$. If parameter Ψ for a secret key is an access structure (policy), it is called key-policy ABE (KP-ABE). If parameter Υ for encryption is a policy, it is ciphertext-policy ABE (CP-ABE).

Inner-product predicate encryption (IPE) [30] is a class of FE for inner-product relations (predicates), where each parameter for encryption and secret key is a vector over a field or ring (e.g., $\vec{x} := (x_1, \ldots, x_n) \in \mathbb{F}_q^n$ and $\vec{v} := (v_1, \ldots, v_n) \in \mathbb{F}_q^n$ for encryption and secret key, respectively), and $R(\vec{v}, \vec{x})$ holds iff $\vec{x} \cdot \vec{v} = 0$, where $\vec{x} \cdot \vec{v}$ is the inner-product of $\vec{x}$ and $\vec{v}$. The inner-product relation represents a wide class of relations including equality, conjunction and disjunction (more generally, CNF and DNF) of equality relations and polynomial relations.

There are two types of secrecy on ciphertexts in FE, *attribute-hiding* (private-index) and *payload-hiding* (public-index) [30]. Roughly speaking, attribute-hiding requires that a ciphertext conceal the associated parameter as well as the plaintext, while payload-hiding only requires that a ciphertext conceal the plaintext. *Anonymous* IBE and *hidden-vector encryption* (HVE) [15] are a special class of attribute-hiding IPE.

Although many practical FE schemes such as ABE and IPE schemes have been presented over the last decade, existing fully secure (adaptively secure) practical FE schemes

only support some restricted classes of relations, e.g., monotone access structures with equality relations, and inner-product relations.

1.2. *Our Result*

In this paper, we propose fully secure practical FE schemes that supports more general relations than monotone access structures with equality relations and inner-product relations. Our scheme is secure in the standard assumption, the decisional linear (DLIN) assumption (over any type of prime-order bilinear groups), in the standard model.

More precisely, this paper presents a fully secure (adaptively secure against CPA) practical FE scheme for a large class of relations, that are specified by *non-monotone* access structures combined with *inner-product* relations. Similarly to the existing ABE schemes, we propose three types of FE schemes, the KP-FE and CP-FE schemes (in Sects. 4, 5) as well as a generalized notion of KP-FE and CP-FE, unified-policy FE (UP-FE).[1] (in Sect. 6).

In our KP-FE scheme, parameter Υ for a ciphertext is a tuple of (attribute) vectors and parameter Ψ for a secret key is a non-monotone access structure or span program $\hat{M} := (M, \rho)$ along with a tuple of vectors, e.g., $\Upsilon := (\vec{x}_1, \ldots, \vec{x}_\iota) \in \mathbb{F}_q^{n_1+\cdots+n_\iota}$, and $\Psi := (\hat{M}, (\vec{v}_1, \ldots, \vec{v}_\iota) \in \mathbb{F}_q^{n_1+\cdots+n_\iota})$. The component-wise inner-product relations for attribute vector components, e.g., $\{\vec{x}_t \cdot \vec{v}_t = 0 \text{ or not }\}_{t\in\{1,\ldots,\iota\}}$, are input to (non-monotone/monotone) span program $\hat{M}$, and $R(\Psi, \Upsilon)$ holds iff the truth-value vector of $(\mathsf{T}(\vec{x}_1 \cdot \vec{v}_1 = 0), \ldots, \mathsf{T}(\vec{x}_\iota \cdot \vec{v}_\iota = 0))$ is accepted by span program $\hat{M}$.

The proposed FE scheme is practical. For example, if the proposed FE scheme is specialized to IPE, the ciphertext size of our IPE scheme ("Appendix F.2") is $(3n+2)\cdot|\mathbb{G}|$, whose information theoretical lower bound is $n \cdot |\mathbb{F}_q|$ if the vector elements are from $\mathbb{F}_q$. Here, n is the dimension of the attribute vectors, and $|\mathbb{G}|$ and $|\mathbb{F}_q|$ denote the sizes of an element of prime order pairing group $\mathbb{G}$ (for ciphertexts) and finite field $\mathbb{F}_q$, respectively, e.g., both are 256 bits. Then, the ciphertext size of our IPE scheme is just around three times longer than the theoretical lower bound.

It is easy to convert the (CPA-secure) proposed FE scheme to a CCA-secure FE scheme by employing an existing general conversion such as that by Canetti et al. [17] or that by Boneh and Katz [13] (using additional seven-dimensional dual spaces $(\mathbb{B}_{d+1}, \mathbb{B}^*_{d+1})$ with $n_{d+1} := 2$ on the proposed FE scheme, and a strongly unforgeable one-time signature scheme or message authentication code with encapsulation) (see Sect. 7).

Since the proposed FE scheme supports a large class of relations, it includes the following schemes as special cases:

1. The (KP, CP and UP)-ABE schemes for *non-monotone* access structures with equality relations. Here, the underlying vectors of our FE scheme, $\{\vec{x}_t\}_{t\in\{1,\ldots,d\}}$ and $\{\vec{v}_t\}_{t\in\{1,\ldots,d\}}$, are specialized to two-dimensional vectors for the equality relation, e.g., $\vec{x}_t := (1, x_t)$ and $\vec{v}_t := (v_t, -1)$, where $\vec{x}_t \cdot \vec{v}_t = 0$ iff $x_t = v_t$ (see "Appendix F.1" for KP-ABE).

   In these ABE schemes, attribute x_t is expressed by the form of (t, x_t) in place of just attribute x_t. Here, t identifies a subuniverse or category of attributes, and x_t is

[1] The notion of UP-ABE and the first UP-ABE scheme were proposed by Attrapadung and Imai [3]

an attribute in subuniverse t (examples of (t, x_t) are (Name, Alice) and (Affiliation, Institute X)). The number of subuniverses, d, is a polynomial of security parameter λ, and the number of attributes in a subuniverse is exponential in λ.

2. The (zero-)IPE and nonzero-IPE schemes, where a nonzero-IPE scheme is a class of FE with $R(\vec{v}, \vec{x})$ iff $\vec{x} \cdot \vec{v} \neq 0$. Here, the underlying access structure $\mathbb{S}$ of our FE scheme is specialized to the 1-out-of-1 secret sharing.

   See "Appendix F.2" for our IPE scheme, which is slightly modified from a straightforward IPE-specialization of our FE scheme for improving efficiency. Note that the IPE scheme is 'weakly attribute-hiding,' where a type of key queries are not allowed in 'weakly attribute-hiding' (see the definition in [32]). It is easy to modify this IPE scheme to a 'fully attribute-hiding ([30])' scheme by simply expanding the dimension of the space [38], while its security proof is quite different from that shown in "Appendix F.2" (see [38] for the security proof of fully attribute-hiding).
3. If the underlying access structure is specialized to the d-out-of-d secret sharing (conjunction formula), our FE scheme can be specialized to a *hierarchical* zero/nonzero-IPE scheme by adding delegation and re-randomization mechanisms. We show two hierarchical (zero-)IPE (HIPE) schemes in "Appendix G", where one is payload-hiding and the other (weakly) attribute-hiding.
4. If the underlying access structure is a *monotone* formula with n-dimensional vectors, our FE scheme can be specialized to *spatial encryption* (for n-dimensional spaces) [12,19].

   Here, we give some simple examples.

   - Let A be a s-dimensional subspace in the n-dimensional vector space V ($0 < s < n$), which can be characterized by $(n - s)$ independent vectors in V, $(\vec{v}_1, .., \vec{v}_{n-s})$, such that $\vec{v}_i$ is orthogonal to A for all $i = 1, .., n - s$.

     We construct a spatial encryption (SE) scheme from our KP-FE scheme such that a secret key with subspace A, sk_A, is realized by the $(n-s)$-out-of-$(n-s)$ secret sharing (i.e., conjunction formula) along with $(\vec{v}_1, .., \vec{v}_{n-s})$. A ciphertext is associated with a vector $\vec{x} \in V$ and message m, i.e., $\mathsf{ct}_{(m,\vec{x})} := \mathsf{Enc}(\mathsf{pk}, (m, \vec{x}))$.

     The ciphertext $\mathsf{ct}_{(m,\vec{x})}$ can be decrypted to m by sk_A iff $\vec{x} \in A$, since $\vec{x} \in A$ iff $\bigwedge_{i=1}^{n-s} \vec{x} \cdot \vec{v}_i = 0$.
   - We can easily extend the above SE schemes with vector subspaces into SE schemes with affine subspaces. An affine subspace B can be expressed as $A + \vec{z}$, where A is a vector subspace in the n-dimensional vector space V, which is specified by orthogonal vectors $(\vec{v}_1, .., \vec{v}_{n-s})$, and $\vec{z}$ is an element in V. Hence, $\vec{x} \in B$ iff $\bigwedge_{i=1}^{n-s} (\vec{x} - \vec{z}) \cdot \vec{v}_i = 0$, i.e., $\bigwedge_{i=1}^{n-s} (\vec{x}, 1) \cdot (\vec{v}_i, -c_i) = 0$, where $c_i := \vec{z} \cdot \vec{v}_i$. We can then construct SE schemes with affine space B by replacing $\vec{x}$ and $\vec{v}_i$ in the above schemes by $(\vec{x}, 1)$ and $(\vec{v}_i, -c_i)$.

   These SE schemes using only conjunction formulas, which covers basic spacial encryption, can achieve the *attribute-hiding* in a manner similar to those for the (hierarchical) IPE schemes ("Appendix F.2, G").

5. If the underlying access structure is a *non-monotone* formula with n-dimensional vectors, our FE scheme can be a more general class of FE than spatial encryption.

   For example, let subspace A be defined by $(\vec{v}_1, .., \vec{v}_{n-s})$ in the same manner as above. Then, we can realize a FE scheme such that a ciphertext, $\mathsf{ct}_{(m,\vec{x})} := \mathsf{Enc}(\mathsf{pk}, (m, \vec{x}))$, can be decrypted to m by sk_A iff $\vec{x} \notin A$.

1.3. *Key Ideas and Techniques*

This section shows the key ideas and techniques in our result.

Since our scheme is constructed on the concept of dual pairing vector spaces (DPVS) [36], we first show the concept and main techniques of DPVS intuitively. We then show a key methodology to realize the *non-monotone* policy in our result. Finally, in this section, we describe how to achieve the adaptive security of our FE scheme in the DPVS framework.

1.3.1. *Concept of DPVS*

Roughly speaking, DPVS is an extension from bilinear pairing groups to higher-dimensional vector spaces, which are typically realized as direct products of bilinear pairing groups (or tuples of pairing group elements). Why is a vector space extension of pairing groups so useful for such applications?

There are two reasons. The first one is that the most natural methodology of constructing FE schemes on bilinear pairing groups is considered to realize them over the notion of vector spaces on pairing groups. Actually, many existing pairing-based schemes implicitly employ higher-dimensional vector spaces with using the form of computation like $\prod_{i=1}^{N} e(a_i, b_i)$, which is a pairing operation over higher-dimensional vector spaces (see 1. in Sect. 1.3.2), e.g., the Boneh–Boyen IBE schemes in decryption [6,7].

The second reason is that standard assumptions over pairing groups such as DDH and DLIN assumptions are *subspace* assumptions over vector spaces.

For example, the DDH assumption is a subspace assumption in a two-dimensional vector space (and DLIN is a subspace assumption in a three-dimensional vector space). The DDH assumption over a group $\mathbb{G}$ is expressed as given $\boldsymbol{x} := (g, g^a)$, and it is hard to tell $\boldsymbol{y} := (g^b, g^{ab})$ from $\boldsymbol{z} := (g^b, g^c)$, where $a, b, c \xleftarrow{\cup} \mathbb{F}_q$, $g \in \mathbb{G}$. (Note that when A is a set, $a \xleftarrow{\cup} A$ denotes that a is uniformly selected from A, and that $\mathbb{F}_q$ is the finite field of order q.) Here, $\boldsymbol{y}$ can be formalized as a scalar multiplication of $\boldsymbol{x}, b\boldsymbol{x}$, in a (two-dimensional) vector space. Since $b \xleftarrow{\cup} \mathbb{F}_q$, $\boldsymbol{y}$ is distributed over the (two-dimensional) subspace generated by $\boldsymbol{x}$, i.e., $\mathsf{span}\langle \boldsymbol{x} \rangle$. Since $b, c \xleftarrow{\cup} \mathbb{F}_q$, $\boldsymbol{z}$ is distributed over the whole (two-dimensional) vector space. Hence, the DDH problem is rephrased by one to tell $\boldsymbol{y}$ distributed over a one-dimensional subspace from $\boldsymbol{z}$ over the (two-dimensional) whole space.

We now briefly describe the concept of DPVS, that consists of vector space $\mathbb{V}$, pairing operation e over $\mathbb{V}$ and dual bases, $\mathbb{B}$ and $\mathbb{B}^*$. We start from a standard building block of (symmetric) pairing groups, $(\mathbb{G}, \mathbb{G}_T, g, q, e)$, where $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is a non-degenerate bilinear pairing operation, g is a generator of $\mathbb{G}$, q is a prime order of $\mathbb{G}$

and $\mathbb{G}_T$. Here, we denote the group operation of $\mathbb{G}$ and $\mathbb{G}_T$ by multiplication.[2] Note that DPVS is constructed over *asymmetric* pairing groups in general, although we use symmetric pairing groups here for simplicity of presentation.

Vector space: First, we construct an N-dimensional vector space $\mathbb{V}$ from group $\mathbb{G}$, where $\boldsymbol{x} \in \mathbb{V}$ is $(g_1, .., g_N) \in \mathbb{G}^N$. Vector additions and scalar multiplications over $\mathbb{V}$ are naturally introduced such that $\boldsymbol{x} + \boldsymbol{y} := (g_1 h_1, .., g_N h_N)$, and $a\boldsymbol{x} := (g_1^a, .., g_N^a)$, where $\boldsymbol{x} := (g_1, .., g_N)$, $\boldsymbol{y} := (h_1, .., h_N)$ and $a \in \mathbb{F}_q$. Note that a bold face letter denotes an element of vector space $\mathbb{V}$, e.g., $\boldsymbol{x} \in \mathbb{V}$.

Pairing operation: We naturally introduce the pairing operation $e : \mathbb{V} \times \mathbb{V} \to \mathbb{G}_T$ as $e(\boldsymbol{x}, \boldsymbol{y}) := \prod_{i=1}^N e(g^{x_i}, g^{y_i}) = e(g, g)^{\sum_{i=1}^N x_i y_i} = e(g, g)^{\vec{x} \cdot \vec{y}} \in \mathbb{G}_T$ for $\boldsymbol{x} := (g^{x_1}, .., g^{x_N}) \in \mathbb{V}$ and $\boldsymbol{y} := (g^{y_1}, .., g^{y_N}) \in \mathbb{V}$, where $\vec{x} := (x_1, .., x_N)$ and $\vec{y} := (y_1, .., y_N)$. Note that a vector symbol $\vec{x}$ denotes vector representation over $\mathbb{F}_q$, e.g., $\vec{x} := (x_1, \ldots, x_n) \in \mathbb{F}_q^n$, and $\vec{x} \cdot \vec{y}$ denotes the inner-product of $\vec{x}$ and $\vec{y}$ (in $\mathbb{F}_q$).

Bases: We then introduce a (random) basis $\mathbb{B} := (\boldsymbol{b}_1, \cdots, \boldsymbol{b}_N)$, of $\mathbb{V}$, using a uniformly chosen (regular) linear transformation, $X := (\chi_{i,j})_{i,j \in \{1,..,N\}} \xleftarrow{\cup} GF(N, \mathbb{F}_q)$, such that $\boldsymbol{b}_i := (g^{\chi_{i,1}}, \cdots, g^{\chi_{i,N}}) \in \mathbb{G}^N$ for $i = 1, .., N$. Here, $GL(N, \mathbb{F}_q)$ denotes the general linear group of degree N over $\mathbb{F}_q$.

We also compute another basis $\mathbb{B}^* := (\boldsymbol{b}_1^*, .., \boldsymbol{b}_N^*)$ of $\mathbb{V}$ by using $\alpha(X^{\mathrm{T}})^{-1}$ ($\alpha \in \mathbb{F}_q$) in place of X, where X^{T} denotes the transpose of X. Let $g_T := e(g, g)^{\alpha}$. We denote $(x_1, \ldots, x_N)_{\mathbb{B}} := \sum_{i=1}^N x_i \boldsymbol{b}_i$ and $(y_1, \ldots, y_N)_{\mathbb{B}^*} := \sum_{i=1}^N y_i \boldsymbol{b}_i^*$.

We then see that $e(\boldsymbol{b}_i, \boldsymbol{b}_j^*) = g_T^{\delta_{i,j}}$ for $i, j \in \{1, .., N\}$, where $\delta_{i,j} = 1$ if $i = j$ and $\delta_{i,j} = 0$ if $i \neq j$. That is, $\mathbb{B}$ and $\mathbb{B}^*$ are dual orthonormal bases of $\mathbb{V}$. Due to the orthonormality, for $\boldsymbol{x} := (\vec{x})_{\mathbb{B}}$ and $\boldsymbol{y} := (\vec{y})_{\mathbb{B}^*}$, pairing operation $e(\boldsymbol{x}, \boldsymbol{y}) = g_T^{\vec{x} \cdot \vec{y}}$, where $\vec{x} := (x_1, .., x_N)$ and $\vec{y} := (y_1, .., y_N)$.

In cryptographic applications of DPVS, (a part of) $\mathbb{B}$ is used as a public parameter (public key), $\mathbb{B}^*$ is used as a (master) secret key, and X is used as the top-level secret key. It is an advantage of this approach that we can make various levels/types of secret keys to meet the requirements on secret keys in applications, from the top level of secret key, X, to a lower level of secret key, which may be a form of partial information of $\mathbb{B}^*$.

1.3.2. *Properties of DPVS*

DPVS has the following properties that are useful for many applications:

1. Hard decomposability As mentioned above, vector treatment of bilinear pairing groups have been already developed and employed in the literature especially

[2] Only in Sect. 1.3, we express bilinear group $\mathbb{G}$ as a *multiplicative* group to follow the tradition of cryptocommunity, but in this paper except Sect. 1.3, we express it as an additive group for the consistency with the vector space expressions.

in the areas of IBE, ABE and BE (Broadcast Encryption) (e.g., [5,8,12,16,28,29,44]). For example, in a typical vector treatment of bilinear pairing groups, two forms of $X := (g^{x_1}, g^{x_2}, \ldots, g^{x_N})$ for vector $\vec{x} := (x_1, .., x_N)$, and $Y := (g^{y_1}, g^{y_2}, \ldots, g^{y_N})$ for vector $\vec{y} := (y_1, .., y_N)$ are set and pairing of X and Y is operated such that $e(X, Y) := \prod_{i=1}^{N} e(g^{x_i}, g^{y_i}) = e(g, g)^{\sum_{i=1}^{N} x_i y_i} = e(g, g)^{\vec{x} \cdot \vec{y}}$.

The major drawback of this approach is that it is easy to decompose x_i's element, g^{x_i}, from $X := (g^{x_1}, g^{x_2}, \ldots, g^{x_N})$.

In contrast, a remarkable property of DPVS over (random) basis $\mathbb{B}$ is that it seems hard to decompose x_i's element, $x_i \boldsymbol{b}_i$, from $\boldsymbol{x} := x_1 \boldsymbol{b}_1 + \cdots + x_N \boldsymbol{b}_N$ and $\mathbb{B}$. Here note that we can compute a value regarding $\vec{x} \cdot \vec{y}$ (corresponding to $e(g, g)^{\vec{x} \cdot \vec{y}}$ above) by the pairing operation of $\boldsymbol{x}$ and $\boldsymbol{y} := y_1 \boldsymbol{b}_1^* + \cdots + y_n \boldsymbol{b}_N^*$, i.e., $e(\boldsymbol{x}, \boldsymbol{y}) = g_T^{\vec{x} \cdot \vec{y}}$.

2. Information theoretically hidden subspaces Let $\mathbb{B} := (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_N)$ and $\mathbb{B}^* := (\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_N^*)$ be dual orthonormal bases with $X \xleftarrow{\cup} GL(N, \mathbb{F}_q)$. In many applications of DPVS, public parameters or (master) public key are $\hat{\mathbb{B}}$ that is a part of $\mathbb{B}$. For example, $\hat{\mathbb{B}} := (\boldsymbol{b}_1, .., \boldsymbol{b}_n)$, where $n < N$. Here note that $\boldsymbol{b}_{n+1}, .., \boldsymbol{b}_N$ are information theoretically hidden, since $X \xleftarrow{\cup} GF(N, \mathbb{F}_q)$ and bases $(\boldsymbol{b}_{n+1}, .., \boldsymbol{b}_N)$ are perfectly independently chosen from $(\boldsymbol{b}_1, .., \boldsymbol{b}_n)$. In addition, $\mathbb{B}^* := (\boldsymbol{b}_1^*, .., \boldsymbol{b}_N^*)$ can be also hidden as a secret key.

In the DPVS approach, we have developed several information theoretical transformation techniques based on this information theoretical property.

We will describe these techniques in Sect. 1.3.3.

3. Inner-product operability As mentioned above, for $\boldsymbol{x} := x_1 \boldsymbol{b}_1 + \cdots + x_N \boldsymbol{b}_N = (\vec{x})_{\mathbb{B}}$ and $\boldsymbol{y} := y_1 \boldsymbol{b}_1^* + \cdots + y_N \boldsymbol{b}_N^* = (\vec{y})_{\mathbb{B}^*}$, the inner-product value $\vec{x} \cdot \vec{y}$ is indirectly computed through the pairing computation, $e(\boldsymbol{x}, \boldsymbol{y}) = g_T^{\vec{x} \cdot \vec{y}}$.

Composite-order pairing groups are often employed to achieve the property 1. (Hard decomposability) [11,33,34]. An advantage of our DPVS approach over the composite-order pairing group approach is that our approach is realized on prime-order groups of any type (symmetric and asymmetric) and the implementations on prime-order groups are more efficient than those on composite-order groups. In addition, several non-standard computational assumptions are always used to prove the security in the composite-order group approach, while many schemes in our DPVS approach have been proven solely under the DLIN assumption.

Some conversion from composite-order group schemes to prime-order group schemes has been proposed based on our DPVS methodology [31], and it may lead to the thoughts that the whole properties of the DPVS approach would be achieved by this type of conversion, but it is not the case. Such conversion usually focuses on the property 1. but not on the property 2. (Information theoretically hidden subspaces) of DPVS.

1.3.3. *Key Techniques of DPVS*

By using the above-mentioned properties of DPVS, we have developed two key techniques on DPVS, one is a hierarchical reductions to DLIN (for computationally indistinguishable game changes) and the other information theoretical transformations (for conceptual game changes).

1. Hierarchical Reductions to DLIN In the hierarchical reduction methodology, the top level of the security proof for the proposed scheme directly employs only top level assumptions (assumptions of Problems 1 and 2 in this paper), that are specified in the DPVS framework. The methodology bridges the top-level assumptions and the primitive one, the DLIN assumption, in a hierarchical manner, where several levels of assumptions (problems) are constructed hierarchically. Such a modular way of proof greatly clarifies the logic of a complicated security proof. (See Fig. 1 for the global view of the methodology.)

- **Lower-level Reductions**
  The following basic (subspace) assumptions over the three-dimensional case on DPVS are reduced to the DLIN assumption.

  The DLIN assumption is that, given $(g, g^{\xi}, g^{\kappa}, g^{\delta\xi}, g^{\sigma\kappa}) \in \mathbb{G}^5$, it is hard to tell $g^{\delta+\sigma}$ from g^{γ}, where $\xi, \kappa, \delta, \sigma, \gamma \xleftarrow{\cup} \mathbb{F}_q$. Let $\mathbb{B} := (\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3)$ and $\mathbb{B}^* := (\boldsymbol{b}_1^*, \boldsymbol{b}_2^*, \boldsymbol{b}_3^*)$ be dual orthonormal bases with $X \xleftarrow{\cup} GL(3, \mathbb{F}_q)$ and $g_T := e(\boldsymbol{b}_i, \boldsymbol{b}_i^*) \in \mathbb{G}_T$ $(i = 1, 2, 3)$.

  Basic Problem 0 (Definition 18) assumption for ciphertexts: Let $\hat{\mathbb{B}}* := (\boldsymbol{b}_1^*, \boldsymbol{b}_2^*)$, $\boldsymbol{c}_0 := (\delta, \sigma, \boxed{0})_{\mathbb{B}}$ and $\boldsymbol{c}_1 := (\delta, \sigma, \boxed{\rho})_{\mathbb{B}}$, where $\delta, \sigma, \rho \xleftarrow{\cup} \mathbb{F}_q$. Then, given $(\hat{\mathbb{B}}*, \mathbb{B})$, it is hard to tell $\boldsymbol{c}_0$ from $\boldsymbol{c}_1$.

  Basic Problem 0 assumption for secret keys: Let $\hat{\mathbb{B}} := (\boldsymbol{b}_1, \boldsymbol{b}_2)$, $\boldsymbol{c} := (\omega, 0, \boxed{\tau})_{\mathbb{B}}$, $\boldsymbol{k}_0^* := (\delta, \sigma, \boxed{0})_{\mathbb{B}^*}$ and $\boldsymbol{k}_1^* := (\delta, \sigma, \boxed{\rho})_{\mathbb{B}^*}$, where $\delta, \sigma, \rho, \omega, \tau \xleftarrow{\cup} \mathbb{F}_q$. Then, given $(\hat{\mathbb{B}}, \mathbb{B}^*, \boldsymbol{c})$, it is hard to tell $\boldsymbol{k}_0^*$ from $\boldsymbol{k}_1^*$.

  In the reduction of these assumptions to DLIN, a DLIN instance $(g, g^{\xi}, g^{\kappa}, g^{\delta\xi}, g^{\sigma\kappa}, y_\beta) \in \mathbb{G}^6$ (where $\beta \in \{0, 1\}$, $y_0 = g^{\delta+\sigma}$ and $y_1 = g^{\gamma}$) is converted to an instance of Basic Problem 0 assumptions. First, we express the DLIN instance as a subspace assumption instance, $(\boldsymbol{u}_1 := (g^{\xi}, 1, g), \boldsymbol{u}_2 := (1, g^{\kappa}, g), \boldsymbol{u}_3 := (1, 1, g), \boldsymbol{w}_\beta)$ (where $\boldsymbol{w}_0 = \delta\boldsymbol{u}_1 + \sigma\boldsymbol{u}_2 = (g^{\delta\xi}, g^{\sigma\kappa}, y_0)$ and $\boldsymbol{w}_1 = \delta\boldsymbol{u}_1 + \sigma\boldsymbol{u}_2 + \rho\boldsymbol{u}_3 = (g^{\delta\xi}, g^{\sigma\kappa}, y_1)$ with $\rho := \gamma - (\delta + \sigma)$). Here $\mathbb{U} := (\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3)$ is a basis of DPVS, and the linear transformation Π to generate $\mathbb{U}$ and the adjoint matrix $(\Pi^*)^{\mathrm{T}}$ are

$$\Pi := \begin{pmatrix} \xi & & 1 \\ & \kappa & 1 \\ & & 1 \end{pmatrix}, \quad \Pi^* := \begin{pmatrix} \kappa & & \\ & \xi & \\ -\kappa & -\xi & \kappa\xi \end{pmatrix},$$

where $(\Pi^*)^{\mathrm{T}} = \kappa\xi \cdot \Pi^{-1}$ and a blank element in the matrices denotes 0. for $\mathbb{U}^* := (\boldsymbol{u}_1^* := (g^{\kappa}, 1, 1), \boldsymbol{u}_2^* := (1, g^{\xi}, 1), \boldsymbol{u}_3^* := (g^{-\kappa}, g^{-\xi}, g^{\kappa\xi}))$, the DPVS bases $\mathbb{U}$ and $\mathbb{U}^*$ are dual orthonormal bases with Π, and $g_T := e(g,g)^{\kappa\xi} = e(g^{\kappa}, g^{\xi})$. Therefore, a converted DLIN assumption on DPVS is that, given $(\boldsymbol{u}_1^*, \boldsymbol{u}_2^*, \mathbb{U})$, it is hard to tell $\boldsymbol{w}_0 := (\delta, \sigma, 0)_{\mathbb{U}}$ from $\boldsymbol{w}_1 := (\delta, \sigma, \rho)_{\mathbb{U}}$. Here note that $g^{\kappa\xi}$ is not included in the DLIN instance and $\boldsymbol{u}_3^*$ (with $g^{\kappa\xi}$) is not included in the above instance. Based on this type of conversion, the Basic Problem 0 assumptions can be reduced to DLIN by applying additional random linear transformation (by random matrix W) on a special form of orthonormal bases $\mathbb{U}$ and $\mathbb{U}^*$ to obtain random orthonormal bases $\mathbb{B}$ and $\mathbb{B}^*$ (Lemma 14).

- **Middle-Level Reductions**
  Here, we show some middle-level assumptions, (subspace) assumptions on higher-dimensional DPVS, which are simplified versions of Basic Problems 1 and 2 (Definitions 19, 20) assumptions.

  Simplified Version of Basic Problem 1 (Definition 19) assumption
  Let $\mathbb{B} := (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{3n+2})$ and $\mathbb{B}^* := (\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_{3n+2}^*)$ be dual orthonormal bases, $\widehat{\mathbb{B}}^* := (\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_{n+1}^*, \boldsymbol{b}_{n+3}^*, .., \boldsymbol{b}_{3n+2}^*)$, $\boldsymbol{c}_0 := (0, \delta\vec{e}_1, \boxed{0^n}, \sigma)_{\mathbb{B}}$, and $\boldsymbol{c}_1 := (0, \delta\vec{e}_1, \boxed{\rho\vec{e}_1}, \sigma)_{\mathbb{B}}$, where $\delta, \sigma, \rho \xleftarrow{\cup} \mathbb{F}_q$. Then, given $\mathbb{B}$ and $\widehat{\mathbb{B}}^*$, it is hard to tell $\boldsymbol{c}_0$ from $\boldsymbol{c}_1$.

  Simplified Version of Basic Problem 2 (Definition 20) assumption
  Let $\mathbb{B} := (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{3n+2})$ and $\mathbb{B}^* := (\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_{3n+2}^*)$ be dual orthonormal bases, $\widehat{\mathbb{B}} := (\boldsymbol{b}_1, .., \boldsymbol{b}_{n+1}, \boldsymbol{b}_{2n+2}, .., \boldsymbol{b}_{3n+2})$, $\boldsymbol{c}_i := (0, \omega\vec{e}_i, \boxed{\tau\vec{e}_i}, 0^n, 0)_{\mathbb{B}}$, $\boldsymbol{k}_{0,i}^* := (0, \delta\vec{e}_i, \boxed{0^n}, \vec{\eta}_i, 0)_{\mathbb{B}^*}$ and $\boldsymbol{k}_{1,i}^* := (0, \delta\vec{e}_i, \boxed{\rho\vec{e}_i}, \vec{\eta}_i, 0)_{\mathbb{B}^*}$, where $i = 1, .., n$, $\delta, \rho, \omega, \tau \xleftarrow{\cup} \mathbb{F}_q$ and $\vec{\eta}_i \xleftarrow{\cup} \mathbb{F}_q^n$. Then, given $\widehat{\mathbb{B}}$, $\mathbb{B}^*$ and $\{\boldsymbol{c}_i\}_{i=1,..,n}$, it is hard to tell $\{\boldsymbol{k}_{0,i}\}_{i=1,..,n}$ from $\{\boldsymbol{k}_{1,i}\}_{i=1,..,n}$. We then show the simplified version of Basic Problems 1 and 2 to Basic Problem 0 assumption, which implies the reduction of these assumptions to the DLIN assumption via the lowest level reduction (hierarchical reduction).

  - The simplified version of Basic Problem 1 can be expressed as $\boldsymbol{c}_0 := (0, \boxed{\delta}, 0^{n-1}, \boxed{0}, 0^{2n-1}, \boxed{\sigma})_{\mathbb{B}}$, and $\boldsymbol{c}_1 := (0, \boxed{\delta}, 0^{n-1}, \boxed{\rho}, 0^{2n-1}, \boxed{\sigma})_{\mathbb{B}}$. Hence, it can be reduced to Basic Problem 0 for ciphertexts by embedding the Basic Problem 0 instance into the $(3n+2)$-dimensional space.
  - The simplified version of Basic Problem 2 can be expressed as $\boldsymbol{c}_i := (0, 0^{i-1}, \boxed{\omega}, 0^{n-i}, 0^{i-1}, \boxed{\tau}, 0^{n-1}, \boxed{0^n}, 0)_{\mathbb{B}}, \boldsymbol{k}_{0,i}^* := (0, 0^{i-1}, \boxed{\delta}, 0^{n-i}, 0^{i-1}, \boxed{0}, 0^{n-i}, \boxed{\vec{\eta}}, 0)_{\mathbb{B}}$, and $\boldsymbol{k}_{1,i}^* := (0, 0^{i-1}, \boxed{\delta}, 0^{n-i}, 0^{i-1}, \boxed{\rho}, 0^{n-i}, \boxed{\vec{\eta}}, 0)_{\mathbb{B}}$. Hence, it can be reduced to Basic Problem 0 for secret keys by embedding the Basic Problem 0 instance into the $(3n+2)$-dimensional space, where the σ part of the Basic Problem 0 element is embedded into the η_i part with $(\eta_1, .., \eta_n) := \vec{\eta}$.

The reductions from Basic Problems 1 and 2 to Basic Problem 0 are essentially the same as the above-mentioned middle-level reduction except that Basic Problems 1

and 2 have multiple spaces on bases $(\mathbb{B}_t, \mathbb{B}_t^*)$ with $t = 0, 1, .., d$, while the simplified version of Basic Problems 1 and 2 are on $(\mathbb{B}, \mathbb{B}^*)$ (Lemmas 15, 17).

- **Higher-Level Reductions**
  Top-level assumptions, Problems 1 and 2 (Definitions 4, 5), are reduced to Basic Problems 1 and 2 by using *Intra-subspace* information theoretical transformation to be explained just below (see Lemmas 16, 18 for the reduction precisely).

  Problem 1 and 2 assumptions are used for computationally indistinguishable game changes of top level of security proof (full security proof of the proposed FE scheme).

  See Fig. 1 for the hierarchical structure of reductions.

2. Information theoretical transformations We have developed several information theoretical transformation techniques based on the property 2. of DPVS. There are two basic information theoretical techniques, *intra-subspace* and *inter-subspace* transformations, by the hidden base changes. Here we use the same example as that given in the property 2. of Sect. 1.3.2.

Intra-subspace transformation:
Hidden bases $(\boldsymbol{b}_{n+1}, .., \boldsymbol{b}_N)$ and $(\boldsymbol{b}^*_{n+1}, .., \boldsymbol{b}^*_N)$ are (conceptually) changed to $(\boldsymbol{d}_{n+1}, .., \boldsymbol{d}_N) := (\boldsymbol{b}_{n+1}, .., \boldsymbol{b}_N) \cdot (Z^{-1})^{\mathrm{T}}$, and $(\boldsymbol{d}^*_{n+1}, .., \boldsymbol{d}^*_N) := (\boldsymbol{b}^*_{n+1}, .., \boldsymbol{b}^*_N) \cdot Z^{\mathrm{T}}$, where $Z \in GL(N-n, \mathbb{F}_q)$. We then have new dual orthonormal bases of $\mathbb{V}$, $\mathbb{D} := (\boldsymbol{b}_1, .., \boldsymbol{b}_n, \boxed{\boldsymbol{d}_{n+1}, .., \boldsymbol{d}_N})$ and $\mathbb{D}^* := (\boldsymbol{b}^*_1, .., \boldsymbol{b}^*_n, \boxed{\boldsymbol{d}^*_{n+1}, .., \boldsymbol{d}^*_N})$. Then, ciphertext $\boldsymbol{c} := (\vec{\psi}_1, \vec{\psi}_2)_{\mathbb{B}}$ with $\vec{\psi}_i \in \mathbb{F}_q^n$ $(i = 1, 2)$ can be expressed by $(\vec{\psi}_1, \boxed{\vec{\psi}_2 \cdot Z})_{\mathbb{D}}$, and secret key $\boldsymbol{k}^* := (\vec{\xi}_1, \vec{\xi}_2)_{\mathbb{B}^*}$ with $\vec{\xi}_i \in \mathbb{F}_q^n$ $(i = 1, 2)$ can be by $(\vec{\xi}_1, \boxed{\vec{\xi}_2 \cdot (Z^{-1})^{\mathrm{T}}})_{\mathbb{D}^*}$.

As mentioned above, the intra-subspace transformation is employed to reduce Problem 1 and 2 assumptions to Basic Problems 1 and 2.

Inter-subspace transformation:
Hidden bases $(\boldsymbol{b}_{n+1}, .., \boldsymbol{b}_N)$ $(N = n + m)$ and $(\boldsymbol{b}^*_1, .., \boldsymbol{b}^*_n)$ are (conceptually) changed to $(\boldsymbol{d}_{n+1}, .., \boldsymbol{d}_N) := (\boldsymbol{b}_{n+1} - \sum_{j=1}^n f_{1,j}\boldsymbol{b}_j, .., \boldsymbol{b}_N - \sum_{j=1}^n f_{m,j}\boldsymbol{b}_j)$, and $(\boldsymbol{d}^*_1, .., \boldsymbol{d}^*_n) := (\boldsymbol{b}^*_1 + \sum_{i=1}^m f_{i,1}\boldsymbol{b}^*_{n+i}, .., \boldsymbol{b}^*_n + \sum_{i=1}^m f_{i,n}\boldsymbol{b}^*_{n+i}$, where $F := (f_{i,j}) \in \mathbb{F}_q^{m \times n}$. We then have new dual orthonormal bases of $\mathbb{V}$, $\mathbb{D} := (\boldsymbol{b}_1, .., \boldsymbol{b}_n, \boxed{\boldsymbol{d}_{n+1}, .., \boldsymbol{d}_N})$ and $\mathbb{D}^* := (\boxed{\boldsymbol{d}^*_1, .., \boldsymbol{d}^*_n}, \boldsymbol{b}^*_{n+1}, .., \boldsymbol{b}^*_N)$. Then, ciphertext $\boldsymbol{c} := (\vec{\psi}_1, \vec{\psi}_2)_{\mathbb{B}}$ can be expressed by $(\boxed{\vec{\psi}_1 + \vec{\psi}_2 \cdot F}, \vec{\psi}_2)_{\mathbb{D}}$, and secret key $\boldsymbol{k}^* := (\vec{\xi}_1, \vec{\xi}_2)_{\mathbb{B}^*}$ can be by $(\vec{\xi}_1, \boxed{\vec{\xi}_2 - \vec{\xi}_1 \cdot F^{\mathrm{T}}})_{\mathbb{D}^*}$.

The inter-subspace transformation is employed to prove the small advantage gaps between Game 2-ν and Game 3 in Fig. 1, where $F \xleftarrow{\cup} \mathbb{F}_q^{1\times 1}$ (a random scalar in $\mathbb{F}_q$) is employed. This transformation is also employed in the corresponding places in the security proof of Sects. F.2 and G , where more general forms of F are employed.

1.3.4. *Non-monotone Policy*

Non-monotone policies and predicates should be used in many FE applications. For example, an access policy (for a user) regarding a confidential audit report on 'K Institute' could be in the following form: NOT(Affiliation = 'K Institute') AND $(\cdots)$.

To achieve a non-monotone policy on attributes in universe $\mathcal{U}$, it is essentially required to introduce a concept of categories or subuniverses, where a category or subuniverse, $\mathcal{U}_t$ ($t \in \mathbb{N}$ is an identity of a category), is a subset of universe $\mathcal{U}$. In the above-mentioned example, a subset of affiliations, $\mathcal{U}_{\mathsf{affiliation}}$ is a category. Then, the policy on attribute X of a user is expressed as $(X \neq$ 'K Institute' $\wedge\, X \in \mathcal{U}_{\mathsf{affiliation}})$ AND $(\cdots)$.

Without such a notion of categories or subuniverses, a non-monotone policy cannot be correctly captured. For example, if a policy on attribute X is just $(X \neq$ 'K Institute') AND $(\cdots)$, any attribute (e.g., 'Professor', 'Male', and 'Japanese') different from 'K Institute' in any category satisfies the clause with substituting such an attribute to X . (A straightforward application of a monotone ABE scheme [42] may have this problem.)

This paper presents an elegant solution to this issue by using dual subspaces of DPVS without using an explicit formula such as $(\ldots \wedge\, X \in \mathcal{U}_{\mathsf{affiliation}})$. Here, an attribute is expressed by the form of (t, x_t) with $t \in T \subseteq \{1, \ldots, d\}$ in place of just an attribute x, where t identifies a subuniverse or category of attributes, and x_t is an attribute in subuniverse t (examples of (t, x_t) are ('Affiliation', 'K Institute'), ('Title', 'Professor'), ('Gender', 'Male') and ('Nationality', 'Japanese')).

In our scheme, each (t, x_t) is encoded as a value in a subspace, $\mathsf{span}\langle\mathbb{B}_t\rangle$, spanned by bases $\mathbb{B}_t$ (or $\mathbb{B}_t^*$) of DPVS, and a non-monotone policy on category t (e.g., $X_t \neq$ 'K Institute', $t =$ 'Affiliation') is also encoded in a subspace, $\mathsf{span}\langle\mathbb{B}_t^*\rangle$, spanned by bases $\mathbb{B}_t^*$ (or $\mathbb{B}_t$), where independent d bases $(\mathbb{B}_1, \ldots, \mathbb{B}_d)$ (and the dual bases, $(\mathbb{B}_1^*, \ldots, \mathbb{B}_d^*)$) are set up in our scheme.

Roughly speaking, only a value in $\mathsf{span}\langle\mathbb{B}_t\rangle$ can be correctly operated with a value in $\mathsf{span}\langle\mathbb{B}_t^*\rangle$. That is, only an attribute x_t encoded in $\mathsf{span}\langle\mathbb{B}_t\rangle$ can be correctly operated with a non-monotone policy on t (e.g., $X_t \neq$ 'K Institute') encoded in $\mathsf{span}\langle\mathbb{B}_t^*\rangle$.

This can be formally ensured in the security proof by the fact that the information theoretical transformation via hidden base changes is shared by $\mathsf{span}\langle\mathbb{B}_t\rangle$ and $\mathsf{span}\langle\mathbb{B}_t^*\rangle$, but it is perfectly independent from the other subspace spanned by different bases $\mathbb{B}_{t'}$ and $\mathbb{B}_{t'}^*$ with $t' \neq t$. In other words, the condition that $X \in \mathcal{U}_{\mathsf{affiliation}}$ is realized in the correct operation mechanism between corresponding dual subspaces, $\mathsf{span}\langle\mathbb{B}_t\rangle$ and $\mathsf{span}\langle\mathbb{B}_t^*\rangle$. Hence, a non-monotone policy on t, $X_t \neq$ 'K Institute' with $t =$ 'Affiliation', can be correctly operated with an attribute of ('Affiliation', *) encoded in $\mathsf{span}\langle\mathbb{B}_t^*\rangle$ but not with ('Title,' *) in $\mathsf{span}\langle\mathbb{B}_{t'}^*\rangle$, ('Gender', *) in $\mathsf{span}\langle\mathbb{B}_{t''}^*\rangle$, and ('Nationality', *) in $\mathsf{span}\langle\mathbb{B}_{t'''}^*\rangle$.

More precisely, in our scheme, vectors, $\vec{x}$ and $\vec{v}$, are employed in place of attributes, and each vector is categorized to a category or subuniverse, $\mathcal{U}_t$, i.e., vector $\vec{x}$ in $\mathcal{U}_t$ is expressed by the form of $(t, \vec{x})$ and encoded in $\mathsf{span}\langle\mathbb{B}_t\rangle$.

For example, in our KP-FE scheme, a ciphertext $\boldsymbol{c}$ with a n-dimensional vector $(t, \vec{x})$ is realized as the form of

$$\boldsymbol{c} := (\omega\vec{x}, 0^n, 0^n, \varphi)_{\mathbb{B}_t},$$

and a secret key $\boldsymbol{k}_i^*$ for the ith entry of a *negation* term of a span program (s_i is the corresponding share) associated with a vector $(t', \vec{v}_i)$ is of the form of

$$\boldsymbol{k}_i^* := (s_i \vec{v}_i, 0^{n_t}, \vec{\eta}_i, 0)_{\mathbb{B}_{t'}^*}.$$

Hence, in the decryption process,

$$e(\boldsymbol{c}, \boldsymbol{k}_i^*)^{1/\vec{x}\cdot\vec{v}_i} = g_T^{\omega s_i} \quad (\text{iff } t = t' \text{and } \vec{x} \cdot \vec{v}_i \neq 0).$$

That is, due to the decryption property and the above-mentioned property that only $\vec{x}$ encoded in $\mathsf{span}\langle\mathbb{B}_t\rangle$ can be correctly operated with $\vec{v}_i$ encoded in $\mathsf{span}\langle\mathbb{B}_t^*\rangle$, the ith share s_i of the span program is recovered iff $t = t'$ and $\vec{x} \cdot \vec{v}_i \neq 0$.

1.3.5. *Adaptive Security*

To achieve the adaptive security, this paper elaborately combines the dual system encryption technique proposed by Waters [49] and the DPVS methodology.

In the dual system encryption, roughly there are two forms of ciphertexts and secret keys, normal and semi-functional forms. One of the advantages of the DPVS methodology is that the two forms can be indistinguishable based on the above-mentioned Problems 1 and 2 assumptions, which are reduced to the DLIN assumption via the hierarchical reduction technique. See the security proof (outline) of Theorem 1 for more details of these forms and security game transformations.

In the security proof, we also apply the information theoretical technique using hidden bases in DPVS, which has been described above as the inter-subspace transformation.

1.4. *Related Works*

The definitional works for functional encryption were initiated by Boneh et al. [14] and O'Neill [41]. They presented two types of definitions, the simulation (SIM)-based one and the indistinguishability (IND)-based one. Boneh et al. [14], Agrawal et al. [1] and Caro et al. [18] showed that a FE scheme with unbounded number of keys and ciphertexts in the standard model cannot be achieved in the SIM-based definition. Therefore, a fully secure functional encryption (with unbounded number of keys and ciphertexts) in the standard model should be realized in the IND-based definition.

As described before, there are two properties of functional encryption, attribute-hiding (or private-index) and payload-hiding (or public-index) [14,30].

Although several FE schemes for general circuits or Turing machines are presented by using indistinguishable obfuscations (iO) or multi-linear maps [2,22,23,26], while these primitives are currently on fragile ground and extremely inefficient.

The largest class of relations supported by a (public-index) FE scheme without using iO and multi-linear maps is general circuits [27]; however, they are not fully secure but *selectively secure* and still impractical.

To the best of our knowledge, the largest class of relations supported by a *fully secure* practical (public-index) FE scheme in the IND-based definition (with unbounded number of keys and ciphertexts) under a standard assumption in the standard model is non-monotone span programs with inner-product relations, which is achieved by this paper. The ABE scheme in [32] supports only monotone span programs with the equality relation, and the assumptions are non-standard on composite-order pairing groups.

Spatial encryption [12,19] supports a fairly large class of relations but still a limited class of those by the proposed scheme. Although some extensions of spatial encryption have been proposed [20], the relations supported by the scheme are also covered by those of the proposed FE scheme.

To the best of our knowledge, the largest class of a fully secure and (weakly) attribute-hiding practical FE scheme in the IND-based definition under reasonable assumptions in the standard model is the conjunction of inner-product relations (e.g., hierarchical inner-product relations and basic spacial encryption), which is achieved in this paper. The (H)IPE scheme in [32] is (weakly) attribute-hiding under a non-standard assumption.

Although an attribute-hiding FE scheme, (H)IPE scheme, specialized from the proposed FE scheme in this paper, is weakly attribute-hiding, *fully*-attribute-hiding (H)IPE schemes (in the IND-based definition) were presented under the same assumption, DLIN assumption, by [38,39].

Our general access structures, i.e., span programs over inner-product predicates, have nice applications with sparse matrix DPVS techniques [40], for example, semi-adaptively secure KP-ABE scheme for span programs with constant-size ciphertexts (from DLIN) [46] and adaptively secure KP- and CP-ABE schemes from DLIN which allow attribute reuse in an available formula without the redundant multiple encoding technique given in "Appendix E" [47].

1.5. *Notations*

When A is a random variable or distribution, $y \xleftarrow{\mathsf{R}} A$ denotes that y is randomly selected from A according to its distribution. When A is a set, $y \xleftarrow{\mathsf{U}} A$ denotes that y is uniformly selected from A. $y := z$ denotes that y is set, defined or substituted by z. When a is a fixed value, $A(x) \to a$ (e.g., $A(x) \to 1$) denotes the event that machine (algorithm) A outputs a on input x. A function $f : \mathbb{N} \to \mathbb{R}$ is *negligible* in λ, if for every constant $c > 0$, there exists an integer n such that $f(\lambda) < \lambda^{-c}$ for all $\lambda > n$.

We denote the finite field of order q by $\mathbb{F}_q$, and $\mathbb{F}_q \backslash \{0\}$ by $\mathbb{F}_q^\times$. 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 abused as the zero vector in $\mathbb{F}_q^n$ for any n. X^{T} denotes the transpose of matrix X. I_ℓ and 0_ℓ denote the $\ell \times \ell$ identity matrix and the $\ell \times \ell$ zero matrix, respectively. A bold face letter denotes an element of vector space $\mathbb{V}$, e.g., $\boldsymbol{x} \in \mathbb{V}$. When $\boldsymbol{b}_i \in \mathbb{V}$ $(i = 1, \ldots, n)$, $\mathsf{span}\langle \boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \rangle \subseteq \mathbb{V}$ (resp. $\mathsf{span}\langle \vec{x}_1, \ldots, \vec{x}_n \rangle$) denotes the subspace generated by $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ (resp. $\vec{x}_1, \ldots, \vec{x}_n$). For vectors $\vec{x} := (x_1, \ldots, x_N)$, $\vec{y} := (y_1, \ldots, y_N) \in \mathbb{F}_q^N$ and bases $\mathbb{B} := (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_N)$, $\mathbb{B}^* := (\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_N^*)$, $(\vec{x})_{\mathbb{B}}$ $(= (x_1, \ldots, x_N)_{\mathbb{B}})$ denotes linear combination $\sum_{i=1}^N x_i \boldsymbol{b}_i$, and $(\vec{y})_{\mathbb{B}^*}$ $(= (y_1, \ldots, y_N)_{\mathbb{B}^*})$ denotes $\sum_{i=1}^N y_i \boldsymbol{b}_i^*$. For a format of attribute vectors $\vec{n} := (d; n_1, \ldots, n_d)$ that indicates dimensions of vector spaces, $\vec{e}_{t,j}$ denotes the canonical basis vector $(\overbrace{0 \cdots 0}^{j-1}, 1, \overbrace{0 \cdots 0}^{n_t - j}) \in \mathbb{F}_q^{n_t}$ for $t = 1, \ldots, d$ and $j = 1, \ldots, n_t$. $GL(n, \mathbb{F}_q)$ denotes the general linear group of degree n over $\mathbb{F}_q$.

2. Dual Pairing Vector Spaces (DPVS) and Main Lemmas

In this section, we present the notion of dual pairing vector spaces (DPVS) and a typical construction of DPVS from pairing groups. We also show main lemmas on DPVS, which are directly employed for the security proof of the proposed FE schemes.

2.1. *DPVS 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) [35,36] 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 A.2". 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 non-degenerate 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 non-degenerate 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 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 using $\mathcal{G}_{\mathsf{bpg}}$.

Remark 1. For matrix $W := (w_{i,j})_{i,j=1,\ldots,N} \in \mathbb{F}_q^{N \times N}$ and element $\boldsymbol{g} := (G_1, \ldots, G_N)$ in N-dimensional $\mathbb{V}$, $\boldsymbol{g}W$ denotes $(\sum_{i=1}^{N} G_i w_{i,1}, \ldots, \sum_{i=1}^{N} G_i w_{i,N}) = (\sum_{i=1}^{N} w_{i,1} G_i, \ldots, \sum_{i=1}^{N} w_{i,N} G_i)$ by a natural multiplication of a N-dim. row vector and a $N \times N$ matrix. Thus, it holds an associative law as $(\boldsymbol{g}W)W^{-1} = \boldsymbol{g}(WW^{-1}) = \boldsymbol{g}$ and a pairing invariance property $e(\boldsymbol{g}W, \boldsymbol{h}(W^{-1})^{\mathrm{T}}) = e(\boldsymbol{g}, \boldsymbol{h})$ for any $\boldsymbol{g}, \boldsymbol{h} \in \mathbb{V}$.

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

$$
\begin{aligned}
&\mathcal{G}_{\mathsf{ob}}(1^\lambda, \vec{n} := (d; n_1, \ldots, n_d)) : \mathsf{param}_{\mathbb{G}} := (q, \mathbb{G}, \mathbb{G}_T, G, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{bpg}}(1^\lambda), \quad \psi \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times,\\
&\quad N_0 := 5, \quad N_t := 3n_t + 1 \ \text{ for } t = 1, \ldots, d,\\
&\quad \text{for } t = 0, \ldots, d,\\
&\qquad \mathsf{param}_{\mathbb{V}_t} := (q, \mathbb{V}_t, \mathbb{G}_T, \mathbb{A}_t, e) := \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, N_t, \mathsf{param}_{\mathbb{G}}),\\
&\qquad X_t := \begin{pmatrix} \vec{\chi}_{t,1} \\ \vdots \\ \vec{\chi}_{t,N_t} \end{pmatrix} := (\chi_{t,i,j})_{i,j} \xleftarrow{\mathsf{U}} GL(N_t, \mathbb{F}_q), \quad \begin{pmatrix} \vec{\vartheta}_{t,1} \\ \vdots \\ \vec{\vartheta}_{t,N_t} \end{pmatrix} := (\vartheta_{t,i,j})_{i,j} := \psi \cdot (X_t^{\mathrm{T}})^{-1},\\
&\qquad \boldsymbol{b}_{t,i} := (\vec{\chi}_{t,i})_{\mathbb{A}_t} = \textstyle\sum_{j=1}^{N_t} \chi_{t,i,j} \boldsymbol{a}_{t,j} \ \text{ for } i = 1, \ldots, N_t, \quad \mathbb{B}_t := (\boldsymbol{b}_{t,1}, \ldots, \boldsymbol{b}_{t,N_t}),\\
&\qquad \boldsymbol{b}^*_{t,i} := (\vec{\vartheta}_{t,i})_{\mathbb{A}_t} = \sum_{j=1}^{N_t} \vartheta_{t,i,j} \boldsymbol{a}_{t,j} \ \text{ for } i = 1, \ldots, N_t, \quad \mathbb{B}^*_t := (\boldsymbol{b}^*_{t,1}, \ldots, \boldsymbol{b}^*_{t,N_t}),\\
&\quad g_T := e(G, G)^\psi, \quad \mathsf{param}_{\vec{n}} := (\{\mathsf{param}_{\mathbb{V}_t}\}_{t=0,\ldots,d}, \ g_T),\\
&\quad \text{return } \ (\mathsf{param}_{\vec{n}}, \{\mathbb{B}_t, \mathbb{B}^*_t\}_{t=0,\ldots,d}).
\end{aligned}
$$

We note that $g_T = e(\boldsymbol{b}_{t,i}, \boldsymbol{b}^*_{t,i})$ for $t = 0, \ldots, d; i = 1, \ldots, N_t$.

2.2. *Decisional Linear (DLIN) Assumption*

Definition 3. (*DLIN: decisional linear assumption* [9]) The DLIN problem is to guess $\beta \in \{0, 1\}$, given $(\mathsf{param}_{\mathbb{G}}, \ G, \xi G, \kappa G, \delta\xi G, \sigma\kappa G, Y_\beta) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{DLIN}}_\beta(1^\lambda)$, where

$$
\begin{aligned}
&\mathcal{G}^{\mathsf{DLIN}}_\beta(1^\lambda) : \mathsf{param}_{\mathbb{G}} := (q, \mathbb{G}, \mathbb{G}_T, G, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{bpg}}(1^\lambda),\\
&\quad \kappa, \delta, \xi, \sigma \xleftarrow{\mathsf{U}} \mathbb{F}_q, \quad Y_0 := (\delta + \sigma)G, \quad Y_1 \xleftarrow{\mathsf{U}} \mathbb{G},\\
&\quad \text{return } \ (\mathsf{param}_{\mathbb{G}}, \ G, \xi G, \kappa G, \delta\xi G, \sigma\kappa G, Y_\beta),
\end{aligned}
$$

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:

$$
\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}}(\lambda) := \left| \mathsf{Pr}\left[\mathcal{E}(1^\lambda, \varrho) \to 1 \;\middle|\; \varrho \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{DLIN}}_0(1^\lambda) \right] - \mathsf{Pr}\left[\mathcal{E}(1^\lambda, \varrho) \to 1 \;\middle|\; \varrho \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{DLIN}}_1(1^\lambda) \right] \right|.
$$

The DLIN assumption is: For any probabilistic polynomial-time adversary $\mathcal{E}$, the advantage $\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}}(\lambda)$ is negligible in λ.

2.3. *Main Lemmas (Lemmas 1, 2 and 3)*

We will show three lemmas directly employed in the proof of Theorems 1 and 2 . The proofs of the lemmas are given in "Appendix B".

Definition 4. *(Problem 1)* Problem 1 is to guess β, given $(\mathsf{param}_{\vec{n}}, \mathbb{B}_0, \widehat{\mathbb{B}}_0^*, \boldsymbol{e}_{\beta,0}, \{\mathbb{B}_t, \widehat{\mathbb{B}}_t^*, \boldsymbol{e}_{\beta,t,1}, \boldsymbol{e}_{t,i}\}_{t=1,\ldots,d;i=2,\ldots,n_t}) \xleftarrow{\mathsf{R}} \mathcal{G}_\beta^{\mathsf{P1}}(1^\lambda, \vec{n})$, where

$$
\begin{aligned}
&\mathcal{G}_\beta^{\mathsf{P1}}(1^\lambda, \vec{n}): \quad (\mathsf{param}_{\vec{n}}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0,\ldots,d}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}(1^\lambda, \vec{n}),\\
&\widehat{\mathbb{B}}_0^* := (\boldsymbol{b}_{0,1}^*, \boldsymbol{b}_{0,3}^*, .., \boldsymbol{b}_{0,5}^*),\ \widehat{\mathbb{B}}_t^* := (\boldsymbol{b}_{t,1}^*, .., \boldsymbol{b}_{t,n_t}^*, \boldsymbol{b}_{t,2n_t+1}^*, .., \boldsymbol{b}_{t,3n_t+1}^*) \ \text{ for } t = 1, .., d,\\
&\omega, z_0, \gamma_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q, \quad \boldsymbol{e}_{0,0} := (\omega, 0, 0, 0, \gamma_0)_{\mathbb{B}_0}, \quad \boldsymbol{e}_{1,0} := (\omega, z_0, 0, 0, \gamma_0)_{\mathbb{B}_0},\\
&\text{for } t = 1, \ldots, d;\\
&\quad \vec{e}_{t,1} := (1, 0^{n_t-1}) \in \mathbb{F}_q^{n_t},\ \vec{z}_t \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_t},\ \gamma_t \xleftarrow{\mathsf{U}} \mathbb{F}_q,\\
&\quad \boldsymbol{e}_{0,t,1} := (\ \overbrace{\omega\vec{e}_{t,1},}^{n_t}\ \overbrace{0^{n_t},}^{n_t}\ \overbrace{0^{n_t},}^{n_t}\ \overbrace{\gamma_t}^{1}\ )_{\mathbb{B}_t},\\
&\quad \boldsymbol{e}_{1,t,1} := (\ \omega\vec{e}_{t,1}, \quad \vec{z}_t, \quad 0^{n_t}, \quad \gamma_t\ )_{\mathbb{B}_t},\\
&\quad \boldsymbol{e}_{t,i} := \omega\boldsymbol{b}_{t,i} \ \text{ for } i = 2, \ldots, n_t,\\
&\text{return } (\mathsf{param}_{\vec{n}}, \mathbb{B}_0, \widehat{\mathbb{B}}_0^*, \boldsymbol{e}_{\beta,0}, \{\mathbb{B}_t, \widehat{\mathbb{B}}_t^*, \boldsymbol{e}_{\beta,t,1}, \boldsymbol{e}_{t,i}\}_{t=1,\ldots,d;i=2,\ldots,n_t}),
\end{aligned}
$$

for $\beta \xleftarrow{\mathsf{U}} \{0, 1\}$. For a probabilistic machine $\mathcal{B}$, we define the advantage of $\mathcal{B}$ as the quantity

$$
\mathsf{Adv}_{\mathcal{B}}^{\mathsf{P1}}(\lambda) := \left| \mathsf{Pr}\left[\mathcal{B}(1^\lambda, \varrho) \to 1 \,\middle|\, \varrho \xleftarrow{\mathsf{R}} \mathcal{G}_0^{\mathsf{P1}}(1^\lambda, \vec{n}) \right] - \mathsf{Pr}\left[\mathcal{B}(1^\lambda, \varrho) \to 1 \,\middle|\, \varrho \xleftarrow{\mathsf{R}} \mathcal{G}_1^{\mathsf{P1}}(1^\lambda, \vec{n}) \right] \right|.
$$

Lemma 1. *For any adversary $\mathcal{B}$, there exist probabilistic machines $\mathcal{E}$, whose running times are essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{P1}}(\lambda) \le \mathsf{Adv}_{\mathcal{E}}^{DLIN}(\lambda) + (d+6)/q$.*

Definition 5. *(Problem 2)* Problem 2 is to guess β, given $(\mathsf{param}_{\vec{n}}, \widehat{\mathbb{B}}_0, \mathbb{B}_0^*, \boldsymbol{h}_{\beta,0}^*, \boldsymbol{e}_0, \{\widehat{\mathbb{B}}_t, \mathbb{B}_t^*, \boldsymbol{h}_{\beta,t,i}^*, \boldsymbol{e}_{t,i}\}_{t=1,\ldots,d;i=1,\ldots,n_t}) \xleftarrow{\mathsf{R}} \mathcal{G}_\beta^{\mathsf{P2}}(1^\lambda, \vec{n})$, where

$$
\begin{aligned}
&\mathcal{G}_\beta^{\mathsf{P2}}(1^\lambda, \vec{n}): \quad (\mathsf{param}_{\vec{n}}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0,\ldots,d}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}(1^\lambda, \vec{n}),\\
&\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, .., \boldsymbol{b}_{0,5}),\ \widehat{\mathbb{B}}_t := (\boldsymbol{b}_{t,1}, .., \boldsymbol{b}_{t,n_t}, \boldsymbol{b}_{t,2n_t+1}, .., \boldsymbol{b}_{t,3n_t+1}) \ \text{ for } t = 1, .., d,\\
&\delta, \delta_0, \omega \xleftarrow{\mathsf{U}} \mathbb{F}_q,\ \tau, u_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times,\ z_0 := u_0^{-1},\\
&\begin{pmatrix} \vec{z}_{t,1} \\ \vdots \\ \vec{z}_{t,n_t} \end{pmatrix} := Z_t \xleftarrow{\mathsf{U}} GL(n_t, \mathbb{F}_q),\ \begin{pmatrix} \vec{u}_{t,1} \\ \vdots \\ \vec{u}_{t,n_t} \end{pmatrix} := (Z_t^{-1})^{\mathrm{T}} \text{ for } t = 1, .., d,\\
&\boldsymbol{h}_{0,0}^* := (\delta, 0, 0, \delta_0, 0)_{\mathbb{B}_0^*},\ \boldsymbol{h}_{1,0}^* := (\delta, u_0, 0, \delta_0, 0)_{\mathbb{B}_0^*},\ \boldsymbol{e}_0 := (\omega, \tau z_0, 0, 0, 0)_{\mathbb{B}_0},\\
&\text{for } t = 1, \ldots, d;\ i = 1, \ldots, n_t;\\
&\quad \vec{e}_{t,i} := (0^{i-1}, 1, 0^{n_t-i}) \in \mathbb{F}_q^{n_t},\ \vec{\delta}_{t,i} \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_t},
\end{aligned}
$$

$$
\begin{aligned}
&\boldsymbol{h}^*_{0,t,i} := (\ \overbrace{\delta\vec{e}_{t,i},}^{n_t}\ \overbrace{0^{n_t},}^{n_t}\ \overbrace{\vec{\delta}_{t,i},}^{n_t}\ \overbrace{0}^{1}\ )_{\mathbb{B}^*_t}\\
&\boldsymbol{h}^*_{1,t,i} := (\ \delta\vec{e}_{t,i},\ \vec{u}_{t,i},\ \vec{\delta}_{t,i},\ 0\ )_{\mathbb{B}^*_t}\\
&\boldsymbol{e}_{t,i} := (\ \omega\vec{e}_{t,i},\ \tau\vec{z}_{t,i},\ 0^{n_t},\ 0\ )_{\mathbb{B}_t},\\
&\text{return } (\mathsf{param}_{\vec{n}}, \widehat{\mathbb{B}}_0, \mathbb{B}^*_0, \boldsymbol{h}^*_{\beta,0}, \boldsymbol{e}_0, \{\widehat{\mathbb{B}}_t, \mathbb{B}^*_t, \boldsymbol{h}^*_{\beta,t,i}, \boldsymbol{e}_{t,i}\}_{t=1,..,d;i=1,..,n_t}),
\end{aligned}
$$

for $\beta \xleftarrow{\cup} \{0,1\}$. 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 4.

Lemma 2. *For any adversary $\mathcal{B}$, 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, $\mathsf{Adv}^{P2}_{\mathcal{B}}(\lambda) \leq \mathsf{Adv}^{DLIN}_{\mathcal{E}}(\lambda) + 5/q$.*

Lemma 3. *For $p \in \mathbb{F}_q$, let $C_p := \{(\vec{x},\vec{v}) | \vec{x}\cdot\vec{v} = p, \vec{x} \neq \vec{0}, \vec{v} \neq \vec{0}\} \subset \mathbb{F}_q^n \times \mathbb{F}_q^n$. For all $(\vec{x},\vec{v}) \in C_p$, for all $(\vec{r},\vec{w}) \in C_p$, $\Pr\left[\vec{x}U = \vec{r} \ \wedge\ \vec{v}Z = \vec{w}\right] = \Pr\left[\vec{x}Z = \vec{r} \ \wedge\ \vec{v}U = \vec{w}\right] = 1/\sharp C_p$, where $Z \xleftarrow{\cup} GL(n,\mathbb{F}_q)$, $U := (Z^{-1})^{\mathrm{T}}$.*

3. Functional Encryption with a Large Class of Relations

In this section, we provide the definition of functional encryption with a large class of relations, which are specified by non-monotone access structures combined with inner-product relations.

As described in Sect. 1.3.4, vectors, $\vec{x}$ and $\vec{v}$, with a ciphertext and secret key are expressed by the form of $(t,\vec{x})$ and $(t,\vec{v})$, which mean that $\vec{x}$ and $\vec{v}$ are in a category or subuniverse, $\mathcal{U}_t$, i.e., t is the identity of a category or subuniverse, $\mathcal{U}_t$.

Non-monotone access structures can be realized by span programs (Definition 6) and be combined with inner-product relations (Definition 7).

3.1. *Span Programs and Non-Monotone Access Structures*

Definition 6. (*Span programs* [4]) Let $\{p_1,\ldots,p_n\}$ be a set of variables. A span program over $\mathbb{F}_q$ is a labeled matrix $\hat{M} := (M,\rho)$ where M is a $(\ell \times r)$ matrix over $\mathbb{F}_q$ and ρ is a labeling of the rows of M by literals from $\{p_1,\ldots,p_n,\neg p_1,\ldots,\neg p_n\}$ (every row is labeled by one literal), i.e., $\rho : \{1,\ldots,\ell\} \to \{p_1,\ldots,p_n,\neg p_1,\ldots,\neg p_n\}$.

A span program accepts or rejects an input by the following criterion. For every input sequence $\delta \in \{0,1\}^n$ define the submatrix M_δ of M consisting of those rows whose labels are set to 1 by the input δ, i.e., either rows labeled by some p_i such that $\delta_i = 1$ or rows labeled by some $\neg p_i$ such that $\delta_i = 0$. (i.e., $\gamma : \{1,\ldots,\ell\} \to \{0,1\}$ is defined by $\gamma(j) = 1$ if $[\rho(j) = p_i] \wedge [\delta_i = 1]$ or $[\rho(j) = \neg p_i] \wedge [\delta_i = 0]$, and $\gamma(j) = 0$ otherwise. $M_\delta := (M_j)_{\gamma(j)=1}$, where M_j is the jth row of M.)

The span program $\hat{M}$ accepts δ if and only if $\vec{1} \in \mathsf{span}\langle M_\delta\rangle$, i.e., some linear combination of the rows of M_δ gives the all one vector $\vec{1}$. (The row vector has the value 1 in each coordinate.) A span program computes a Boolean function f if it accepts exactly those inputs δ where $f(\delta) = 1$.

A span program is called monotone if the labels of the rows are only the positive literals $\{p_1, \ldots, p_n\}$. Monotone span programs compute monotone functions. (So, a span program in general is "non"-monotone.)

We assume that no row M_i $(i = 1, \ldots, \ell)$ of the matrix M is $\vec{0}$. We now introduce a non-monotone access structure with evaluating map γ by using the inner-product of attribute vectors, that is employed in the proposed functional encryption schemes.

Definition 7. (*Inner-products of attribute vectors and access structures*) $\mathcal{U}_t$ $(t = 1, \ldots, d$ and $\mathcal{U}_t \subset \{0,1\}^*)$ is a subuniverse, a set of vectors, each of which is expressed by a pair of subuniverse id and n_t-dimensional vector, i.e., $(t, \vec{v})$, where $t \in \{1, \ldots, d\}$ and $\vec{v} \in \mathbb{F}_q^{n_t} \backslash \{\vec{0}\}$.

We now define such an attribute to be a variable p of a span program $\hat{M} := (M, \rho)$, i.e., $p := (t, \vec{v})$. An access structure $\mathbb{S}$ is span program $\hat{M} := (M, \rho)$ along with variables $p := (t, \vec{v})$, $p' := (t', \vec{v}')$, ..., i.e., $\mathbb{S} := (M, \rho)$ such that $\rho : \{1, \ldots, \ell\} \to \{(t, \vec{v}), (t', \vec{v}'), \ldots, \neg(t, \vec{v}), \neg(t', \vec{v}'), \ldots\}$.

Let Γ be a set of attributes, i.e., $\Gamma := \{(t, \vec{x}_t) \mid \vec{x}_t \in \mathbb{F}_q^{n_t} \backslash \{\vec{0}\}, 1 \le t \le d\}$, where $1 \le t \le d$ means that t is an element of some subset of $\{1, \ldots, d\}$.

When Γ is given to access structure $\mathbb{S}$, map $\gamma : \{1, \ldots, \ell\} \to \{0, 1\}$ for span program $\hat{M} := (M, \rho)$ is defined as follows: For $i = 1, \ldots, \ell$, set $\gamma(i) = 1$ if $[\rho(i) = (t, \vec{v}_i)] \wedge [(t, \vec{x}_t) \in \Gamma] \wedge [\vec{v}_i \cdot \vec{x}_t = 0]$ or $[\rho(i) = \neg(t, \vec{v}_i)] \wedge [(t, \vec{x}_t) \in \Gamma] \wedge [\vec{v}_i \cdot \vec{x}_t \neq 0]$. Set $\gamma(i) = 0$ otherwise.

Access structure $\mathbb{S} := (M, \rho)$ accepts Γ iff $\vec{1} \in \mathsf{span}\langle (M_i)_{\gamma(i)=1} \rangle$.

Remark 2. The restriction that $\vec{v} \neq \vec{0}$ and $\vec{x}_t \neq \vec{0}$ above is required by the security proof or more specifically by Lemma 3. This restriction is reasonable in many applications. For example, in the equality relations for ABE, $\vec{v} := (v, -1)$ and $\vec{x} := (1, x)$, where $v = x$ iff $\vec{v} \cdot \vec{x} = 0$.

We now construct a secret-sharing scheme for a non-monotone access structure or span program.

Definition 8. A secret-sharing scheme for span program $\hat{M} := (M, \rho)$ is:

1. Let M be $\ell \times r$ matrix. Let column vector $\vec{f}^{\mathrm{T}} := (f_1, \ldots, f_r)^{\mathrm{T}} \xleftarrow{\cup} \mathbb{F}_q^r$. Then, $s_0 := \vec{1} \cdot \vec{f}^{\mathrm{T}} = \sum_{k=1}^{r} f_k$ is the secret to be shared, and $\vec{s}^{\mathrm{T}} := (s_1, \ldots, s_\ell)^{\mathrm{T}} := M \cdot \vec{f}^{\mathrm{T}}$ is the vector of ℓ shares of the secret s_0 and the share s_i belongs to $\rho(i)$.
2. If span program $\hat{M} := (M, \rho)$ accept δ, or access structure $\mathbb{S} := (M, \rho)$ accepts Γ, i.e., $\vec{1} \in \mathsf{span}\langle (M_i)_{\gamma(i)=1} \rangle$ with $\gamma : \{1, \ldots, \ell\} \to \{0, 1\}$, then there exist constants $\{\alpha_i \in \mathbb{F}_q \mid i \in I\}$ such that $I \subseteq \{i \in \{1, \ldots, \ell\} \mid \gamma(i) = 1\}$ and $\sum_{i \in I} \alpha_i s_i = s_0$. Furthermore, these constants $\{\alpha_i\}$ can be computed in time polynomial in the size of matrix M.

3.2. *Key-Policy Functional Encryption with a Large Class of Relations*

Definition 9. (*Key-policy functional encryption: KP-FE*) A key-policy functional encryption scheme consists of four algorithms.

Setup This is a randomized algorithm that takes as input security parameter and format $\vec{n} := (d; n_1, \ldots, n_d)$ of attributes. It outputs public parameters pk and master secret key sk.

KeyGen This is a randomized algorithm that takes as input access structure $\mathbb{S} := (M, \rho)$, pk and sk. It outputs a decryption key $\mathsf{sk}_{\mathbb{S}}$.

Enc This is a randomized algorithm that takes as input message m, a set of attributes, $\Gamma := \{(t, \vec{x}_t) | \vec{x}_t \in \mathbb{F}_q^{n_t} \backslash \{\vec{0}\}, 1 \leq t \leq d\}$, and public parameters pk. It outputs a ciphertext ct_Γ.

Dec This takes as input ciphertext ct_Γ that was encrypted under a set of attributes Γ, decryption key $\mathsf{sk}_{\mathbb{S}}$ for access structure $\mathbb{S}$, and public parameters pk. It outputs either plaintext m or the distinguished symbol $\perp$.

A KP-FE scheme should have the following correctness property: for all $(\mathsf{pk}, \mathsf{sk}) \xleftarrow{R} \mathsf{Setup}(1^\lambda, \vec{n})$, all access structures $\mathbb{S}$, all decryption keys $\mathsf{sk}_{\mathbb{S}} \xleftarrow{R} \mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \mathbb{S})$, all messages m, all attribute sets Γ, all ciphertexts $\mathsf{ct}_\Gamma \xleftarrow{R} \mathsf{Enc}(\mathsf{pk}, m, \Gamma)$, it holds that $m = \mathsf{Dec}(\mathsf{pk}, \mathsf{sk}_{\mathbb{S}}, \mathsf{ct}_\Gamma)$ with overwhelming probability, if $\mathbb{S}$ accepts Γ.

Definition 10. The model for proving the adaptively payload-hiding security of KP-FE under chosen-plaintext attack is:

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

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

Challenge The adversary submits two messages $m^{(0)}, m^{(1)}$ and a set of attributes, Γ, provided that no $\mathbb{S}$ queried to the challenger in Phase 1 accepts Γ. The challenger flips a coin $b \xleftarrow{U} \{0, 1\}$, and computes $\mathsf{ct}_\Gamma^{(b)} \xleftarrow{R} \mathsf{Enc}(\mathsf{pk}, m^{(b)}, \Gamma)$. It gives $\mathsf{ct}_\Gamma^{(b)}$ to the adversary.

Phase 2 The adversary is allowed to adaptively issue a polynomial number of queries, $\mathbb{S}$, to the challenger or oracle $\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \cdot)$ for private keys, $\mathsf{sk}_{\mathbb{S}}$ associated with $\mathbb{S}$, provided that $\mathbb{S}$ does not accept Γ.

Guess The adversary outputs a guess b' of b.

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

We note that the model can easily be extended to handle chosen-ciphertext attacks (CCA) by allowing for decryption queries in Phases 1 and 2. The advantage of adversary

$\mathcal{A}$ in the CCA game is defined as $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{KP-FE,CCA-PH}}(\lambda) := \Pr[b' = b] - 1/2$ for any security parameter λ.

3.3. *Ciphertext-Policy Functional Encryption with a Large Class of Relations*

Definition 11. (*Ciphertext-policy functional encryption: CP-FE*) A ciphertext-policy functional encryption scheme consists of four algorithms.

Setup This is a randomized algorithm that takes as input security parameter and format $\vec{n} := (d; n_1, \ldots, n_d)$ of attributes. It outputs the public parameters pk and a master key sk.

KeyGen This is a randomized algorithm that takes as input a set of attributes, $\Gamma := \{(t, \vec{x}_t) | \vec{x}_t \in \mathbb{F}_q^{n_t}, 1 \le t \le d\}$, pk and sk. It outputs a decryption key.

Enc This is a randomized algorithm that takes as input message m, access structure $\mathbb{S} := (M, \rho)$, and the public parameters pk. It outputs the ciphertext.

Dec This takes as input the ciphertext that was encrypted under access structure $\mathbb{S}$, the decryption key for a set of attributes Γ, and the public parameters pk. It outputs either plaintext m or the distinguished symbol $\perp$.

A CP-FE scheme should have the following correctness property: for all $(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\mathsf{R}} \mathsf{Setup}(1^\lambda, \vec{n})$, all attribute sets Γ, all decryption keys $\mathsf{sk}_\Gamma \xleftarrow{\mathsf{R}} \mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \Gamma)$, all messages m, all access structures $\mathbb{S}$, all ciphertexts $\mathsf{ct}_\mathbb{S} \xleftarrow{\mathsf{R}} \mathsf{Enc}(\mathsf{pk}, m, \mathbb{S})$, it holds that $m = \mathsf{Dec}(\mathsf{pk}, \mathsf{sk}_\Gamma, \mathsf{ct}_\mathbb{S})$ with overwhelming probability, if $\mathbb{S}$ accepts Γ.

Definition 12. The model for proving the adaptively payload-hiding security of CP-FE under chosen-plaintext attack is:

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

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

Challenge The adversary submits two messages $m^{(0)}$, $m^{(1)}$ and an access structure, $\mathbb{S} := (M, \rho)$, provided that the $\mathbb{S}$ does not accept any Γ sent to the challenger in Phase 1. The challenger flips a random coin $b \xleftarrow{\mathsf{U}} \{0, 1\}$, and computes $\mathsf{ct}_\mathbb{S}^{(b)} \xleftarrow{\mathsf{R}} \mathsf{Enc}(\mathsf{pk}, m^{(b)}, \mathbb{S})$. It gives $\mathsf{ct}_\mathbb{S}^{(b)}$ to the adversary.

Phase 2 The adversary is allowed to issue a polynomial number of queries, Γ, to the challenger or oracle $\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \cdot)$ for private keys, sk_Γ associated with Γ, provided that $\mathbb{S}$ does not accept Γ.

Guess The adversary outputs a guess b' of b.

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

We note that the model can easily be extended to handle chosen-ciphertext attacks (CCA) by allowing for decryption queries in Phase 1 and 2. The advantage of an adversary

$\mathcal{A}$ in the CCA game is defined as $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{CP-FE,CCA-PH}}(\lambda) := \Pr[b' = b] - 1/2$ for any security parameter λ.

3.4. *Unified-Policy Functional Encryption with a Large Class of Relations*

Definition 13. (*Unified-Policy Functional Encryption: UP-FE*) A unified-policy functional encryption scheme consists of four algorithms.

Setup This is a randomized algorithm that takes as input security parameter and format $\vec{n} := ((d^{\mathsf{KP}}; n_1^{\mathsf{KP}}, \ldots, n_{d^{\mathsf{KP}}}^{\mathsf{KP}}), (d^{\mathsf{CP}}; n_1^{\mathsf{CP}}, \ldots, n_{d^{\mathsf{CP}}}^{\mathsf{CP}}))$ of attributes. It outputs public parameters pk and master secret key sk.

KeyGen This is a randomized algorithm that takes as input access structure $\mathbb{S}^{\mathsf{KP}} := (M^{\mathsf{KP}}, \rho^{\mathsf{KP}})$, a set of attributes, $\Gamma^{\mathsf{CP}} := \{(t, \vec{x}_t^{\mathsf{CP}}) | \vec{x}_t^{\mathsf{CP}} \in \mathbb{F}_q^{n_t^{\mathsf{CP}}} \setminus \{\vec{0}\}, 1 \leq t \leq d^{\mathsf{CP}}\}$, pk and sk. It outputs a decryption key $\mathsf{sk}_{(\mathbb{S}^{\mathsf{KP}}, \Gamma^{\mathsf{CP}})}$.

Enc This is a randomized algorithm that takes as input message m, a set of attributes, $\Gamma^{\mathsf{KP}} := \{(t, \vec{x}_t^{\mathsf{KP}}) | \vec{x}_t^{\mathsf{KP}} \in \mathbb{F}_q^{n_t^{\mathsf{KP}}} \setminus \{\vec{0}\}, 1 \leq t \leq d^{\mathsf{KP}}\}$, access structure $\mathbb{S}^{\mathsf{CP}} := (M^{\mathsf{CP}}, \rho^{\mathsf{CP}})$, and public parameters pk. It outputs a ciphertext $\mathsf{ct}_{(\Gamma^{\mathsf{KP}}, \mathbb{S}^{\mathsf{CP}})}$.

Dec This takes as input a ciphertext $\mathsf{ct}_{(\Gamma^{\mathsf{KP}}, \mathbb{S}^{\mathsf{CP}})}$ that was encrypted under a set of attributes and access structure, $(\Gamma^{\mathsf{KP}}, \mathbb{S}^{\mathsf{CP}})$, decryption key $\mathsf{sk}_{(\mathbb{S}^{\mathsf{KP}}, \Gamma^{\mathsf{CP}})}$ for access structure and a set of attributes, $(\mathbb{S}^{\mathsf{KP}}, \Gamma^{\mathsf{CP}})$, and public parameters pk. It outputs either plaintext m or the distinguished symbol $\perp$.

A UP-FE scheme should have the following correctness property: for all $(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\mathsf{R}} \mathsf{Setup}(1^\lambda, \vec{n})$, all access structures $\mathbb{S}^{\mathsf{KP}}$, all attribute sets Γ^{CP}, all decryption keys $\mathsf{sk}_{(\mathbb{S}^{\mathsf{KP}}, \Gamma^{\mathsf{CP}})} \xleftarrow{\mathsf{R}} \mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \mathbb{S}^{\mathsf{KP}}, \Gamma^{\mathsf{CP}})$, all messages m, all attribute sets Γ^{KP}, all access structures $\mathbb{S}^{\mathsf{CP}}$, all ciphertexts $\mathsf{ct}_{(\Gamma^{\mathsf{KP}}, \mathbb{S}^{\mathsf{CP}})} \xleftarrow{\mathsf{R}} \mathsf{Enc}(\mathsf{pk}, m, \Gamma^{\mathsf{KP}}, \mathbb{S}^{\mathsf{CP}})$, it holds that $m = \mathsf{Dec}(\mathsf{pk}, \mathsf{sk}_{(\mathbb{S}^{\mathsf{KP}}, \Gamma^{\mathsf{CP}})}, \mathsf{ct}_{(\Gamma^{\mathsf{KP}}, \mathbb{S}^{\mathsf{CP}})})$ with overwhelming probability, if $\mathbb{S}^{\mathsf{KP}}$ accepts Γ^{KP} and $\mathbb{S}^{\mathsf{CP}}$ accepts Γ^{CP}.

The adaptively payload-hiding security of UP-FE under chosen-plaintext attack (and chosen-ciphertext attack) are defined similarly as those of KP-FE and CP-FE. (See Definition 10, 12.)

4. KP-FE Scheme

This section presents a KP-FE scheme with the large class of relations, which is defined in Sect. 3.2.

4.1. *Key Idea of the Construction*

Our construction is based on the dual pairing vector spaces (DPVS) (Sect. 1.3.3). A pair of dual (or orthonormal) bases, $\mathbb{B}$ and $\mathbb{B}^*$, are randomly generated using random linear transformation, and a part of $\mathbb{B}$ (say $\hat{\mathbb{B}}$) is used as a public key and the corresponding part of $\mathbb{B}^*$ (say $\hat{\mathbb{B}}*$) is used as a secret key or trapdoor.

As mentioned in Sect. 1.3.4, in our KP-FE scheme, a ciphertext $\boldsymbol{c}$ with a n-dimensional vector $(t, \vec{x})$ is realized as

$$\boldsymbol{c} := (\omega\vec{x}, 0^n, 0^n, \varphi)_{\mathbb{B}_t},$$

where $\omega, \varphi \xleftarrow{\cup} \mathbb{F}_q$ and $\vec{x}$ is normalized as $(1, *, \ldots, *)$. A secret key $\boldsymbol{k}_i^*$ for the ith entry of a span program associated with a vector $(t, \vec{v}_i)$ is realized as

$$\begin{aligned}
\boldsymbol{k}_i^* &:= (s_i\vec{e}_1 + \theta_i\vec{v}_i, 0^n, \vec{\eta}_i, 0)_{\mathbb{B}_t^*} \quad \text{(if the } i\text{th entry is labeled 'positive')},\\
\boldsymbol{k}_i^* &:= (s_i\vec{v}_i, 0^{n_t}, \vec{\eta}_i, 0)_{\mathbb{B}_t^*} \quad \text{(if the } i\text{th entry is labeled 'negative')},
\end{aligned}$$

where s_i is the i-entry's share of the span program, $\vec{e}_1 := (1, 0, \ldots 0) \in \mathbb{F}_q^n$, $\theta_i \xleftarrow{\cup} \mathbb{F}_q$, $\vec{\eta}_i \xleftarrow{\cup} \mathbb{F}_q^n$.

The pairing operation of $\boldsymbol{c} \in \mathsf{span}\langle\mathbb{B}_t\rangle$ and $\boldsymbol{k}_i \in \mathsf{span}\langle\mathbb{B}_t^*\rangle$ is possible and is

$$\begin{aligned}
e(\boldsymbol{c}, \boldsymbol{k}_i^*) &= g_T^{\omega s_i + \omega\theta_i\vec{x}\cdot\vec{v}_i} \quad \text{(if the } i\text{th entry is labeled 'positive')},\\
e(\boldsymbol{c}, \boldsymbol{k}_i^*) &= g_T^{\omega s_i\vec{x}\cdot\vec{v}_i} \quad \text{(if the } i\text{th entry is labeled 'negative')},
\end{aligned}$$

Therefore,

$$\begin{aligned}
e(\boldsymbol{c}, \boldsymbol{k}_i^*) &= g_T^{\omega s_i} \quad \text{(if the } i\text{th entry is labeled 'positive' and } \vec{x}\cdot\vec{v}_i = 0),\\
e(\boldsymbol{c}, \boldsymbol{k}_i^*)^{1/\vec{x}\cdot\vec{v}_i} &= g_T^{\omega s_i} \quad \text{(if the } i\text{th entry is labeled 'negative' and } \vec{x}\cdot\vec{v}_i \neq 0),
\end{aligned}$$

When a subset of entries, where $g_T^{\omega s_i}$ is revealed, span the program, or the relation for the parameters of ciphertext and secret key holds in our scheme, a ciphertext can be decrypted.

A nice property of DPVS is that we can set a hidden linear subspace by concealing the basis of a subspace from the public key. Here, $\mathsf{span}\langle\mathbb{B}\rangle$ and $\mathsf{span}\langle\mathbb{B}^*\rangle$, are $(3n+1)$-dimensional (where the dimension of vectors is n), and, as for public parameter $\hat{\mathbb{B}}$, $\mathsf{span}\langle\hat{\mathbb{B}}\rangle$ is $(n+1)$-dimensional, i.e., the basis for the remaining $2n$-dimensional space is information theoretically concealed (ambiguous). The n-dimensional space in the space is employed for the randomness, $\vec{\eta}_i$, in a secret key, and the remaining n-dimensional hidden subspace is employed to realize the semi-functional forms of ciphertext and secret keys. Problems 1 and 2 assumptions (Definitions 4, 5) bridge the normal and semi-functional forms of ciphertext and secret keys.

4.2. *Construction*

We define function $\widetilde{\rho} : \{1, \ldots, \ell\} \to \{1, \ldots, d\}$ by $\widetilde{\rho}(i) := t$ if $\rho(i) = (t, \vec{v})$ or $\rho(i) = \neg(t, \vec{v})$, where ρ is given in access structure $\mathbb{S} := (M, \rho)$. In the proposed scheme, we assume that $\widetilde{\rho}$ is injective for $\mathbb{S} := (M, \rho)$ with decryption key $\mathsf{sk}_{\mathbb{S}}$. We will show how to relax the restriction in "Appendix E".

In the description of the scheme, we assume that input vector, $\vec{x}_t := (x_{t,1}, \ldots, x_{t,n_t})$, is normalized such that $x_{t,1} := 1$. (If $\vec{x}_t$ is not normalized, change it to a normalized one by $(1/x_{t,1}) \cdot \vec{x}_t$, assuming that $x_{t,1}$ is nonzero).

Random dual basis generator $\mathcal{G}_{\mathsf{ob}}(1^\lambda, \vec{n})$ is defined at the end of Sect. 2.1. We refer to Sect. 1.5 for notations on DPVS.

$\mathsf{Setup}(1^\lambda,\ \vec{n} := (d; n_1, \ldots, n_d)) :\quad (\mathsf{param}_{\vec{n}}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0,\ldots,d}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}(1^\lambda, \vec{n}),$

$\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5}),\ \ \widehat{\mathbb{B}}_t := (\boldsymbol{b}_{t,1}, .., \boldsymbol{b}_{t,n_t}, \boldsymbol{b}_{t,3n_t+1})\ \ \text{for } t = 1, .., d,$

$\widehat{\mathbb{B}}_0^* := (\boldsymbol{b}_{0,1}^*, \boldsymbol{b}_{0,3}^*, \boldsymbol{b}_{0,4}^*),\ \ \widehat{\mathbb{B}}_t^* := (\boldsymbol{b}_{t,1}^*, .., \boldsymbol{b}_{t,n_t}^*, \boldsymbol{b}_{t,2n_t+1}^*, .., \boldsymbol{b}_{t,3n_t}^*)\ \ \text{for } t = 1, .., d,$

$\mathsf{pk} := (1^\lambda, \mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}_t\}_{t=0,\ldots,d}),\ \ \mathsf{sk} := \{\widehat{\mathbb{B}}_t^*\}_{t=0,\ldots,d},$

return $\mathsf{pk}, \mathsf{sk}.$

$\mathsf{KeyGen}(\mathsf{pk},\ \mathsf{sk},\ \mathbb{S} := (M, \rho)) :$

$\vec{f} \xleftarrow{\mathsf{U}} \mathbb{F}_q^r,\ \vec{s}^{\mathrm{T}} := (s_1, \ldots, s_\ell)^{\mathrm{T}} := M \cdot \vec{f}^{\mathrm{T}},\ s_0 := \vec{1} \cdot \vec{f}^{\mathrm{T}},\ \eta_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q,$

$\boldsymbol{k}_0^* := (-s_0,\ 0,\ 1,\ \eta_0,\ 0)_{\mathbb{B}_0^*},$

for $i = 1, \ldots, \ell,$

if $\rho(i) = (t, \vec{v}_i := (v_{i,1}, \ldots, v_{i,n_t}) \in \mathbb{F}_q^{n_t} \backslash \{\vec{0}\}),\qquad \theta_i \xleftarrow{\mathsf{U}} \mathbb{F}_q,\ \vec{\eta}_i \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_t},$

$$\boldsymbol{k}_i^* := (\ \overbrace{s_i \vec{e}_{t,1} + \theta_i \vec{v}_i}^{n_t},\ \overbrace{0^{n_t}}^{n_t},\ \overbrace{\vec{\eta}_i}^{n_t},\ \overbrace{0}^{1}\ )_{\mathbb{B}_t^*},$$

if $\rho(i) = \neg(t, \vec{v}_i),\qquad \vec{\eta}_i \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_t},$

$$\boldsymbol{k}_i^* := (\ \overbrace{s_i \vec{v}_i}^{n_t},\ \overbrace{0^{n_t}}^{n_t},\ \overbrace{\vec{\eta}_i}^{n_t},\ \overbrace{0}^{1}\ )_{\mathbb{B}_t^*},$$

return $\mathsf{sk}_{\mathbb{S}} := (\mathbb{S}, \boldsymbol{k}_0^*, \boldsymbol{k}_1^*, \ldots, \boldsymbol{k}_\ell^*).$

$\mathsf{Enc}(\mathsf{pk},\ m,\ \Gamma := \{(t, \vec{x}_t := (x_{t,1}, .., x_{t,n_t}) \in \mathbb{F}_q^{n_t} \backslash \{\vec{0}\}) \mid 1 \le t \le d, x_{t,1} := 1\}) :$

$\omega, \varphi_0, \varphi_t, \zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q \text{ for } (t, \vec{x}_t) \in \Gamma,$

$\boldsymbol{c}_0 := (\omega,\ 0,\ \zeta,\ 0,\ \varphi_0)_{\mathbb{B}_0},$

$$\boldsymbol{c}_t := (\ \overbrace{\omega \vec{x}_t}^{n_t},\ \overbrace{0^{n_t}}^{n_t},\ \overbrace{0^{n_t}}^{n_t},\ \overbrace{\varphi_t}^{1}\ )_{\mathbb{B}_t} \text{ for } (t, \vec{x}_t) \in \Gamma,$$

$c_{d+1} := g_T^\zeta m,\ \ \mathsf{ct}_\Gamma := (\Gamma, \boldsymbol{c}_0, \{\boldsymbol{c}_t\}_{(t,\vec{x}_t)\in\Gamma}, c_{d+1}),$

return $\mathsf{ct}_\Gamma.$

$\mathsf{Dec}(\mathsf{pk},\ \mathsf{sk}_{\mathbb{S}} := (\mathbb{S}, \boldsymbol{k}_0^*, \boldsymbol{k}_1^*, \ldots, \boldsymbol{k}_\ell^*),\ \mathsf{ct}_\Gamma := (\Gamma, \boldsymbol{c}_0, \{\boldsymbol{c}_t\}_{(t,\vec{x}_t)\in\Gamma}, c_{d+1})) :$

If $\mathbb{S} := (M, \rho)$ accepts $\Gamma := \{(t, \vec{x}_t)\},$ then compute I and $\{\alpha_i\}_{i\in I}$ such that

$$\vec{1} = \sum_{i\in I} \alpha_i M_i,\ \text{where } M_i \text{ is the } i\text{th row of } M,\ \text{and}$$

$$I \subseteq \{i \in \{1, \ldots, \ell\}\ \mid\ [\rho(i) = (t, \vec{v}_i)\ \wedge\ (t, \vec{x}_t) \in \Gamma\ \wedge\ \vec{v}_i \cdot \vec{x}_t = 0]$$
$$\vee\ [\rho(i) = \neg(t, \vec{v}_i)\ \wedge\ (t, \vec{x}_t) \in \Gamma\ \wedge\ \vec{v}_i \cdot \vec{x}_t \neq 0]\ \},$$

$$K := e(\boldsymbol{c}_0, \boldsymbol{k}_0^*) \prod_{i\in I\ \wedge\ \rho(i)=(t,\vec{v}_i)} e(\boldsymbol{c}_t, \boldsymbol{k}_i^*)^{\alpha_i} \prod_{i\in I\ \wedge\ \rho(i)=\neg(t,\vec{v}_i)} e(\boldsymbol{c}_t, \boldsymbol{k}_i^*)^{\alpha_i/(\vec{v}_i\cdot\vec{x}_t)},$$

return $m' := c_{d+1}/K.$

[Correctness] If $\mathbb{S} := (M, \rho)$ accepts $\Gamma := \{(t, \vec{x}_t)\}$,

$$\begin{aligned} & e(\boldsymbol{c}_0, \boldsymbol{k}_0^*) \prod_{i\in I \ \wedge \ \rho(i)=(t,\vec{v}_i)} e(\boldsymbol{c}_t, \boldsymbol{k}_i^*)^{\alpha_i} \cdot \prod_{i\in I \ \wedge \ \rho(i)=\neg(t,\vec{v}_i)} e(\boldsymbol{c}_t, \boldsymbol{k}_i^*)^{\alpha_i/(\vec{v}_i\cdot\vec{x}_t)} \\ &= g_T^{-\omega s_0+\zeta} \prod_{i\in I \ \wedge \ \rho(i)=(t,\vec{v}_i)} g_T^{\omega\alpha_i s_i} \prod_{i\in I \ \wedge \ \rho(i)=\neg(t,\vec{v}_i)} g_T^{\omega\alpha_i s_i(\vec{v}_i\cdot\vec{x}_t)/(\vec{v}_i\cdot\vec{x}_t)} \\ &= g_T^{\omega(-s_0+\sum_{i\in I}\alpha_i s_i)+\zeta} = g_T^{\zeta}. \end{aligned}$$

4.3. *Security*

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

For any adversary $\mathcal{A}$, there exist probabilistic machines $\mathcal{E}_1$, $\mathcal{E}_2^+$, and $\mathcal{E}_2$, whose running times are essentially the same as that of $\mathcal{A}$, such that for any security parameter λ,

$$\mathsf{Adv}_{\mathcal{A}}^{\mathsf{KP-FE,PH}}(\lambda) \le \mathsf{Adv}_{\mathcal{E}_1}^{\mathsf{DLIN}}(\lambda) + \sum_{h=0}^{\nu-1} \left(\mathsf{Adv}_{\mathcal{E}_{2,h}^+}^{\mathsf{DLIN}}(\lambda) + \mathsf{Adv}_{\mathcal{E}_{2,h+1}}^{\mathsf{DLIN}}(\lambda) \right) + \epsilon,$$

where $\mathcal{E}_{2,h}^+(\cdot) := \mathcal{E}_2^+(h, \cdot)$, $\mathcal{E}_{2,h+1}(\cdot) := \mathcal{E}_2(h, \cdot)$ $(h = 0, \ldots, \nu-1)$, ν is the maximum number of $\mathcal{A}$'s key queries and $\epsilon := (2d\nu + 16\nu + d + 7)/q$.

Proof Outline of Theorem 1: At the top level of strategy of the security proof, we follow the dual system encryption methodology proposed by Waters [49]. In the methodology, ciphertexts and secret keys have two forms, *normal* and *semi-functional*. In the proof herein, we also introduce another form called *pre-semi-functional*. The real system uses only normal ciphertexts and normal secret keys, and semi-functional/pre-semi-functional ciphertexts and 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 2ν game changes from Game 1 (Game 2-0), Game 2-0^+, Game 2-1 through Game 2-$(\nu-1)^+$ and Game 2-ν. In Game 2-h, the first h keys are semi-functional while the remaining keys are normal, and the challenge ciphertext is semi-functional. In Game 2-h^+, the first h keys are semi-functional and the $(h+1)$th key is *pre-semi-functional* while the remaining keys are normal, and the challenge ciphertext is *pre-semi-functional*. The final game with advantage 0 is changed from Game 2-ν. As usual, we prove that the advantage gaps between neighboring games are negligible.

For $\mathsf{sk}_{\mathbb{S}} := (\mathbb{S}, \boldsymbol{k}_0^*, \boldsymbol{k}_1^*, \ldots, \boldsymbol{k}_\ell^*)$ and $\mathsf{ct}_\Gamma := (\Gamma, \boldsymbol{c}_0, \{\boldsymbol{c}_t\}_{(t,\vec{x}_t)\in\Gamma}, c_{d+1})$, we focus on $\vec{\boldsymbol{k}}_{\mathbb{S}}^* := (\boldsymbol{k}_0^*, \boldsymbol{k}_1^*, \ldots, \boldsymbol{k}_\ell^*)$ and $\vec{\boldsymbol{c}}_\Gamma := (\boldsymbol{c}_0, \{\boldsymbol{c}_t\}_{(t,\vec{x}_t)\in\Gamma})$, and ignore the other part of $\mathsf{sk}_{\mathbb{S}}$ and ct_Γ (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 $A \le B + \epsilon(\lambda)$ where $\epsilon(\lambda)$ is negligible in security parameter λ.

A *normal* secret key, $\vec{\boldsymbol{k}}_{\mathbb{S}}^{*\,\mathsf{norm}}$ (with access structure $\mathbb{S}$), is the correct form of the secret key of the proposed FE scheme, and is expressed by Eq. (1). Similarly, a *normal* ciphertext (with attribute set Γ), $\vec{\boldsymbol{c}}_\Gamma^{\mathsf{norm}}$, is expressed by Eq. (2). A *semi-functional* secret

key, $\vec{k}_{\mathbb{S}}^{*\,\mathsf{semi}}$, is expressed by Eq. (8), and a *semi-functional* ciphertext, $\vec{c}_{\Gamma}^{\mathsf{semi}}$, is expressed by Eqs. (3)–(5). A *pre-semi-functional* secret key, $\vec{k}_{\mathbb{S}}^{*\,\mathsf{pre-semi}}$, and *pre-semi-functional* ciphertext, $\vec{c}_{\Gamma}^{\mathsf{pre\text{-}semi}}$, are expressed by Eq. (6) and Eqs. (3), (7) and (5), respectively.

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 $\beta \xleftarrow{\mathsf{U}} \{0, 1\}$ 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 those of Game 1 when $\beta = 1$. That is, the advantage of Problem 1 is equivalent to the advantage gap between Games 0 and 1 (Lemma 4). The advantage of Problem 1 is proven to be equivalent to that of the DLIN assumption (Lemma 1).

The advantage gap between Games 2-h and 2-h^+ is similarly shown to be bounded by the advantage of Problem 2 (i.e., advantage of the DLIN assumption) (Lemmas 5 and 2). Here, we introduce *special forms of pre-semi-functional* keys and ciphertexts, $\vec{k}_{\mathbb{S}}^{*\,\mathsf{spec.pre-semi}}$ and $\vec{c}_{\Gamma}^{\,\mathsf{spec.pre-semi}}$, respectively, such that they are equivalent to pre-semi-functional keys and ciphertexts, $\vec{k}_{\mathbb{S}}^{*\,\mathsf{pre-semi}}$ and $\vec{c}_{\Gamma}^{\,\mathsf{pre\text{-}semi}}$, respectively, except that $w_0 r_0 = a_0 := \sum_{k=1}^{r} g_k$ and $r_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q$ (note that $r_0, w_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q$ for $\vec{k}_{\mathbb{S}}^{*\,\mathsf{pre-semi}}$ and $\vec{c}_{\Gamma}^{\,\mathsf{pre-semi}}$). These forms of keys and ciphertexts, $\vec{k}_{\mathbb{S}}^{*\,\mathsf{spec.pre-semi}}$ and $\vec{c}_{\Gamma}^{\,\mathsf{spec.pre-semi}}$, are simulated using Problem 2 with $\beta = 1$. From the definition of these forms, $\vec{k}_{\mathbb{S}}^{*\,\mathsf{spec.pre\text{-}semi}}$ can decrypt $\vec{c}_{\Gamma}^{\,\mathsf{spec.pre\text{-}semi}}$ for any Γ when $\mathbb{S}$ accepts Γ, i.e., it is hard for simulator $\mathcal{B}_2^+$ to tell $(\vec{k}_{\mathbb{S}}^{*\,\mathsf{spec.pre-semi}}, \vec{c}_{\Gamma}^{\,\mathsf{spec.pre-semi}})$ for Game 2-h^+ from $(\vec{k}_{\mathbb{S}}^{*\,\mathsf{norm}}, \vec{c}_{\Gamma}^{\,\mathsf{semi}})$ for Game 2-h under the assumption of Problem 2. On the other hand, $a_0 (= w_0 r_0)$ is independently distributed from the other variables when $\mathbb{S}$ does not accept Γ (shown in Proof of Claim 1 by using Lemma 3). That is, the joint distribution of $\vec{k}_{\mathbb{S}}^{*\,\mathsf{pre-semi}}$ and $\vec{c}_{\Gamma}^{\,\mathsf{pre-semi}}$ is equivalent to that of $\vec{k}_{\mathbb{S}}^{*\,\mathsf{spec.pre-semi}}$ and $\vec{c}_{\Gamma}^{\,\mathsf{spec.pre-semi}}$, when $\mathbb{S}$ does not accept Γ (i.e., $\mathcal{B}_2^+$'s simulation using Problem 2 with $\beta = 1$ is the same distribution as that of Game 2-h^+ from the adversary's view). In other words, w_0 and r_0 in $\vec{k}_{\mathbb{S}}^{*\,\mathsf{spec.pre\text{-}semi}}$ and $\vec{c}_{\Gamma}^{\,\mathsf{spec.pre\text{-}semi}}$ (given by $\mathcal{B}_2^+$'s simulation using Problem 2 with $\beta = 1$) are correlated for the case that $\mathbb{S}$ accepts Γ or for simulator $\mathcal{B}_2^+$'s view, but adversary $\mathcal{A}$ cannot notice the correlation since $\mathcal{A}$'s queries should satisfy the condition that $\mathbb{S}$ does not accept Γ.

The advantage gap between Games 2-h^+ and 2-$(h + 1)$ is similarly shown to be bounded by the advantage of Problem 2, i.e., advantage of the DLIN assumption (Lemmas 6 and 2).

Finally, we show that Game 2-ν can be conceptually changed to Game 3 (Lemma 7).

The game transformations as well as (hierarchical) reductions of Problem 1 and 2 assumptions to the DLIN assumption are summarized in Fig. 1. (For the (hierarchical) reductions, refer to "Appendix B".)

Proof of Theorem 1. To prove Theorem 1, we consider the following $(2\nu + 3)$ 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 which were changed in a game from the previous game.

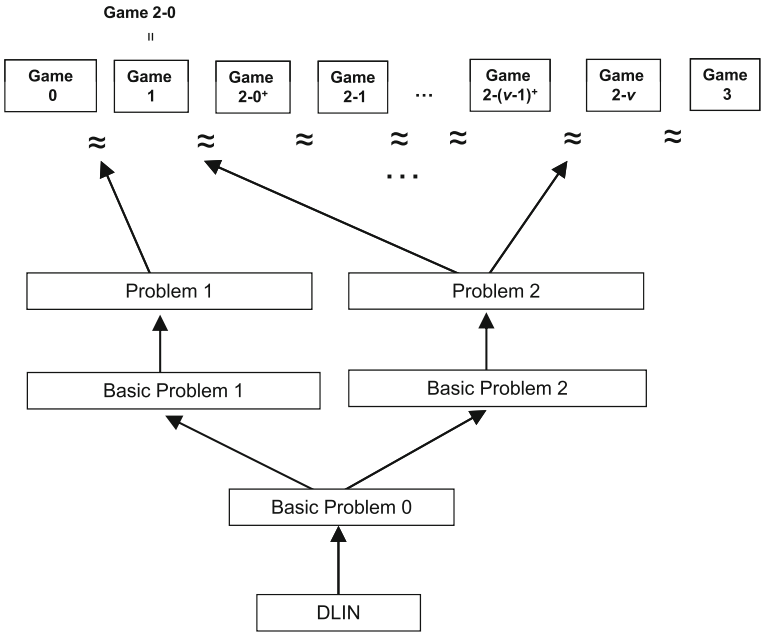

Fig. 1. Structure of reductions for the proposed KP-FE and CP-FE (in Sect. 5) schemes.

Game 0 : Original game. That is, the reply to a key query for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix M is:

$$\left.\begin{array}{l} \boldsymbol{k}_0^* := (-s_0,\ \boxed{0},\ 1,\ \eta_0,\ 0)_{\mathbb{B}_0^*}, \\ \text{for } i = 1, \ldots, \ell, \\ \quad \text{if } \rho(i) = (t, \vec{v}_i),\ \ \boldsymbol{k}_i^* := (s_i \vec{e}_{t,1} + \theta_i \vec{v}_i,\ \boxed{0^{n_t}},\ \vec{\eta}_i,\ 0)_{\mathbb{B}_t^*}, \\ \quad \text{if } \rho(i) = \neg(t, \vec{v}_i),\ \ \boldsymbol{k}_i^* := (s_i \vec{v}_i,\ \boxed{0^{n_t}},\ \vec{\eta}_i,\ 0)_{\mathbb{B}_t^*}, \end{array}\right\} \tag{1}$$

where $\vec{f} \xleftarrow{\cup} \mathbb{F}_q^r$, $\vec{s}^{\mathrm{T}} := (s_1, \ldots, s_\ell)^{\mathrm{T}} := M \cdot \vec{f}^{\mathrm{T}}$, $s_0 := \vec{1} \cdot \vec{f}^{\mathrm{T}}$, $\theta_i, \eta_0 \xleftarrow{\cup} \mathbb{F}_q$, $\vec{\eta}_i \xleftarrow{\cup} \mathbb{F}_q^{n_t}$, $\vec{e}_{t,1} = (1, 0, \ldots, 0) \in \mathbb{F}_q^{n_t}$, and $\vec{v}_i \in \mathbb{F}_q^{n_t} \backslash \{\vec{0}\}$. The challenge ciphertext for challenge plaintexts $(m^{(0)}, m^{(1)})$ and $\Gamma := \{(t, \vec{x}_t) \mid 1 \le t \le d\}$ is:

$$\left.\begin{array}{l} \boldsymbol{c}_0 := (\delta,\ \boxed{0},\ \boxed{\zeta},\ 0,\ \varphi_0)_{\mathbb{B}_0}, \\ \boldsymbol{c}_t := (\delta \vec{x}_t,\ \boxed{0^{n_t}},\ 0^{n_t},\ \varphi_t)_{\mathbb{B}_t} \text{ for } (t, \vec{x}_t) \in \Gamma, \\ c_{d+1} := g_T^{\zeta} m^{(b)}, \end{array}\right\} \tag{2}$$

where $b \xleftarrow{\cup} \{0, 1\}$; $\delta, \zeta, \varphi_0, \varphi_t \xleftarrow{\cup} \mathbb{F}_q$, and $\vec{x}_t \in \mathbb{F}_q^{n_t} \backslash \{\vec{0}\}$.

Game 1 : Same as Game 0 except that the challenge ciphertext is:

$$\boldsymbol{c}_0 := (\delta,\ \boxed{r_0},\ \zeta,\ 0,\ \varphi_0)_{\mathbb{B}_0}, \tag{3}$$

$$\boldsymbol{c}_t := (\delta \vec{x}_t,\ \boxed{\vec{r}_t},\ 0^{n_t},\ \varphi_t)_{\mathbb{B}_t} \text{ for } (t, \vec{x}_t) \in \Gamma, \tag{4}$$

$$c_{d+1} := g_T^{\zeta} m^{(b)}, \tag{5}$$

where $r_0 \xleftarrow{\cup} \mathbb{F}_q, \vec{r}_t \xleftarrow{\cup} \mathbb{F}_q^{n_t}$, and all the other variables are generated as in Game 0.

Game 2-h^+ ($\boldsymbol{h} = \mathbf{0}, \ldots, \boldsymbol{v} - \mathbf{1}$) : Game 2-0 is Game 1. Game 2-h^+ is the same as Game 2-h except the reply to the $(h+1)$th key query for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix M, and $\boldsymbol{c}_t$ of the challenge ciphertext are:

$$\left.\begin{array}{l} \boldsymbol{k}_0^* := (-s_0,\ \boxed{w_0},\ 1,\ \eta_0,\ 0)_{\mathbb{B}_0^*}, \\ \text{for } i = 1, \ldots, \ell, \\ \quad \text{if } \rho(i) = (t, \vec{v}_i), \\ \quad\quad \boldsymbol{k}_i^* := (s_i \vec{e}_{t,1} + \theta_i \vec{v}_i,\ \boxed{(a_i \vec{e}_{t,1} + \pi_i \vec{v}_i) \cdot Z_t},\ \vec{\eta}_i,\ 0)_{\mathbb{B}_t^*}, \\ \quad \text{if } \rho(i) = \neg(t, \vec{v}_i), \\ \quad\quad \boldsymbol{k}_i^* := (s_i \vec{v}_i,\ \boxed{a_i \vec{v}_i \cdot Z_t},\ \vec{\eta}_i,\ 0)_{\mathbb{B}_t^*}, \end{array}\right\} \tag{6}$$

$$\boldsymbol{c}_t := (\delta \vec{x}_t,\ \boxed{\vec{x}_t \cdot U_t},\ 0^{n_t},\ \varphi_t)_{\mathbb{B}_t} \text{ for } (t, \vec{x}_t) \in \Gamma, \tag{7}$$

where $w_0 \xleftarrow{\cup} \mathbb{F}_q$, $\vec{g} \xleftarrow{\cup} \mathbb{F}_q^r$, $\vec{a}^{\mathrm{T}} := (a_1, \ldots, a_\ell)^{\mathrm{T}} := M \cdot \vec{g}^{\mathrm{T}}$, $\pi_i \xleftarrow{\cup} \mathbb{F}_q$ $(i = 1, \ldots, \ell)$, $Z_t \xleftarrow{\cup} GL(n_t, \mathbb{F}_q)$, $U_t := (Z_t^{-1})^{\mathrm{T}}$ for $t = 1, \ldots, d$, and all the other variables are generated as in Game 2-h.

Game 2-($\boldsymbol{h}+\mathbf{1}$) ($\boldsymbol{h} = \mathbf{0}, \ldots, \boldsymbol{v} - \mathbf{1}$) : Game 2-$(h+1)$ is the same as Game 2-h^+ except the reply to the $(h+1)$th key query for $\mathbb{S} := (M, \rho)$ with $\ell \times r$ matrix M, and $\boldsymbol{c}_t$ of the challenge ciphertext are:

$$\left.\begin{array}{l} \boldsymbol{k}_0^* := (-s_0,\ w_0,\ 1,\ \eta_0,\ 0)_{\mathbb{B}_0^*}, \\ \text{for } i = 1, \ldots, \ell, \\ \quad \text{if } \rho(i) = (t, \vec{v}_i),\ \ \boldsymbol{k}_i^* := (s_i \vec{e}_{t,1} + \theta_i \vec{v}_i,\ \boxed{0^{n_t}},\ \vec{\eta}_i,\ 0)_{\mathbb{B}_t^*}, \\ \quad \text{if } \rho(i) = \neg(t, \vec{v}_i),\ \ \boldsymbol{k}_i^* := (s_i \vec{v}_i,\ \boxed{0^{n_t}},\ \vec{\eta}_i,\ 0)_{\mathbb{B}_t^*}, \end{array}\right\}$$

$$\boldsymbol{c}_t := (\delta \vec{x}_t,\ \boxed{\vec{r}_t},\ 0^{n_t},\ \varphi_t)_{\mathbb{B}_t} \text{ for } (t, \vec{x}_t) \in \Gamma, \tag{8}$$

where $\vec{r}_t \xleftarrow{\cup} \mathbb{F}_q^{n_t}$, and all the other variables are generated as in Game 2-h^+.

Game 3 : Same as Game 2-v except that $\boldsymbol{c}_0$ and c_{d+1} of the challenge ciphertext are

$$\boldsymbol{c}_0 := (\delta,\ r_0,\ \boxed{\zeta'},\ 0,\ \varphi_0)_{\mathbb{B}_0}, \quad c_{d+1} := g_T^{\zeta} m^{(b)},$$

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

Let $\mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda)$, $\mathsf{Adv}^{(1)}_{\mathcal{A}}(\lambda)$, $\mathsf{Adv}^{(2-h)}_{\mathcal{A}}(\lambda)$, $\mathsf{Adv}^{(2-h^+)}_{\mathcal{A}}(\lambda)$ and $\mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda)$ be the advantage of $\mathcal{A}$ in Game 0, 1, $2-h$, $2-h^+$ and 3, respectively. $\mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda)$ is equivalent to $\mathsf{Adv}^{\mathsf{KP-FE,PH}}_{\mathcal{A}}(\lambda)$ and it is clear that $\mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda) = 0$ by Lemma 8.

We will show four lemmas (Lemmas 4–7) that evaluate the gaps between pairs of $\mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda)$, $\mathsf{Adv}^{(1)}_{\mathcal{A}}(\lambda)$, $\mathsf{Adv}^{(2-h)}_{\mathcal{A}}(\lambda)$, $\mathsf{Adv}^{(2-h^+)}_{\mathcal{A}}(\lambda)$, $\mathsf{Adv}^{(2-(h+1))}_{\mathcal{A}}(\lambda)$ for $h = 0, \ldots, \nu-1$ and $\mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda)$. From these lemmas and Lemmas 1 and 2 , we obtain

$$
\begin{aligned}
\mathsf{Adv}^{\mathsf{KP\text{-}FE,\ PH}}_{\mathcal{A}}(\lambda) = \mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda) &\le \left|\mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(1)}_{\mathcal{A}}(\lambda)\right| + \sum_{h=0}^{\nu-1}\left|\mathsf{Adv}^{(2-h)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(2-h^+)}_{\mathcal{A}}(\lambda)\right| \\
&+ \sum_{h=0}^{\nu-1}\left|\mathsf{Adv}^{(2-h^+)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(2-(h+1))}_{\mathcal{A}}(\lambda)\right| + \left|\mathsf{Adv}^{(2-\nu)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda)\right| + \mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda) \\
&\le \mathsf{Adv}^{\mathsf{P1}}_{\mathcal{B}_1}(\lambda) + \sum_{h=0}^{\nu-1}\mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}^+_{2,h}}(\lambda) + \sum_{h=0}^{\nu-1}\mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}_{2,h+1}}(\lambda) + (2d\nu + 6\nu + 1)/q \\
&\le \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_1}(\lambda) + \sum_{h=0}^{\nu-1}\left(\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}^+_{2,h}}(\lambda) + \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_{2,h+1}}(\lambda)\right) + (2d\nu + 16\nu + d + 7)/q.
\end{aligned}
$$

This completes the proof of Theorem 1. □

Lemma 4. *For any adversary $\mathcal{A}$, 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, $|\mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda) - \mathit{Adv}^{(1)}_{\mathcal{A}}(\lambda)| \le \mathsf{Adv}^{\mathit{P1}}_{\mathcal{B}_1}(\lambda)$.*

Proof. In order to prove Lemma 4, 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, $(\mathsf{param}_{\vec{n}}, \mathbb{B}_0, \widehat{\mathbb{B}}^*_0, \boldsymbol{e}_{\beta,0}, \{\mathbb{B}_t, \widehat{\mathbb{B}}^*_t, \boldsymbol{e}_{\beta,t,1}, \boldsymbol{e}_{t,j}\}_{t=1,\ldots,d;\, j=2,\ldots,n_t})$.
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 $\mathsf{pk} := (1^\lambda, \mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}_t\}_{t=0,\ldots,d})$ of Game 0 (and 1), where $\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5})$ and $\widehat{\mathbb{B}}_t := (\boldsymbol{b}_{t,1}, .., \boldsymbol{b}_{t,n_t}, \boldsymbol{b}_{t,3n_t+1})$ for $t = 1, .., d$, that are obtained from the Problem 1 instance.
4. When a key query is issued for access structure $\mathbb{S} := (M, \rho)$, $\mathcal{B}_1$ answers normal key $(\boldsymbol{k}^*_0, \ldots, \boldsymbol{k}^*_\ell)$ with Eq. (1), that is computed using $\{\widehat{\mathbb{B}}^*_t\}_{t=0,\ldots,d}$ of the Problem 1 instance.
5. When $\mathcal{B}_1$ receives an encryption query with challenge plaintexts $(m^{(0)}, m^{(1)})$ and $\Gamma := \{(t, \vec{x}_t) \mid 1 \le t \le d\}$ from $\mathcal{A}$, $\mathcal{B}_1$ computes the challenge ciphertext $(\boldsymbol{c}_0, \{\boldsymbol{c}_t\}_{(t,\vec{x}_t)\in\Gamma}, c_{d+1})$ such that

$$\boldsymbol{c}_0 := \boldsymbol{e}_{\beta,0} + \zeta \boldsymbol{b}_{0,3}, \quad \boldsymbol{c}_t := x_{t,1}\boldsymbol{e}_{\beta,t,1} + \textstyle\sum_{j=2}^{n_t} x_{t,j}\boldsymbol{e}_{t,j}, \quad c_{d+1} := g_T^{\zeta} m^{(b)},$$

where $\zeta \xleftarrow{\cup} \mathbb{F}_q$, $b \xleftarrow{\cup} \{0, 1\}$, and $(\boldsymbol{b}_{0,3}, \boldsymbol{e}_{\beta,0}, \{\boldsymbol{e}_{\beta,t,1}, \boldsymbol{e}_{t,j}\}_{t=1,\ldots,d;\, j=2,\ldots,n_t})$ 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 $\beta' := 0$.

It is straightforward that the distribution by $\mathcal{B}_1$'s simulation given a Problem 1 instance with β is equivalent to that in Game 0 (resp. Game 1), when $\beta = 0$ (resp. $\beta = 1$) since $x_{t,1} = 1$. □

Lemma 5. *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_2^+$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $|\mathsf{Adv}_{\mathcal{A}}^{(2-h)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(2-h^+)}(\lambda)| \leq \mathsf{Adv}^{P2}_{\mathcal{B}_{2,h}^+}(\lambda) + (d+3)/q$, where $\mathcal{B}_{2,h}^+(\cdot) := \mathcal{B}_2^+(h, \cdot)$.*

Proof. In order to prove Lemma 5, we construct a probabilistic machine $\mathcal{B}_2^+$ against Problem 2 using an adversary $\mathcal{A}$ in a security game (Game 2-h or 2-h^+) as a black box as follows:

1. $\mathcal{B}_2^+$ is given an integer h and a Problem 2 instance, $(\mathsf{param}_{\vec{n}}, \widehat{\mathbb{B}}_0, \mathbb{B}_0^*, \boldsymbol{h}^*_{\beta,0}, \boldsymbol{e}_0, \{\widehat{\mathbb{B}}_t, \mathbb{B}_t^*, \boldsymbol{h}^*_{\beta,t,j}, \boldsymbol{e}_{t,j}\}_{t=1,\ldots,d;\, j=1,\ldots,n_t})$.
2. $\mathcal{B}_2^+$ plays a role of the challenger in the security game against adversary $\mathcal{A}$.
3. At the first step of the game, $\mathcal{B}_2^+$ provides $\mathcal{A}$ a public key $\mathsf{pk} := (1^\lambda, \mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}'_t\}_{t=0,\ldots,d})$ of Game 2-h (and 2-h^+), where $\widehat{\mathbb{B}}'_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5})$ and $\widehat{\mathbb{B}}'_t := (\boldsymbol{b}_{t,1}, .., \boldsymbol{b}_{t,n_t}, \boldsymbol{b}_{t,3n_t+1})$ for $t = 1, .., d$, that are obtained from the Problem 2 instance.
4. When the ιth key query is issued for access structure $\mathbb{S} := (M, \rho)$, $\mathcal{B}_2^+$ answers as follows:
   (a) When $1 \leq \iota \leq h$, $\mathcal{B}_2^+$ answers semi-functional key $(\boldsymbol{k}_0^*, \ldots, \boldsymbol{k}_\ell^*)$ with Eq. (8), that is computed using $\{\mathbb{B}_t^*\}_{t=0,\ldots,d}$ of the Problem 2 instance.
   (b) When $\iota = h+1$, $\mathcal{B}_2^+$ calculates $(\boldsymbol{k}_0^*, \ldots, \boldsymbol{k}_\ell^*)$ using $(\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{h}^*_{\beta,0}, \{\boldsymbol{b}^*_{t,j}, \boldsymbol{h}^*_{\beta,t,j}\}_{t=1,\ldots,d;\, j=1,\ldots,n_t})$ of the Problem 2 instance as follows:

$$
\begin{aligned}
&\pi_t, \mu_t, g_k, \widetilde{\mu}_k \xleftarrow{\cup} \mathbb{F}_q \quad \text{for } t = 1, \ldots, d;\ k = 1, \ldots, r,\\
&\widetilde{\boldsymbol{p}}^*_{\beta,0} := \textstyle\sum_{k=1}^{r} \left(g_k \boldsymbol{h}^*_{\beta,0} + \widetilde{\mu}_k \boldsymbol{b}^*_{0,1} \right),\\
&\text{for } t = 1, \ldots, d;\ k = 1, \ldots, r;\ j = 1, \ldots, n_t;\\
&\quad \boldsymbol{p}^*_{\beta,t,j} := \pi_t \boldsymbol{h}^*_{\beta,t,j} + \mu_t \boldsymbol{b}^*_{t,j}, \qquad \widetilde{\boldsymbol{p}}^*_{\beta,t,k,j} := g_k \boldsymbol{h}^*_{\beta,t,j} + \widetilde{\mu}_k \boldsymbol{b}^*_{t,j},\\
&\boldsymbol{k}_0^* := -\widetilde{\boldsymbol{p}}^*_{\beta,0} + \boldsymbol{b}^*_{0,3},\\
&\text{for } i = 1, \ldots, \ell,\\
&\quad \text{if } \rho(i) = (t, \vec{v}_i), \quad \boldsymbol{k}_i^* := \textstyle\sum_{j=1}^{n_t} v_{i,j} \boldsymbol{p}^*_{\beta,t,j} + \sum_{k=1}^{r} M_{i,k} \widetilde{\boldsymbol{p}}^*_{\beta,t,k,1},\\
&\quad \text{if } \rho(i) = \neg(t, \vec{v}_i), \quad \boldsymbol{k}_i^* := \textstyle\sum_{j=1}^{n_t} v_{i,j} (\sum_{k=1}^{r} M_{i,k} \widetilde{\boldsymbol{p}}^*_{\beta,t,k,j}),
\end{aligned}
$$

   where $(M_{i,k})_{i=1,\ldots,\ell;k=1,\ldots,r} := M$.
   (c) When $\iota \geq h+2$, $\mathcal{B}_2^+$ answers normal key $(\boldsymbol{k}_0^*, \ldots, \boldsymbol{k}_\ell^*)$ with Eq. (1), that is computed using $\{\mathbb{B}_t^*\}_{t=0,\ldots,d}$ of the Problem 2 instance.

5. When $\mathcal{B}_2^+$ receives an encryption query with challenge plaintexts $(m^{(0)}, m^{(1)})$ and $\Gamma := \{(t, \vec{x}_t) \mid 1 \le t \le d\}$ from $\mathcal{A}$, $\mathcal{B}_2^+$ computes the challenge ciphertext $(\boldsymbol{c}_0, \{\boldsymbol{c}_t\}_{(t,\vec{x}_t)\in\Gamma}, c_{d+1})$ such that for $(t, \vec{x}_t) \in \Gamma$,

$$\boldsymbol{c}_0 := \boldsymbol{e}_0 + \zeta \boldsymbol{b}_{0,3} + \boldsymbol{q}_0, \quad \boldsymbol{c}_t := \textstyle\sum_{j=1}^{n_t} x_{t,j} \boldsymbol{e}_{t,j} + \boldsymbol{q}_t, \quad c_{d+1} := g_T^{\zeta} m^{(b)},$$

where $\zeta \xleftarrow{\cup} \mathbb{F}_q$, $b \xleftarrow{\cup} \{0,1\}$, $\boldsymbol{q}_0 \xleftarrow{\cup} \mathsf{span}\langle \boldsymbol{b}_{0,5}\rangle$, $\boldsymbol{q}_t \xleftarrow{\cup} \mathsf{span}\langle \boldsymbol{b}_{t,3n_t+1}\rangle$, and $(\boldsymbol{b}_{0,3}, \boldsymbol{e}_0, \{\boldsymbol{e}_{t,j}\}_{t=1,..,d;\, j=1,..,n_t})$ is a part of the Problem 2 instance.
6. When a key query is issued by $\mathcal{A}$ after the encryption query, $\mathcal{B}_2^+$ executes the same procedure as that of step 4.3.
7. $\mathcal{A}$ finally outputs bit b'. If $b = b'$, $\mathcal{B}_2^+$ outputs $\beta' := 1$. Otherwise, $\mathcal{B}_2^+$ outputs $\beta' := 0$.

Remark 3. $\widetilde{\boldsymbol{p}}^*_{\beta,0}, \boldsymbol{p}^*_{\beta,t,j}, \widetilde{\boldsymbol{p}}^*_{\beta,t,k,j}$ for $t = 1, \ldots, d; k = 1, \ldots, r; j = 1, \ldots, n_t$ calculated in case (b) of steps 4 and 6 in the above simulation are expressed as:

$$\theta_t := \pi_t \delta + \mu_t,\ f_k := g_k \delta + \widetilde{\mu}_k,\ s_0 := \textstyle\sum_{k=1}^{r} f_k,\ a_0 := \sum_{k=1}^{r} g_k, w_0 := a_0/z_0 (= a_0 u_0),$$
$$\widetilde{\boldsymbol{p}}^*_{0,0} = (s_0, 0, 0, a_0\delta_0, 0)_{\mathbb{B}_0^*},\quad \widetilde{\boldsymbol{p}}^*_{1,0} = (s_0, w_0, 0, a_0\delta_0, 0)_{\mathbb{B}_0^*},$$

$$\begin{array}{lccccc}
 & \overbrace{\quad}^{n_t} & \overbrace{\quad}^{n_t} & \overbrace{\quad}^{n_t} & \overbrace{\quad}^{1} & \\
\boldsymbol{p}^*_{0,t,j} := (& \theta_t \vec{e}_{t,j}, & 0^{n_t}, & \pi_t \vec{\delta}_{t,j}, & 0 &)_{\mathbb{B}_t^*}, \\
\widetilde{\boldsymbol{p}}^*_{0,t,k,j} := (& f_k \vec{e}_{t,j}, & 0^{n_t}, & g_k \vec{\delta}_{t,j}, & 0 &)_{\mathbb{B}_t^*}, \\
\boldsymbol{p}^*_{1,t,j} := (& \theta_t \vec{e}_{t,j}, & \pi_t \vec{u}_{t,j}, & \pi_t \vec{\delta}_{t,j}, & 0 &)_{\mathbb{B}_t^*}, \\
\widetilde{\boldsymbol{p}}^*_{1,t,k,j} := (& f_k \vec{e}_{t,j}, & g_k \vec{u}_{t,j}, & g_k \vec{\delta}_{t,j}, & 0 &)_{\mathbb{B}_t^*},
\end{array}$$

where $\delta, z_0, \delta_0, \{\vec{e}_{t,j}, \vec{u}_{t,j}, \vec{\delta}_{t,j}\}_{t=1,..,d;\, j=1,..,n_t}$ are defined in Problem 2. Note that variables $\{\theta_t, \pi_t\}_{t=1,..,d}, \{f_k, g_k\}_{k=1,..,r}$ are independently and uniformly distributed. Therefore, $\{\boldsymbol{k}_i^*\}_{i=0,..,\ell}$ are distributed as Eq. (6) except $w_0 := a_0/r_0$, i.e., $w_0 r_0 = a_0$, using a_0 and $r_0 := z_0 \xleftarrow{\cup} \mathbb{F}_q$ in $\boldsymbol{c}_0$ (Eq. 3).

Claim 1. *The distribution of the view of adversary $\mathcal{A}$ in the above-mentioned game simulated by $\mathcal{B}_2^+$ given a Problem 2 instance with $\beta \in \{0,1\}$ is the same as that in Game 2-h (resp. Game 2-h^+) if $\beta = 0$ (resp. $\beta = 1$) except with probability $(d+2)/q$ (resp. $1/q$).*

Proof. It is clear that $\mathcal{B}_2^+$'s simulation of the public key generation (step 4.3) and the ιth key query's answer for $\iota \neq h+1$ (cases (a) and (c) of steps 4.3 and 6) is perfect, i.e., exactly the same as the Setup and the KeyGen oracle in Game 2-h and Game 2-h^+.

Therefore, to prove this lemma we will show that the joint distribution of the $(h+1)$-the key query's answer and the challenge ciphertext by $\mathcal{B}_2^+$'s simulation given a Problem 2 instance with β is equivalent to that in Game 2-h (resp. Game 2-h^+), when $\beta = 0$ (resp. $\beta = 1$).

When $\beta = 0$, it is straightforward to show that they are equivalent except that δ defined in Problem 2 is zero or there exists $t \in \{0, \ldots, d\}$ such that $\vec{r}_t = \vec{0}$, where $\vec{r}_t$ are defined in Eqs. (3) and (4), i.e., except with probability $(d+2)/q$.

When $\beta = 1$, the distribution by $\mathcal{B}_2^+$'s simulation is Eq. (6) for the key and Eqs. (3), (5), and (7) for the challenge ciphertext, where the distribution is the same as that defined in these equations except $w_0 := a_0/r_0$, i.e., $w_0 r_0 = a_0$, using $a_0 := \vec{1} \cdot \vec{g}^{\mathrm{T}}$ and $r_0 \xleftarrow{\cup} \mathbb{F}_q$ in $\boldsymbol{c}_0$ (Eq. 3) from Remark 3. The corresponding distribution in Game 2-h^+ is Eq. (6) and Eqs. (3), (5), and (7) where r_0, $w_0 \xleftarrow{\cup} \mathbb{F}_q$ as defined in the equations.

Therefore, we will show that a_0 is uniformly and independently distributed from the other variables in the joint distribution of $\mathcal{B}_2^+$'s simulation. Since $a_0 := \vec{1} \cdot \vec{g}^{\mathrm{T}}$ is only related to $(a_1, \ldots, a_\ell)^{\mathrm{T}} := M \cdot \vec{g}^{\mathrm{T}}$ and $U_t = (Z_t^{-1})^{\mathrm{T}}$ holds, a_0 is only related to $\{\vec{w}_i\}_{i=1,\ldots,\ell}$, $\{\vec{\overline{w}}_i\}_{i=1,\ldots,\ell}$ and $\{\vec{r}_t\}_{t=1,\ldots,d}$, where $\vec{w}_i := (a_i \vec{e}_{t,1} + \pi_i \vec{v}_i) \cdot Z_t := ((a_i, 0, \ldots, 0) + \pi_i \vec{v}_i) \cdot Z_t$ and $\vec{\overline{w}}_i := a_i \vec{v}_i \cdot Z_t$ in Eq. (6) for $i = 1, \ldots, \ell$, and $\vec{r}_t := \vec{x}_t \cdot U_t$ in Eq. (7) for $t = 1, \ldots, d$ with $t := \widetilde{\rho}(i)$. ($\widetilde{\rho}$ is defined at the start of Sect. 4.) With respect to the joint distribution of these variables, there are five cases for each $i \in \{1, \ldots, \ell\}$. Note that for any $i \in \{1, \ldots, \ell\}$, (Z_t, U_t) with $t := \widetilde{\rho}(i)$ is independent from the other variables, since $\widetilde{\rho}$ is injective:

1. $\gamma(i) = 1$ and $[\rho(i) = (t, \vec{v}_i) \ \wedge \ (t, \vec{x}_t) \in \Gamma \ \wedge \ \vec{v}_i \cdot \vec{x}_t = 0]$.
   Then, from Lemma 3, the joint distribution of $(\vec{w}_i, \vec{r}_t)$ is uniformly and independently distributed on $C_{a_i} := \{(\vec{w}, \vec{r}) | \vec{w} \cdot \vec{r} = a_i\}$ (over $Z_t \xleftarrow{\cup} GL(n_t, \mathbb{F}_q)$).
2. $\gamma(i) = 1$ and $[\rho(i) = \neg(t, \vec{v}_i) \ \wedge \ (t, \vec{x}_t) \in \Gamma \ \wedge \vec{v}_i \cdot \vec{x}_t \neq 0]$.
   Then, from Lemma 3, the joint distribution of $(\vec{\overline{w}}_i, \vec{r}_t)$ is uniformly and independently distributed on $C_{(\vec{v}_i \cdot \vec{x}_t) \cdot a_i}$ (over $Z_t \xleftarrow{\cup} GL(n_t, \mathbb{F}_q)$).
3. $\gamma(i) = 0$ and $[\rho(i) = (t, \vec{v}_i) \ \wedge \ (t, \vec{x}_t) \in \Gamma]$ (i.e., $\vec{v}_i \cdot \vec{x}_t \neq 0$).
   Then, from Lemma 3, the joint distribution of $(\vec{w}_i, \vec{r}_t)$ is uniformly and independently distributed on $C_{(\vec{v}_i \cdot \vec{x}_t) \cdot \pi_t + a_i}$ (over $Z_t \xleftarrow{\cup} GL(n_t, \mathbb{F}_q)$) where π_t is defined in Remark 3. Since π_t is uniformly and independently distributed on $\mathbb{F}_q$, the joint distribution of $(\vec{w}_i, \vec{r}_t)$ is uniformly and independently distributed over $\mathbb{F}_q^{2n_t}$.
4. $\gamma(i) = 0$ and $[\rho(i) = \neg(t, \vec{v}_i) \ \wedge \ (t, \vec{x}_t) \in \Gamma]$ (i.e., $\vec{v}_i \cdot \vec{x}_t = 0$).
   Then, from Lemma 3, the joint distribution of $(\vec{\overline{w}}_i, \vec{r}_t)$ is uniformly and independently distributed on C_0 (over $Z_t \xleftarrow{\cup} GL(n_t, \mathbb{F}_q)$).
5. $[\rho(i) = (t, \vec{v}_i) \ \wedge \ (t, \vec{x}_t) \notin \Gamma]$ or $[\rho(i) = \neg(t, \vec{v}_i) \ \wedge \ (t, \vec{x}_t) \notin \Gamma]$.
   Then, the distribution of $\vec{w}_i$ is uniformly and independently distributed on $\mathbb{F}_q^{n_t}$ (over $Z_t \xleftarrow{\cup} GL(n_t, \mathbb{F}_q)$).

We then observe the joint distribution (or relation) of a_0, $\{\vec{w}_i\}_{i=1,\ldots,\ell}$, $\{\vec{\overline{w}}_i\}_{i=1,\ldots,\ell}$ and $\{\vec{r}_t\}_{t=1,\ldots,d}$. Those in cases 3-5 are obviously independent from a_0. Due to the restriction of adversary $\mathcal{A}$'s key queries, $\vec{1} \notin \mathsf{span}\langle (M_i)_{\gamma(i)=1} \rangle$. Therefore, $a_0 := \vec{1} \cdot \vec{g}^{\mathrm{T}}$ is independent from the joint distribution of $\{a_i := M_i \cdot \vec{g}^{\mathrm{T}} \mid \gamma(i) = 1\}$ (over the random selection of $\vec{g}$), which can be given by $(\vec{w}_i, \vec{r}_t)$ in case 1 and $(\vec{\overline{w}}_i, \vec{r}_t)$ in case 2. Thus, a_0 is uniformly and independently distributed from the other variables in the joint distribution of $\mathcal{B}_2^+$'s simulation.

Therefore, the view of adversary $\mathcal{A}$ in the game simulated by $\mathcal{B}_2^+$ given a Problem 2 instance with $\beta = 1$ is the same as that in Game 2-h^+ except that δ defined in Problem 2 is zero, i.e., except with probability $1/q$. □

This completes the proof of Lemma 5. □

Lemma 6. *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_2$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $|\mathsf{Adv}_{\mathcal{A}}^{(2-h^+)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(2-(h+1))}(\lambda)| \leq \mathsf{Adv}_{\mathcal{B}_{2,h+1}}^{P2}(\lambda) + (d+3)/q$, where $\mathcal{B}_{2,h+1}(\cdot) := \mathcal{B}_2(h, \cdot)$.*

Proof. In order to prove Lemma 6, we construct a probabilistic machine $\mathcal{B}_2$ against Problem 2 using an adversary $\mathcal{A}$ in a security game (Game 2-h^+ or 2-$(h+1)$) as a black box. $\mathcal{B}_2$ acts in the same way as $\mathcal{B}_2^+$ in the proof of Lemma 5 except the following two points:

1. In case (b) of step 4; $\boldsymbol{k}_0^*$ is calculated as

$$\boldsymbol{k}_0^* := -\widetilde{\boldsymbol{p}}_{\beta,0}^* + r_0' \boldsymbol{b}_{0,2}^* + \boldsymbol{b}_{0,3}^*,$$

   where $r_0' \xleftarrow{\cup} \mathbb{F}_q$, $\widetilde{\boldsymbol{p}}_{\beta,0}^*$ is calculated from $\boldsymbol{h}_{\beta,0}^*$ and $\boldsymbol{b}_{0,1}^*$ as in the proof of Lemma 5, and $\mathbb{B}^* := (\boldsymbol{b}_{0,1}^*, \boldsymbol{b}_{0,2}^*, \boldsymbol{b}_{0,3}^*)$ is in the Problem 2 instance.
2. In the last step; if $b = b'$, $\mathcal{B}_2$ outputs $\beta' := 0$. Otherwise, $\mathcal{B}_2$ outputs $\beta' := 1$.

When $\beta = 0$, it is straightforward that the distribution by $\mathcal{B}_2$'s simulation is equivalent to that in Game 2-$(h+1)$ except that δ defined in Problem 2 is zero, i.e., except with probability $1/q$. When $\beta = 1$, the distribution by $\mathcal{B}_2$'s simulation is equivalent to that in Game 2-h^+ except that δ defined in Problem 2 is zero or there exists $t \in \{0, \ldots, d\}$ such that $\vec{r}_t = \vec{0}$ are defined in Eqs. (3) and (4), i.e., except with probability $(d+2)/q$. □

Lemma 7. *For any adversary $\mathcal{A}$, $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda) \leq \mathsf{Adv}_{\mathcal{A}}^{(2-\nu)}(\lambda) + 1/q$.*

Proof. To prove Lemma 7, we will show distribution $(\mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}_t\}_{t=0,\ldots,d}, \{\mathsf{sk}_{\mathbb{S}}^{(j)*}\}_{j=1,\ldots,\nu}, \boldsymbol{c})$ in Game 2-ν and that in Game 3 are equivalent, where $\mathsf{sk}_{\mathbb{S}}^{(j)*}$ is the answer to the jth key query, and $\boldsymbol{c}$ is the challenge ciphertext. By definition, we only need to consider elements on $\mathbb{V}_0$ or $\mathbb{V}_0^*$. We define new bases $\mathbb{D}_0$ of $\mathbb{V}_0$ and $\mathbb{D}_0^*$ of $\mathbb{V}_0^*$ as follows: We generate $\theta \xleftarrow{\cup} \mathbb{F}_q$, and set

$$\boldsymbol{d}_{0,2} := (0, 1, -\theta, 0, 0)_{\mathbb{B}} = \boldsymbol{b}_{0,2} - \theta \boldsymbol{b}_{0,3}, \quad \boldsymbol{d}_{0,3}^* := (0, \theta, 1, 0, 0)_{\mathbb{B}} = \boldsymbol{b}_{0,3}^* + \theta \boldsymbol{b}_{0,2}^*.$$

We set $\mathbb{D}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{d}_{0,2}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,4}, \boldsymbol{b}_{0,5})$, $\mathbb{D}_0^* := (\boldsymbol{b}_{0,1}^*, \boldsymbol{b}_{0,2}^*, \boldsymbol{d}_{0,3}^*, \boldsymbol{b}_{0,4}^*, \boldsymbol{b}_{0,5}^*)$. We then easily verify that $\mathbb{D}_0$ and $\mathbb{D}_0^*$ are dual orthonormal, and are distributed the same as the original bases, $\mathbb{B}_0$ and $\mathbb{B}_0^*$.

The $\mathbb{V}_0$ components $(\{\boldsymbol{k}_0^{(j)*}\}_{j=1,\ldots,\nu}, \boldsymbol{c}_0)$ in keys and challenge ciphertext $(\{\mathsf{sk}_{\mathbb{S}}^{(j)*}\}_{j=1,\ldots,\nu}, \mathsf{ct}_\Gamma)$ in Game 2-ν are expressed over bases $\mathbb{B}_0$ and $\mathbb{B}_0^*$ as $\boldsymbol{k}_0^{(j)*} = (-s_0^{(j)}, w_0^{(j)}, 1, \eta_0^{(j)}, 0)_{\mathbb{B}_0^*}$, $\boldsymbol{c}_0 = (\delta, r_0, \zeta, 0, \varphi_0)_{\mathbb{B}_0}$. Then,

$$\boldsymbol{k}_0^{(j)*} = \left(-s_0^{(j)}, w_0^{(j)}, 1, \eta_0^{(j)}, 0\right)_{\mathbb{B}_0^*} = \left(-s_0^{(j)}, w_0^{(j)} + \theta, 1, \eta_0^{(j)}, 0\right)_{\mathbb{D}_0^*} = \left(-s_0^{(j)}, \vartheta_0^{(j)}, 1, \eta_0^{(j)}, 0\right)_{\mathbb{D}_0^*},$$

where $\vartheta_0^{(j)} := w_0^{(j)} + \theta$ which are uniformly, independently distributed since $w_0^{(j)} \xleftarrow{\cup} \mathbb{F}_q$.

$$\boldsymbol{c}_0 = (\delta, r_0, \zeta, 0, \varphi_0)_{\mathbb{B}_0} = (\delta, r_0, \zeta + r_0\theta, 0, \varphi_0)_{\mathbb{D}_0} = (\delta, r_0, \zeta', 0, \varphi_0)_{\mathbb{D}_0}$$

where $\zeta' := \zeta + r_0\theta$ which is uniformly, independently distributed since $\theta \xleftarrow{\cup} \mathbb{F}_q$.

In the light of the adversary's view, both $(\mathbb{B}_0, \mathbb{B}_0^*)$ and $(\mathbb{D}_0, \mathbb{D}_0^*)$ are consistent with public key $\mathsf{pk} := (1^\lambda, \mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}_t\}_{t=0,\ldots,d})$. Therefore, $\{\mathsf{sk}_{\mathbb{S}}^{(j)*}\}_{j=1,\ldots,\nu}$ and ct_Γ can be expressed as keys and ciphertext in two ways, in Game 2-ν over bases $(\mathbb{B}_0, \mathbb{B}_0^*)$ and in Game 3 over bases $(\mathbb{D}_0, \mathbb{D}_0^*)$. Thus, Game 2-$\nu$ can be conceptually changed to Game 3 if $r_0 \neq 0$, i.e., except with probability $1/q$. □

Lemma 8. *For any adversary $\mathcal{A}$, $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda) = 0$.*

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

5. CP-FE Scheme

This section presents a CP-FE scheme with the large class of relations, which is defined in Sect. 3.3.

5.1. *Construction*

$\widetilde{\rho} : \{1, \ldots, \ell\} \to \{1, \ldots, d\}$ is defined at the start of Sect. 4. In the proposed scheme, we assume that $\widetilde{\rho}$ is injective for $\mathbb{S} := (M, \rho)$ with ciphertext $\mathsf{ct}_{\mathbb{S}}$. We will show how to relax the restriction in "Appendix E".

In the description of the scheme, we assume that input vector $\vec{x}_t := (x_{t,1}, \ldots, x_{t,n_t})$ is normalized such that $x_{t,1} := 1$. (If $\vec{x}_t$ is not normalized, change it to a normalized one by $(1/x_{t,1}) \cdot \vec{x}_t$ assuming that $x_{t,1}$ is nonzero). In addition, we assume that input vector $\vec{v}_i := (v_{i,1}, \ldots, v_{i,n_t})$ satisfies that $v_{i,n_t} \neq 0$.

Random dual basis generator $\mathcal{G}_{\mathsf{ob}}(1^\lambda, \vec{n})$ is defined at the end of Sect. 2.1. We refer to Sect. 1.5 for notations on DPVS.

$$\begin{aligned}
&\mathsf{Setup}(1^\lambda, \vec{n} := (d; n_1, \ldots, n_d)) : \quad (\mathsf{param}_{\vec{n}}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0,\ldots,d}) \xleftarrow{R} \mathcal{G}_{\mathsf{ob}}(1^\lambda, \vec{n}),\\
&\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5}), \quad \widehat{\mathbb{B}}_t := (\boldsymbol{b}_{t,1}, \ldots, \boldsymbol{b}_{t,n_t}, \boldsymbol{b}_{t,3n_t+1}) \quad \text{for } t = 1, \ldots, d,\\
&\widehat{\mathbb{B}}_0^* := (\boldsymbol{b}_{0,1}^*, \boldsymbol{b}_{0,3}^*, \boldsymbol{b}_{0,4}^*), \quad \widehat{\mathbb{B}}_t^* := (\boldsymbol{b}_{t,1}^*, \ldots, \boldsymbol{b}_{t,n_t}^*, \boldsymbol{b}_{t,2n_t+1}^*, \ldots, \boldsymbol{b}_{t,3n_t}^*) \quad \text{for } t = 1, \ldots, d,
\end{aligned}$$

$\mathsf{pk} := (1^\lambda, \mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}_t\}_{t=0,\ldots,d}),\ \ \mathsf{sk} := \{\widehat{\mathbb{B}}^*_t\}_{t=0,\ldots,d},$

return pk, sk.

$\mathsf{KeyGen}(\mathsf{pk},\ \mathsf{sk},\ \ \Gamma := \{(t, \vec{x}_t := (x_{t,1}, \ldots, x_{t,n_t}) \quad \in \mathbb{F}_q^{n_t} \backslash \{\vec{0}\}) \mid 1 \le t \le d,\ x_{t,1} := 1\}) :$

$\delta, \varphi_0 \xleftarrow{\cup} \mathbb{F}_q, \vec{\varphi}_t \xleftarrow{\cup} \mathbb{F}_q^{n_t}$ such that $(t, \vec{x}_t) \in \Gamma$,

$\boldsymbol{k}_0 := (\delta,\ 0,\ 1,\ \varphi_0,\ 0)_{\mathbb{B}^*_0}$,

$\boldsymbol{k}^*_t := (\overbrace{\quad \delta\vec{x}_t, \quad}^{n_t} \overbrace{\quad 0^{n_t}, \quad}^{n_t} \overbrace{\quad \vec{\varphi}_t, \quad}^{n_t} \overbrace{\quad 0 \quad}^{1})_{\mathbb{B}^*_t}$ for $(t, \vec{x}_t) \in \Gamma$,

$\mathsf{sk}_\Gamma := (\Gamma, \boldsymbol{k}^*_0, \{\boldsymbol{k}^*_t\}_{(t,\vec{x}_t)\in\Gamma})$,

return sk_Γ.

$\mathsf{Enc}(\mathsf{pk},\ m,\ \mathbb{S} := (M, \rho)) :$

$\vec{f} \xleftarrow{R} \mathbb{F}_q^r,\ \vec{s}^{\mathrm{T}} := (s_1, \ldots, s_\ell)^{\mathrm{T}} := M \cdot \vec{f}^{\mathrm{T}},\quad s_0 := \vec{1} \cdot \vec{f}^{\mathrm{T}},\ \eta_0, \eta_i, \theta_i, \zeta \xleftarrow{\cup} \mathbb{F}_q\ (i = 1, .., \ell)$,

$\boldsymbol{c}_0 := (-s_0,\ 0,\ \zeta,\ 0,\ \eta_0)_{\mathbb{B}_0}$,

for $i = 1, \ldots, \ell$,

if $\rho(i) = (t, \vec{v}_i := (v_{i,1}, \ldots, v_{i,n_t}) \in \mathbb{F}_q^{n_t} \backslash \{\vec{0}\})\ (v_{i,n_t} \neq 0)$,

$\boldsymbol{c}_i := (\overbrace{\quad s_i\vec{e}_{t,1} + \theta_i\vec{v}_i, \quad}^{n_t} \overbrace{\quad 0^{n_t}, \quad}^{n_t} \overbrace{\quad 0^{n_t}, \quad}^{n_t} \overbrace{\quad \eta_i \quad}^{1})_{\mathbb{B}_t}$,

if $\rho(i) = \neg(t, \vec{v}_i)$,

$\boldsymbol{c}_i := (\overbrace{\quad s_i\vec{v}_i, \quad}^{n_t} \overbrace{\quad 0^{n_t}, \quad}^{n_t} \overbrace{\quad 0^{n_t}, \quad}^{n_t} \overbrace{\quad \eta_i \quad}^{1})_{\mathbb{B}_t}$,

$c_{d+1} := g_T^\zeta m,\ \ \mathsf{ct}_\mathbb{S} := (\mathbb{S}, \boldsymbol{c}_0, \boldsymbol{c}_1, \ldots, \boldsymbol{c}_\ell, c_{d+1})$,

return $\mathsf{ct}_\mathbb{S}$.

$\mathsf{Dec}(\mathsf{pk},\ \mathsf{sk}_\Gamma := (\Gamma, \boldsymbol{k}^*_0, \{\boldsymbol{k}^*_t\}_{(t,\vec{x}_t)\in\Gamma}),\ \mathsf{ct}_\mathbb{S} := (\mathbb{S}, \boldsymbol{c}_0, \boldsymbol{c}_1, \ldots, \boldsymbol{c}_\ell, c_{d+1})) :$

If $\mathbb{S} := (M, \rho)$ accepts $\Gamma := \{(t, \vec{x}_t)\}$, then compute I and $\{\alpha_i\}_{i\in I}$ such that

$\vec{1} = \sum_{i\in I} \alpha_i M_i$, where M_i is the ith row of M, and

$I \subseteq \{i \in \{1, \ldots, \ell\} \mid [\rho(i) = (t, \vec{v}_i) \ \wedge\ (t, \vec{x}_t) \in \Gamma \ \wedge\ \vec{v}_i \cdot \vec{x}_t = 0]$

$\qquad \vee\ [\rho(i) = \neg(t, \vec{v}_i) \ \wedge\ (t, \vec{x}_t) \in \Gamma \ \wedge\ \vec{v}_i \cdot \vec{x}_t \neq 0]\ \}$,

$K := e(\boldsymbol{c}_0, \boldsymbol{k}^*_0) \prod_{i\in I\ \wedge\ \rho(i)=(t,\vec{v}_i)} e(\boldsymbol{c}_i, \boldsymbol{k}^*_t)^{\alpha_i} \cdot \prod_{i\in I\ \wedge\ \rho(i)=\neg(t,\vec{v}_i)} e(\boldsymbol{c}_i, \boldsymbol{k}^*_t)^{\alpha_i/(\vec{v}_i\cdot\vec{x}_t)}$,

return $m' := c_{d+1}/K$.

[Correctness] If $\mathbb{S} := (M, \rho)$ accepts $\Gamma := \{(t, \vec{x}_t)\}$,

$$
\begin{aligned}
& e(\boldsymbol{c}_0, \boldsymbol{k}^*_0) \textstyle\prod_{i\in I\ \wedge\ \rho(i)=(t,\vec{v}_i)} e(\boldsymbol{c}_i, \boldsymbol{k}^*_t)^{\alpha_i} \cdot \prod_{i\in I\ \wedge\ \rho(i)=\neg(t,\vec{v}_i)} e(\boldsymbol{c}_i, \boldsymbol{k}^*_t)^{\alpha_i/(\vec{v}_i\cdot\vec{x}_t)} \\
&= g_T^{-\delta s_0+\zeta} \textstyle\prod_{i\in I\ \wedge\ \rho(i)=(t,\vec{v}_i)} g_T^{\delta\alpha_i s_i} \prod_{i\in I\ \wedge\ \rho(i)=\neg(t,\vec{v}_i)} g_T^{\delta\alpha_i s_i(\vec{v}_i\cdot\vec{x}_t)/(\vec{v}_i\cdot\vec{x}_t)} \\
&= g_T^{\delta(-s_0+\sum_{i\in I}\alpha_i s_i)+\zeta} = g_T^\zeta.
\end{aligned}
$$

5.2. *Security*

We can prove adaptively payload-hiding security for the CP-FE scheme similarly as the proposed KP-FE case (Theorem 1).

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

For any adversary $\mathcal{A}$, there exist probabilistic machines $\mathcal{E}_1, \mathcal{E}_2^+, \mathcal{E}_2$, whose running times are essentially the same as that of $\mathcal{A}$, such that for any security parameter λ,

$$\mathsf{Adv}_{\mathcal{A}}^{\mathit{CP\text{-}FE,PH}}(\lambda) \leq \mathsf{Adv}_{\mathcal{E}_1}^{\mathsf{DLIN}}(\lambda) + \sum_{h=0}^{\nu-1} \left(\mathsf{Adv}_{\mathcal{E}_{2,h}^+}^{\mathsf{DLIN}}(\lambda) + \mathsf{Adv}_{\mathcal{E}_{2,h+1}}^{\mathsf{DLIN}}(\lambda) \right) + \epsilon,$$

where $\mathcal{E}_{2,h}^+(\cdot) := \mathcal{E}_2^+(h, \cdot)$, $\mathcal{E}_{2,h+1}(\cdot) := \mathcal{E}_2(h, \cdot)$ $(h = 0, \ldots, \nu - 1)$, ν is the maximum number of $\mathcal{A}$'s key queries and $\epsilon := (2d\nu + 16\nu + 2d + 8)/q$.

Proof Outline of Theorem 2: As in the proof of Theorem 1, we follow the dual system encryption methodology proposed by Waters [49], at the top level of strategy of the security proof. In addition, the description of the game transformation is very similar to that of Theorem 1, and the three forms of ciphertexts and secret keys, *normal*, *semi-functional*, and *pre-semi-functional*, are also used as before. Therefore, here, we only describe these forms of ciphertexts and secret keys for the proof of Theorem 2.

For $\mathsf{sk}_\Gamma := (\Gamma, \boldsymbol{k}_0^*, \{\boldsymbol{k}_t^*\}_{(t,\vec{x}_t)\in\Gamma})$ and $\mathsf{ct}_\mathbb{S} := (\mathbb{S}, \boldsymbol{c}_0, \boldsymbol{c}_1, \ldots, \boldsymbol{c}_\ell, c_{d+1})$, we focus on $\vec{\boldsymbol{k}}_\Gamma^* := (\boldsymbol{k}_0^*, \{\boldsymbol{k}_t^*\}_{(t,\vec{x}_t)\in\Gamma})$ and $\vec{\boldsymbol{c}}_\mathbb{S} := (\boldsymbol{c}_0, \boldsymbol{c}_1, \ldots, \boldsymbol{c}_\ell)$, and ignore the other part of sk_Γ and $\mathsf{ct}_\mathbb{S}$ (and call them secret key and ciphertext, respectively) in this proof outline.

A *normal* secret key, $\vec{\boldsymbol{k}}_\Gamma^{*\,\mathsf{norm}}$ (with attribute set Γ), is a correct form of the secret key of the proposed CP-FE scheme, and is expressed by Eq. (9). Similarly, a *normal* ciphertext $\vec{\boldsymbol{c}}_\mathbb{S}^{\,\mathsf{norm}} := (\boldsymbol{c}_0, \ldots, \boldsymbol{c}_\ell)$ (with access structure $\mathbb{S}$) is Eq. (10). A *semi-functional* secret key, $\vec{\boldsymbol{k}}_\Gamma^{*\,\mathsf{semi}}$, is Eq. (16), and a *semi-functional* ciphertext, $\vec{\boldsymbol{c}}_\mathbb{S}^{\,\mathsf{semi}}$, is Eqs. (11)–(13). A *pre-semi-functional* secret key, $\vec{\boldsymbol{k}}_\Gamma^{*\,\mathsf{pre-semi}}$, and *pre-semi-functional* ciphertext, $\vec{\boldsymbol{c}}_\mathbb{S}^{\,\mathsf{pre-semi}}$, are Eq. (14) and Eqs. (11), (15) and (13), respectively.

Proof of Theorem 2. To prove Theorem 2, we consider the following $(2\nu_1 + \nu_2 + 3)$ 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 which were changed in a game from the previous game.

Game 0 : Original game. That is, the reply to a KeyGen query for $\Gamma := \{(t, \vec{x}_t)\}$ are:

$$\left.\begin{aligned} \boldsymbol{k}_0^* &:= (\delta,\ \boxed{0},\ 1,\ \varphi_0,\ 0)_{\mathbb{B}_0^*}, \\ \boldsymbol{k}_t^* &:= (\delta\vec{x}_t,\ \boxed{0^{n_t}},\ \vec{\varphi}_t,\ 0)_{\mathbb{B}_t} \ \text{ for } (t, \vec{x}_t) \in \Gamma, \end{aligned}\right\} \tag{9}$$

where $\delta \xleftarrow{\cup} \mathbb{F}_q^\times$, $\varphi_0 \xleftarrow{\cup} \mathbb{F}_q$, $\vec{\varphi}_t \xleftarrow{\cup} \mathbb{F}_q^{n_t}$ for $(t, \vec{x}_t) \in \Gamma$. The challenge ciphertext for challenge plaintexts $(m^{(0)}, m^{(1)})$ and access structure $\mathbb{S} := (M, \rho)$ is:

$$\left.\begin{array}{l} \boldsymbol{c}_0 := (-s_0,\ \boxed{0},\ \boxed{\zeta},\ 0,\ \eta_0)_{\mathbb{B}_0}, \\ \text{for } i = 1, \ldots, \ell, \\ \quad \text{if } \rho(i) = (t, \vec{v}_i), \quad \boldsymbol{c}_i := (s_i \vec{e}_{t,1} + \theta_i \vec{v}_i,\ \boxed{0^{n_t}},\ 0^{n_t},\ \eta_i)_{\mathbb{B}_t}, \\ \quad \text{if } \rho(i) = \neg(t, \vec{v}_i), \quad \boldsymbol{c}_i := (s_i \vec{v}_i,\ \boxed{0^{n_t}},\ 0^{n_t},\ \eta_i)_{\mathbb{B}_t}, \\ c_{d+1} := g_T^{\zeta} m^{(b)}, \end{array}\right\} \tag{10}$$

where $\vec{f} \xleftarrow{\mathsf{R}} \mathbb{F}_q^r$, $\vec{s}^{\mathrm{T}} := (s_1, \ldots, s_\ell)^{\mathrm{T}} := M \cdot \vec{f}^{\mathrm{T}}$, $s_0 := \vec{1} \cdot \vec{f}^{\mathrm{T}}$, $\eta_0, \theta_i \xleftarrow{\mathsf{U}} \mathbb{F}_q$ $\vec{\eta}_i \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_t}$ for $i = 1, \ldots, \ell$, and $\vec{e}_{t,1} := (1, 0, \ldots, 0) \in \mathbb{F}_q^{n_t}$.

Game 1 : Same as Game 0 except that the challenge ciphertext $(\boldsymbol{c}_0, \ldots, \boldsymbol{c}_\ell, c_{d+1})$ is:

$$\boldsymbol{c}_0 := (-s_0,\ \boxed{w_0},\ \zeta,\ 0,\ \eta_0)_{\mathbb{B}_0}, \tag{11}$$

$$\left.\begin{array}{l} \text{for } i = 1, \ldots, \ell, \\ \quad \text{if } \rho(i) = (t, \vec{v}_i), \quad \boldsymbol{c}_i := (s_i \vec{e}_{t,1} + \theta_i \vec{v}_i,\ \boxed{\vec{w}_i},\ 0^{n_t},\ \eta_i)_{\mathbb{B}_t}, \\ \quad \text{if } \rho(i) = \neg(t, \vec{v}_i), \quad \boldsymbol{c}_i := (s_i \vec{v}_i,\ \boxed{\vec{\overline{w}}_i},\ 0^{n_t},\ \eta_i)_{\mathbb{B}_t}, \end{array}\right\} \tag{12}$$

$$c_{d+1} := g_T^{\zeta} m^{(b)}, \tag{13}$$

where $w_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $\vec{w}_i, \vec{\overline{w}}_i \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_t}$ for $i = 1, \ldots, \ell$, and all the other variables are generated as in Game 0.

Game 2-h^+ ($\boldsymbol{h} = \boldsymbol{0}, \ldots, \boldsymbol{v} - \boldsymbol{1}$) : Game 2-0 is Game 1. Game 2-h^+ is the same as Game 2-h except that $\boldsymbol{k}_t^*$ for $t = 0$ and $(t, \vec{x}_t) \in \Gamma$ of the reply to the $(h+1)$th KeyGen query, and $(\boldsymbol{c}_1, \ldots, \boldsymbol{c}_\ell)$ of the challenge ciphertext are:

$$\left.\begin{array}{l} \boldsymbol{k}_0^* := (\delta,\ \boxed{r_0},\ 1,\ \varphi_0,\ 0)_{\mathbb{B}_0^*}, \\ \boldsymbol{k}_t^* := (\delta \vec{x}_t,\ \boxed{\vec{x}_t \cdot U_t},\ \vec{\varphi}_t,\ 0)_{\mathbb{B}_t} \ \text{ for } (t, \vec{x}_t) \in \Gamma, \end{array}\right\} \tag{14}$$

$$\left.\begin{array}{l} \text{for } i = 1, \ldots, \ell, \\ \quad \text{if } \rho(i) = (t, \vec{v}_i), \quad \boldsymbol{c}_i := (s_i \vec{e}_{t,1} + \theta_i \vec{v}_i,\ \boxed{(a_i \vec{e}_{t,1} + \pi_i \vec{v}_i) \cdot Z_t},\ 0^{n_t},\ \eta_i)_{\mathbb{B}_t}, \\ \quad \text{if } \rho(i) = \neg(t, \vec{v}_i), \quad \boldsymbol{c}_i := (s_i \vec{v}_i,\ \boxed{a_i \vec{v}_i \cdot Z_t},\ 0^{n_t},\ \eta_i)_{\mathbb{B}_t}, \end{array}\right\} \tag{15}$$

where $r_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $\vec{g} \xleftarrow{\mathsf{U}} \mathbb{F}_q^r$, $\vec{a}^{\mathrm{T}} := (a_1, \ldots, a_\ell)^{\mathrm{T}} := M \cdot \vec{g}^{\mathrm{T}}$, $\pi_i \xleftarrow{\mathsf{U}} \mathbb{F}_q$ for $i = 1, \ldots, \ell$, $Z_t \xleftarrow{\mathsf{U}} GL(n_t, \mathbb{F}_q)$, $U_t := (Z_t^{-1})^{\mathrm{T}}$ for $t = 1, \ldots, d$, and all the other variables are generated as in Game 2-h.

Game 2-($\boldsymbol{h+1}$) ($\boldsymbol{h} = \boldsymbol{0}, \ldots, \boldsymbol{v} - \boldsymbol{1}$): Game 2-$(h+1)$ is the same as Game 2-h^+ except that $\boldsymbol{k}_t^*$ for $(t, \vec{x}_t) \in \Gamma$ of the reply to the $(h+1)$th KeyGen query, and $(\boldsymbol{c}_1, \ldots, \boldsymbol{c}_\ell)$ of the challenge ciphertext are:

$$\left.\begin{array}{l} \boldsymbol{k}_0^* := (\delta,\ r_0,\ 1,\ \varphi_0,\ 0)_{\mathbb{B}_0^*}, \\ \boldsymbol{k}_t^* := (\delta \vec{x}_t,\ \boxed{0^{n_t}},\ \vec{\varphi}_t,\ 0)_{\mathbb{B}_t} \ \text{ for } (t, \vec{x}_t) \in \Gamma, \end{array}\right\}$$

$$\begin{aligned}&\text{for } i = 1, \dots, \ell,\\ &\quad \text{if } \rho(i) = (t, \vec{v}_i), \quad \boldsymbol{c}_i := (s_i \vec{e}_{t,1} + \theta_i \vec{v}_i, \boxed{\vec{w}_i}, 0^{n_t}, \eta_i)_{\mathbb{B}_t},\\ &\quad \text{if } \rho(i) = \neg(t, \vec{v}_i), \quad \boldsymbol{c}_i := (s_i \vec{v}_i, \boxed{\vec{\overline{w}}_i}, 0^{n_t}, \eta_i)_{\mathbb{B}_t},\end{aligned} \tag{16}$$

where $\vec{w}_i, \vec{\overline{w}}_i \xleftarrow{\cup} \mathbb{F}_q^{n_t}$ for $i = 1, \dots, \ell$, and all the other variables are generated as in Game 2-h^+.

Game 3 : Same as Game 2-ν except that $\boldsymbol{c}_0$ and c_{d+1} of the challenge ciphertext are

$$\boldsymbol{c}_0 := (-s_0, w_0, \boxed{\zeta'}, 0, \eta_0)_{\mathbb{B}_0}, \quad c_{d+1} := g_T^{\zeta} m^{(b)},$$

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

Let $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$ be $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{CP\text{-}FE,\,PH}}(\lambda)$ in Game 0, and $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2-h^+)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2-h)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)$ be the advantage of $\mathcal{A}$ in Game $1, 2-h, 2-h^+, 3$, respectively. It is clear that $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda) = 0$ by Lemma 13.

We will show four lemmas (Lemmas 9–12) that evaluate the gaps between pairs of $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2-h)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2-h^+)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2-(h+1))}(\lambda)$ for $h = 0, \dots, \nu-1$. From these lemmas and Lemmas 1 and 2 , we obtain

$$\begin{aligned}\mathsf{Adv}_{\mathcal{A}}^{\mathsf{CP\text{-}FE,\,PH}}(\lambda) = \mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda) &\le \left|\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)\right| + \sum_{h=0}^{\nu-1}\left|\mathsf{Adv}_{\mathcal{A}}^{(2-h)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(2-h^+)}(\lambda)\right|\\ &+ \sum_{h=0}^{\nu-1}\left|\mathsf{Adv}_{\mathcal{A}}^{(2-h^+)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(2-(h+1))}(\lambda)\right| + \left|\mathsf{Adv}_{\mathcal{A}}^{(2-\nu)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)\right| + \mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)\\ &\le \mathsf{Adv}_{\mathcal{B}_1}^{\mathsf{P1}}(\lambda) + \sum_{h=0}^{\nu-1}\mathsf{Adv}_{\mathcal{B}_{2,h}^+}^{\mathsf{P2}}(\lambda) + \sum_{h=0}^{\nu-1}\mathsf{Adv}_{\mathcal{B}_{2,h+1}}^{\mathsf{P2}}(\lambda) + (2d\nu + 6\nu + d + 2)/q\\ &\le \mathsf{Adv}_{\mathcal{E}_1}^{\mathsf{DLIN}}(\lambda) + \sum_{h=0}^{\nu-1}\left(\mathsf{Adv}_{\mathcal{E}_{2,h}^+}^{\mathsf{DLIN}}(\lambda) + \mathsf{Adv}_{\mathcal{E}_{2,h+1}}^{\mathsf{DLIN}}(\lambda)\right) + (2d\nu + 16\nu + d + 10)/q.\end{aligned}$$

This completes the proof of Theorem 2. □

Lemma 9. *For any adversary $\mathcal{A}$, 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, $|\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)| \le \mathsf{Adv}_{\mathcal{B}_1}^{\mathsf{P1}}(\lambda) + (d+1)/q$.*

Proof. In order to prove Lemma 9, we construct a probabilistic machine $\mathcal{B}_1$ against Problem 1 using any adversary $\mathcal{A}$ in a security game (Game 0 or 1) as a black box as follows:

1. $\mathcal{B}_1$ is given Problem 1 instance $(\mathsf{param}_{\vec{n}}, \mathbb{B}_0, \widehat{\mathbb{B}}_0^*, \boldsymbol{e}_{\beta,0}, \{\mathbb{B}_t, \widehat{\mathbb{B}}_t^*, \boldsymbol{e}_{\beta,t,1}, \boldsymbol{e}_{t,j}\}_{t=1,\dots,d;\, j=2,\dots,n_t})$.
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, $\mathcal{B}_1$ sets

$$\mathbb{D}_0 := \mathbb{B}_0,\ \mathbb{D}_0^* := \mathbb{B}_0^*,\ \widehat{\mathbb{D}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5}),\ \widehat{\mathbb{D}}_0^* := \widehat{\mathbb{B}}_0^*,$$
$$\mathbb{D}_t := (\boldsymbol{d}_{t,j})_{j=1,\ldots,3n_t+1} := (\boldsymbol{b}_{t,2}, \ldots, \boldsymbol{b}_{t,n_t}, \boldsymbol{b}_{t,1}, \boldsymbol{b}_{t,n_t+1}, \ldots, \boldsymbol{b}_{t,3n_t+1}),$$
$$\mathbb{D}_t^* := (\boldsymbol{d}_{t,j}^*)_{j=1,\ldots,3n_t+1} := (\boldsymbol{b}_{t,2}^*, \ldots, \boldsymbol{b}_{t,n_t}^*, \boldsymbol{b}_{t,1}^*, \boldsymbol{b}_{t,n_t+1}^*, \ldots, \boldsymbol{b}_{t,3n_t+1}^*),$$
$$\widehat{\mathbb{D}}_t := (\boldsymbol{d}_{t,1}, \ldots, \boldsymbol{d}_{t,n_t}, \boldsymbol{d}_{t,3n_t+1}),\quad \widehat{\mathbb{D}}_t^* := (\boldsymbol{d}_{t,1}^*, \ldots, \boldsymbol{d}_{t,n_t}^*, \boldsymbol{d}_{t,2n_t+1}^*, \ldots, \boldsymbol{d}_{t,3n_t}^*),$$

for $t = 1, \ldots, d$. $\mathcal{B}_1$ obtains $\widehat{\mathbb{D}}_t$ and $\widehat{\mathbb{D}}_t^*$ from $\mathbb{B}_t$ and $\widehat{\mathbb{B}}_t^*$ in the Problem 1 instance, and returns $\mathsf{pk} := (1^\lambda, \mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{D}}_t\}_{t=0,\ldots,d})$ to $\mathcal{A}$.
4. When a KeyGen query is issued for attribute sets Γ, $\mathcal{B}_1$ answers normal key sk_Γ computed using $\{\widehat{\mathbb{D}}_t^*\}_{t=0,\ldots,d}$.
5. When $\mathcal{B}_1$ receives an encryption query with challenge plaintexts $(m^{(0)}, m^{(1)})$ and $\mathbb{S} := (M, \rho)$ from $\mathcal{A}$, $\mathcal{B}_1$ calculates the challenge ciphertext $(\boldsymbol{c}_0, \ldots, \boldsymbol{c}_\ell, c_{d+1})$ as follows:

$$\boldsymbol{c}_0 := -s_0\boldsymbol{e}_{\beta,0} + \zeta\boldsymbol{b}_{0,3},\ \ \boldsymbol{c}_i := \textstyle\sum_{j=1}^{n_t-1} c_{i,j}\boldsymbol{e}_{t,j+1} + c_{i,n_t}\boldsymbol{e}_{\beta,t,1} \text{ for } i = 1, \ldots, \ell,\ \ c_{d+1} := g_T^{\zeta} m^{(b)},$$

where $b \xleftarrow{\mathsf{U}} \{0,1\}$, $\vec{f} \xleftarrow{\mathsf{R}} \mathbb{F}_q^r$, $\vec{s}^{\mathrm{T}} := (s_1, \ldots, s_\ell)^{\mathrm{T}} := M \cdot \vec{f}^{\mathrm{T}}$, $s_0 := \vec{1} \cdot \vec{f}^{\mathrm{T}}$, θ_i, $\zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$ for $i = 1, \ldots, \ell$, $\vec{c}_i := s_i\vec{e}_{t,1} + \theta_i\vec{v}_i$ if $\rho(i) = (t, \vec{v}_i)$ or $\vec{c}_i := s_i\vec{v}_i$ if $\rho(i) = (t, \vec{v}_i)$ for $i = 1, \ldots, \ell$, and $\boldsymbol{e}_{\beta,0}, \boldsymbol{b}_{0,3}, \boldsymbol{e}_{\beta,t,1}, \{\boldsymbol{e}_{t,j}\}_{j=2,\ldots,n_t}$ are from the Problem 1 instance. $\mathcal{B}_1$ gives the challenge ciphertext to $\mathcal{A}$.
6. When a KeyGen 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 $\beta' := 0$.

When $\beta = 0$, it is straightforward that the distribution by $\mathcal{B}_1$'s simulation is equivalent to that in Game 0. When $\beta = 1$, the distribution by $\mathcal{B}_1$'s simulation is equivalent to that in Game 1 except for the case that $s_0 = 0$ or there exists an $i \in \{1, \ldots, \ell\}$ such that $c_{i,n_t} = 0$, i.e., except with probability $(\ell + 1)/q \le (d + 1)/q$ since $\ell \le d$. □

Lemma 10. *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_2^+$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $|\mathsf{Adv}_{\mathcal{A}}^{(2-h)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(2-h^+)}(\lambda)| \le \mathsf{Adv}_{\mathcal{B}_{2,h}^+}^{P2}(\lambda) + (d+3)/q$, where $\mathcal{B}_{2,h}^+(\cdot) := \mathcal{B}_2^+(h, \cdot)$.*

Proof. In order to prove Lemma 10, we construct a probabilistic machine $\mathcal{B}_2^+$ against Problem 2 using an adversary $\mathcal{A}$ in a security game (Game 2-h or 2-h^+) as a black box as follows:

1. $\mathcal{B}_2^+$ is given an integer h and a Problem 2 instance, $(\mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}_t, \mathbb{B}_t^*\}_{t=0,\ldots,d}, \boldsymbol{h}_{\beta,0}^*, \boldsymbol{e}_0, \{\boldsymbol{h}_{\beta,t,j}^*, \boldsymbol{e}_{t,j}\}_{t=1,\ldots,d;\, j=1,\ldots,n_t})$.
2. $\mathcal{B}_2^+$ plays a role of the challenger in the security game against adversary $\mathcal{A}$.
3. At the first step of the game, $\mathcal{B}_2^+$ provides $\mathcal{A}$ a public key $\mathsf{pk} := (1^\lambda, \mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}'_t\}_{t=0,\ldots,d})$ of Game 2-h (and 2-h^+), where $\widehat{\mathbb{B}}'_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5})$ and

$\widehat{\mathbb{B}}'_t := (\boldsymbol{b}_{t,1}, .., \boldsymbol{b}_{t,n_t}, \boldsymbol{b}_{t,3n_t+1})$ for $t = 1, .., d$, that are obtained from the Problem 2 instance.

4. When the ιth key query is issued for attribute $\Gamma := \{(t, \vec{x}_t)\}$, $\mathcal{B}_2^+$ answers as follows:
   (a) When $1 \le \iota \le h$, $\mathcal{B}_2^+$ answers semi-functional key $(\boldsymbol{k}_0^*, \{\boldsymbol{k}_t^*\}_{(t,\vec{x}_t)\in\Gamma})$ with Eq. (16), that is computed using $\{\mathbb{B}_t^*\}_{t=0,..,d}$ of the Problem 2 instance.
   (b) When $\iota = h + 1$, $\mathcal{B}_2^+$ calculates $(\boldsymbol{k}_0^*, \{\boldsymbol{k}_t^*\}_{(t,\vec{x}_t)\in\Gamma})$ using $\boldsymbol{b}_{0,3}^*$, $\boldsymbol{h}_{\beta,0}^*, \{\boldsymbol{h}_{\beta,t,j}^*\}_{t=1,..,d;\, j=1,..,n_t}$ of the Problem 2 instance as follows:

$$\boldsymbol{k}_0^* := \boldsymbol{h}_{\beta,0}^* + \boldsymbol{b}_{0,3}^*, \quad \boldsymbol{k}_t^* := \textstyle\sum_{j=1}^{n_t} x_{t,j} \boldsymbol{h}_{\beta,t,j}^* \ \text{ for } (t, \vec{x}_t) \in \Gamma.$$

   (c) When $\iota \ge h + 2$, $\mathcal{B}_2^+$ answers normal key $(\boldsymbol{k}_0^*, \{\boldsymbol{k}_t^*\}_{(t,\vec{x}_t)\in\Gamma})$ with Eq. (9), that is computed using $\{\mathbb{B}_t^*\}_{t=0,..,d}$ of the Problem 2 instance.
5. When $\mathcal{B}_2^+$ receives an encryption query with challenge plaintexts $(m^{(0)}, m^{(1)})$ and $\mathbb{S} := (M, \rho)$ from $\mathcal{A}$, $\mathcal{B}_2^+$ computes challenge ciphertext $(\boldsymbol{c}_0, \ldots, \boldsymbol{c}_\ell, c_{d+1})$ as follows:

$$\begin{aligned}
&\pi'_t, \mu_t, g'_k, \widetilde{\mu}_k \xleftarrow{\cup} \mathbb{F}_q \ \text{ for } t = 1, \ldots, d;\ k = 1, \ldots, r,\\
&\widetilde{\boldsymbol{f}}_0 := \textstyle\sum_{k=1}^{r} \left(g'_k \boldsymbol{e}_0 + \widetilde{\mu}_k \boldsymbol{b}_{0,1}\right),\\
&\text{for } t = 1, \ldots, d;\ k = 1, \ldots, r;\ j = 1, \ldots, n_t;\\
&\quad \boldsymbol{f}_{t,j} := \pi'_t \boldsymbol{e}_{t,j} + \mu_t \boldsymbol{b}_{t,j}, \quad \widetilde{\boldsymbol{f}}_{t,k,j} := g'_k \boldsymbol{e}_{t,j} + \widetilde{\mu}_k \boldsymbol{b}_{t,j},\\
&\zeta \xleftarrow{\cup} \mathbb{F}_q, \quad \boldsymbol{c}_0 := -\widetilde{\boldsymbol{f}}_0 + \zeta \boldsymbol{b}_{0,3} + \boldsymbol{q}_0,\\
&\text{for } i = 1 \ldots, \ell,\\
&\quad \text{if } \rho(i) = (t, \vec{v}_i), \ \ \boldsymbol{c}_i := \textstyle\sum_{j=1}^{n_t} v_{i,j} \boldsymbol{f}_{t,j} + \sum_{k=1}^{r} M_{i,k} \widetilde{\boldsymbol{f}}_{t,k,1} + \boldsymbol{q}_i,\\
&\quad \text{if } \rho(i) = \neg(t, \vec{v}_i), \ \ \boldsymbol{c}_i := \textstyle\sum_{j=1}^{n_t} v_{i,j} (\sum_{k=1}^{r} M_{i,k} \widetilde{\boldsymbol{f}}_{t,k,j}) + \boldsymbol{q}_i,\\
&c_{d+1} := g_T^{\zeta} m^{(b)},
\end{aligned}$$

   where $(M_{i,k})_{i=1,\ldots,\ell;k=1,\ldots,r} := M$, $\boldsymbol{q}_0 \xleftarrow{\cup} \mathsf{span}\langle \boldsymbol{b}_{0,5} \rangle$, and $\boldsymbol{q}_i \xleftarrow{\cup} \mathsf{span}\langle \boldsymbol{b}_{t,3n_t+1} \rangle$ and $(\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{e}_0, \{\boldsymbol{e}_{t,j}\}_{t=1,\ldots,d;\, j=1,\ldots,n_t})$ is a part of the Problem 2 instance. $\mathcal{B}_2^+$ gives the challenge ciphertext to $\mathcal{A}$.
6. When a KeyGen query is issued by $\mathcal{A}$ after the encryption query, $\mathcal{B}_2^+$ executes the same procedure as that of step 4.
7. $\mathcal{A}$ finally outputs bit b'. If $b = b'$, $\mathcal{B}_2^+$ outputs $\beta' := 1$. Otherwise, $\mathcal{B}_2^+$ outputs $\beta' := 0$. □

Remark 4. $\widetilde{\boldsymbol{f}}_0, \boldsymbol{f}_{t,j}, \widetilde{\boldsymbol{f}}_{t,k,j}$ for $t = 1, \ldots, d; k = 1, \ldots, r; j = 1, \ldots, n_t$ calculated in the step 5 in the above simulation are expressed as:

$$\begin{aligned}
&\pi_t := \tau \pi'_t, \ \ \theta_t := \pi_t \omega + \mu_t, \ \ g_k := \tau g'_k, \ \ f_k := g_k \omega + \widetilde{\mu}_k,\\
&s_0 := \textstyle\sum_{k=1}^{r} f_k, \ \ a_0 := \sum_{k=1}^{r} g_k, \ \ w_0 := a_0 / u_0 \ (= a_0 z_0),\\
&\widetilde{\boldsymbol{f}}_0 = (s_0, w_0, 0, 0, 0)_{\mathbb{B}_0},
\end{aligned}$$

$$\begin{array}{rl} & \overbrace{\quad}^{n_t}\ \overbrace{\quad}^{n_t}\ \overbrace{\quad}^{n_t}\ \overbrace{\quad}^{1} \\ \boldsymbol{f}_{t,j} := & (\ \theta_t \vec{e}_{t,j},\ \pi_t \vec{z}_{t,j},\ 0^{n_t},\ 0\ )_{\mathbb{B}_t}, \\ \widetilde{\boldsymbol{f}}_{t,k,j} := & (\ f_k \vec{e}_{t,j},\ g_k \vec{z}_{t,j},\ 0^{n_t},\ 0\ )_{\mathbb{B}_t}, \end{array}$$

where $\tau, \omega, u_0, \{\vec{e}_{t,j}, \vec{z}_{t,j}\}_{t=1,\ldots,d;\, j=1,\ldots,n_t}$ are defined in Problem 2. Note that variables $\{\theta_t, \pi_t\}_{t=1,\ldots,d}$, $\{f_k, g_k\}_{k=1,\ldots,r}$ are independently and uniformly distributed. Therefore, $\{\boldsymbol{c}_i\}_{i=0,\ldots,\ell}$ are distributed as (11) and (15) except $w_0 := a_0/r_0$, i.e., $w_0 r_0 = a_0$, using a_0 and $r_0 := u_0 \xleftarrow{\cup} \mathbb{F}_q$ in $\boldsymbol{k}_0^*$ (Eq. 14).

Claim 2. *The distribution of the view of adversary $\mathcal{A}$ in the above-mentioned game simulated by $\mathcal{B}_2^+$ given a Problem 2 instance with $\beta \in \{0,1\}$ is the same as that in Game 2-h (resp. Game 2-h^+) if $\beta = 0$ (resp. $\beta = 1$) except with probability $(d+2)/q$ (resp. $1/q$).*

Proof. It is clear that $\mathcal{B}_2^+$'s simulation of the public key generation (step 3) and the ιth key query's answer for $\iota \neq h+1$ (cases (a) and (c) of step 4) is perfect, i.e., exactly the same as the Setup and the KeyGen oracle in Game 2-h and Game 2-h^+.

Therefore, to prove this lemma we will show that the joint distribution of the $(h+1)$th key query's answer and the challenge ciphertext by $\mathcal{B}_2^+$'s simulation given a Problem 2 instance with β is equivalent to that in Game 2-h (resp. Game 2-h^+), when $\beta = 0$ (resp. $\beta = 1$).

When $\beta = 0$, it is straightforward to show that they are equivalent except that δ defined in Problem 2 is zero or there exists $i \in \{0, \ldots, \ell\}$ such that $\vec{w}_i = \vec{0}$ with $\rho(i) = (t, \vec{v}_i)$ or $\overrightarrow{w}_i = \vec{0}$ with $\rho(i) = \neg(t, \vec{v}_i)$, where $\vec{w}_i$ and $\overrightarrow{w}_i$ are defined in Eqs. (11) and (12), i.e., except with probability $(\ell+2)/q \leq (d+2)/q$ since $\ell \leq d$.

When $\beta = 1$, the distribution by $\mathcal{B}_2^+$'s simulation is Eq. (14) for the key and Eqs. (11), (13), and (15) for the challenge ciphertext, where the distribution is the same as that defined in these equations except $w_0 := a_0/r_0$, i.e., $w_0 r_0 = a_0$, using $a_0 := \vec{1} \cdot \vec{g}^{\mathrm{T}}$ and $r_0 \xleftarrow{\cup} \mathbb{F}_q$ in $\boldsymbol{k}_0^*$ (Eq. 14) from Remark 4. The corresponding distribution in Game 2-h^+ is Eq. (14) and Eqs. (11), (13), and (15) where $r_0, w_0 \xleftarrow{\cup} \mathbb{F}_q$ as defined in the equations.

Moreover, similarly as in the proof of Claim 1, we can show that a_0 is uniformly and independently distributed from the other variables in the joint distribution of $\mathcal{B}_2^+$'s simulation.

Therefore, the view of adversary $\mathcal{A}$ in the game simulated by $\mathcal{B}_2^+$ given a Problem 2 instance with $\beta = 1$ is the same as that in Game 2-h^+ except that δ defined in Problem 2 is zero, i.e., except with probability $1/q$. □

This completes the proof of Lemma 10. □

Lemma 11. *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_2$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $|\mathsf{Adv}_{\mathcal{A}}^{(2-h^+)}(\lambda) - \mathsf{Adv}_{\mathcal{A}}^{(2-(h+1))}(\lambda)| \leq \mathsf{Adv}_{\mathcal{B}_{2,h+1}}^{P2}(\lambda) + (d+3)/q$, where $\mathcal{B}_{2,h+1}(\cdot) := \mathcal{B}_2(h, \cdot)$.*

Proof. The proof of Lemma 11 is similar to that of Lemma 6. □

Lemma 12. *For any adversary* $\mathcal{A}$, $\mathsf{Adv}^{(2-\nu)}_{\mathcal{A}}(\lambda) \leq \mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda) + 1/q$.

Proof. The proof of Lemma 12 is similar to that of Lemma 7. □

Lemma 13. *For any adversary* $\mathcal{A}$, $\mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda) = 0$.

6. UP-FE Scheme

This section presents a UP-FE scheme with the large class of relations, which is defined in Sect. 3.4.

6.1. *Construction*

In order to obtain a UP-FE scheme, we combine the KP-FE scheme in Sect. 4 and the CP-FE scheme in Sect. 5 using the first vector space $\mathbb{V}_0$ of dimension 8, instead of dimension 5. In the security proof, the semi-functional form of secret keys (resp. ciphertexts) has two-dimensional random component in $\mathsf{span}\langle \boldsymbol{b}^*_{0,3}, \boldsymbol{b}^*_{0,4}\rangle$ (resp. $\mathsf{span}\langle \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,4}\rangle$). For our KP-FE and CP-FE schemes, the corresponding random components are in one-dimensional subspace of $\mathbb{V}_0$ (see Sects. 4, 5).

$\widetilde{\rho} : \{1, \ldots, \ell\} \to \{1, \ldots, d\}$ is defined at the start of Sect. 4. In the proposed scheme, we assume that $\widetilde{\rho}$ is injective for $\mathbb{S} := (M, \rho)$, where $\mathbb{S} := \mathbb{S}^{\mathsf{KP}}, \mathbb{S}^{\mathsf{CP}}$.

In the description of the scheme, we assume that input vectors, $\vec{x}^{\mathsf{KP}}_t := (x^{\mathsf{KP}}_{t,1}, \ldots, x^{\mathsf{KP}}_{t,n_t})$ and $\vec{x}^{\mathsf{CP}}_t := (x^{\mathsf{CP}}_{t,1}, \ldots, x^{\mathsf{CP}}_{t,n_t})$, are normalized such that $x^{\mathsf{KP}}_{t,1} := 1$ and $x^{\mathsf{CP}}_{t,1} := 1$. (If $\vec{x}^{\mathsf{KP}}_t$ (resp. $\vec{x}^{\mathsf{CP}}_t$) is not normalized, change it to a normalized one by $(1/x^{\mathsf{KP}}_{t,1}) \cdot \vec{x}^{\mathsf{KP}}_t$ (resp. $(1/x^{\mathsf{CP}}_{t,1}) \cdot \vec{x}^{\mathsf{CP}}_t$), assuming that $x^{\mathsf{KP}}_{t,1}$ (resp. $x^{\mathsf{CP}}_{t,1}$) is nonzero). In addition, we assume that input vector $\vec{v}^{\mathsf{CP}}_t := (v^{\mathsf{CP}}_{i,1}, \ldots, v^{\mathsf{CP}}_{i,n_t})$ satisfies that $v^{\mathsf{CP}}_{i,n_t} \neq 0$.

For a format of attribute vectors $\vec{n} := ((d^{\mathsf{KP}}; n^{\mathsf{KP}}_1, \ldots, n^{\mathsf{KP}}_{d^{\mathsf{KP}}}), (d^{\mathsf{CP}}; n^{\mathsf{CP}}_1, \ldots, n^{\mathsf{CP}}_{d^{\mathsf{CP}}}))$ that indicates dimensions of vector spaces, $\vec{e}^{\mathsf{KP}}_{t,j}$ (resp. $\vec{e}^{\mathsf{CP}}_{t,j}$) denotes the canonical basis vector $(\overbrace{0 \cdots 0}^{j-1}, 1, \overbrace{0 \cdots 0}^{n^{\mathsf{KP}}_t - j}) \in \mathbb{F}_q^{n^{\mathsf{KP}}_t}$ for $j = 1, \ldots, n^{\mathsf{KP}}_t$ (resp. $(\overbrace{0 \cdots 0}^{j-1}, 1, \overbrace{0 \cdots 0}^{n^{\mathsf{CP}}_t - j}) \in \mathbb{F}_q^{n^{\mathsf{CP}}_t}$ for $j = 1, \ldots, n^{\mathsf{CP}}_t$).

We describe random dual orthonormal basis generator $\mathcal{G}^{\mathsf{UP}}_{\mathsf{ob}}$ below, which is used as a subroutine in the proposed UP-FE scheme. We refer to Sect. 1.5 for notations on DPVS, e.g., $(x_1, \ldots, x_N)_{\mathbb{B}}$, $(y_1, \ldots, y_N)_{\mathbb{B}^*}$ for $x_i, y_i \in \mathbb{F}_q$.

$$
\begin{aligned}
&\mathcal{G}^{\mathsf{UP}}_{\mathsf{ob}}(1^\lambda, \vec{n} := ((d^{\mathsf{KP}}; n^{\mathsf{KP}}_1, \ldots, n^{\mathsf{KP}}_{d^{\mathsf{KP}}}), (d^{\mathsf{CP}}; n^{\mathsf{CP}}_1, \ldots, n^{\mathsf{CP}}_{d^{\mathsf{CP}}})) : \\
&\quad \mathsf{param}_{\mathbb{G}} := (q, \mathbb{G}, \mathbb{G}_T, G, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{bpg}}(1^\lambda), \quad \psi \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times, \\
&\quad N_0 := 8, \quad N^{\mathsf{KP}}_t := 3n^{\mathsf{KP}}_t + 1 \ \text{ for } t = 1, \ldots, d^{\mathsf{KP}}, \quad N^{\mathsf{CP}}_t := 3n^{\mathsf{CP}}_t + 1 \ \text{ for } t = 1, \ldots, d^{\mathsf{CP}}, \\
&\quad \mathsf{param}_{\mathbb{V}_0} := (q, \mathbb{V}_0, \mathbb{G}_T, \mathbb{A}_0, e) := \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, N_0, \mathsf{param}_{\mathbb{G}}),
\end{aligned}
$$

$X_0 := (\chi_{0,i,j})_{i,j} \xleftarrow{\mathsf{U}} GL(N_0, \mathbb{F}_q),\ (\vartheta_{0,i,j})_{i,j} := \psi \cdot (X_0^{\mathrm{T}})^{-1},$

$\boldsymbol{b}_{0,i} := (\chi_{0,i,1}, \ldots, \chi_{0,i,N_0})_{\mathbb{A}_0},\ \mathbb{B}_0 := (\boldsymbol{b}_{0,1}, \ldots, \boldsymbol{b}_{0,N_0}),$

$\boldsymbol{b}^*_{0,i} := (\vartheta_{0,i,1}, \ldots, \vartheta_{0,i,N_0})_{\mathbb{A}_0},\ \mathbb{B}^*_0 := (\boldsymbol{b}^*_{0,1}, \ldots, \boldsymbol{b}^*_{0,N_0}),$

for $t = 1, \ldots, d^{\mathsf{KP}}$, $\mathsf{param}_{\mathbb{V}_t^{\mathsf{KP}}} := (q, \mathbb{V}_t^{\mathsf{KP}}, \mathbb{G}_T, \mathbb{A}_t^{\mathsf{KP}}, e) \quad := \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, N_t^{\mathsf{KP}}, \mathsf{param}_{\mathbb{G}}),$

$X_t^{\mathsf{KP}} := (\chi^{\mathsf{KP}}_{t,i,j})_{i,j} \xleftarrow{\mathsf{U}} GL(N_t^{\mathsf{KP}}, \mathbb{F}_q),\ (\vartheta^{\mathsf{KP}}_{t,i,j})_{i,j} := \psi \cdot ((X_t^{\mathsf{KP}})^{\mathrm{T}})^{-1},$

$\boldsymbol{b}^{\mathsf{KP}}_{t,i} := (\chi^{\mathsf{KP}}_{t,i,1}, \ldots, \chi^{\mathsf{KP}}_{t,i,N_t^{\mathsf{KP}}})_{\mathbb{A}_t^{\mathsf{KP}}},\ \mathbb{B}_t^{\mathsf{KP}} := (\boldsymbol{b}^{\mathsf{KP}}_{t,1}, \ldots, \boldsymbol{b}^{\mathsf{KP}}_{t,N_t^{\mathsf{KP}}}),$

$\boldsymbol{b}^{*\mathsf{KP}}_{t,i} := (\vartheta^{\mathsf{KP}}_{t,i,1}, \ldots, \vartheta^{\mathsf{KP}}_{t,i,N_t^{\mathsf{KP}}})_{\mathbb{A}_t^{\mathsf{KP}}},\ \mathbb{B}_t^{*\mathsf{KP}} := (\boldsymbol{b}^{*\mathsf{KP}}_{t,1}, \ldots, \boldsymbol{b}^{*\mathsf{KP}}_{t,N_t^{\mathsf{KP}}}),$

for $t = 1, \ldots, d^{\mathsf{CP}}$, $\mathsf{param}_{\mathbb{V}_t^{\mathsf{CP}}} := (q, \mathbb{V}_t^{\mathsf{CP}}, \mathbb{G}_T, \mathbb{A}_t^{\mathsf{CP}}, e) \quad := \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, N_t^{\mathsf{CP}}, \mathsf{param}_{\mathbb{G}}),$

$X_t^{\mathsf{CP}} := (\chi^{\mathsf{CP}}_{t,i,j})_{i,j} \xleftarrow{\mathsf{U}} GL(N_t^{\mathsf{CP}}, \mathbb{F}_q),\ (\vartheta^{\mathsf{CP}}_{t,i,j})_{i,j} := \psi \cdot ((X_t^{\mathsf{CP}})^{\mathrm{T}})^{-1},$

$\boldsymbol{b}^{\mathsf{CP}}_{t,i} := (\chi^{\mathsf{CP}}_{t,i,1}, \ldots, \chi^{\mathsf{CP}}_{t,i,N_t^{\mathsf{CP}}})_{\mathbb{A}_t^{\mathsf{CP}}},\ \mathbb{B}_t^{\mathsf{CP}} := (\boldsymbol{b}^{\mathsf{CP}}_{t,1}, \ldots, \boldsymbol{b}^{\mathsf{CP}}_{t,N_t^{\mathsf{CP}}}),$

$\boldsymbol{b}^{*\mathsf{CP}}_{t,i} := (\vartheta^{\mathsf{CP}}_{t,i,1}, \ldots, \vartheta^{\mathsf{CP}}_{t,i,N_t^{\mathsf{CP}}})_{\mathbb{A}_t^{\mathsf{CP}}},\ \mathbb{B}_t^{*\mathsf{CP}} := (\boldsymbol{b}^{*\mathsf{CP}}_{t,1}, \ldots, \boldsymbol{b}^{*\mathsf{CP}}_{t,N_t^{\mathsf{CP}}}),$

$g_T := e(G, G)^{\psi},\ \mathsf{param}_{\vec{n}} := (\mathsf{param}_{\mathbb{V}_0}, \{\mathsf{param}_{\mathbb{V}_t^{\mathsf{KP}}}\}_{t=1,\ldots,d^{\mathsf{KP}}}, \quad \{\mathsf{param}_{\mathbb{V}_t^{\mathsf{CP}}}\}_{t=1,\ldots,d^{\mathsf{CP}}},\ g_T),$

return $(\mathsf{param}_{\vec{n}}, \{\mathbb{B}_0, \mathbb{B}^*_0\}, \{\mathbb{B}_t^{\mathsf{KP}}, \mathbb{B}_t^{*\mathsf{KP}}\}_{t=1,\ldots,d^{\mathsf{KP}}}, \{\mathbb{B}_t^{\mathsf{CP}}, \mathbb{B}_t^{*\mathsf{CP}}\}_{t=1,\ldots,d^{\mathsf{CP}}})$.

The proposed UP-FE scheme is given as:

$\mathsf{Setup}(1^\lambda,\ \vec{n} := ((d^{\mathsf{KP}}; n_1^{\mathsf{KP}}, \ldots, n_{d^{\mathsf{KP}}}^{\mathsf{KP}}), (d^{\mathsf{CP}}; n_1^{\mathsf{CP}}, \ldots, n_{d^{\mathsf{CP}}}^{\mathsf{CP}}))) :$

$(\mathsf{param}_{\vec{n}}, \mathbb{B}_0, \mathbb{B}^*_0, \{\mathbb{B}_t^{\mathsf{KP}}, \mathbb{B}_t^{*\mathsf{KP}}\}_{t=1,\ldots,d^{\mathsf{KP}}}, \{\mathbb{B}_t^{\mathsf{CP}}, \mathbb{B}_t^{*\mathsf{CP}}\}_{t=1,\ldots,d^{\mathsf{CP}}}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}^{\mathsf{UP}}(1^\lambda, \vec{n}),$

$\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,2}, \boldsymbol{b}_{0,5}, \boldsymbol{b}_{0,8}),\ \widehat{\mathbb{B}}^*_0 := (\boldsymbol{b}^*_{0,1}, \boldsymbol{b}^*_{0,2}, \boldsymbol{b}^*_{0,5}, \boldsymbol{b}^*_{0,6}, \boldsymbol{b}^*_{0,7}),$

for $t = 1, .., d^{\mathsf{KP}}$, $\widehat{\mathbb{B}}_t^{\mathsf{KP}} := (\boldsymbol{b}^{\mathsf{KP}}_{t,1}, .., \boldsymbol{b}^{\mathsf{KP}}_{t,n_t^{\mathsf{KP}}}, \boldsymbol{b}^{\mathsf{KP}}_{t,3n_t^{\mathsf{KP}}+1}),$

$\widehat{\mathbb{B}}_t^{*\mathsf{KP}} := (\boldsymbol{b}^{*\mathsf{KP}}_{t,1}, .., \boldsymbol{b}^{*\mathsf{KP}}_{t,n_t^{\mathsf{KP}}}, \boldsymbol{b}^{*\mathsf{KP}}_{t,2n_t^{\mathsf{KP}}+1}, .., \boldsymbol{b}^{*\mathsf{KP}}_{t,3n_t^{\mathsf{KP}}}),$

for $t = 1, .., d^{\mathsf{CP}}$, $\widehat{\mathbb{B}}_t^{\mathsf{CP}} := (\boldsymbol{b}^{\mathsf{CP}}_{t,1}, .., \boldsymbol{b}^{\mathsf{CP}}_{t,n_t^{\mathsf{CP}}}, \boldsymbol{b}^{\mathsf{CP}}_{t,3n_t^{\mathsf{CP}}+1}),$

$\widehat{\mathbb{B}}_t^{*\mathsf{CP}} := (\boldsymbol{b}^{*\mathsf{CP}}_{t,1}, .., \boldsymbol{b}^{*\mathsf{CP}}_{t,n_t^{\mathsf{CP}}}, \boldsymbol{b}^{*\mathsf{CP}}_{t,2n_t^{\mathsf{CP}}+1}, .., \boldsymbol{b}^{*\mathsf{CP}}_{t,3n_t^{\mathsf{CP}}}),$

$\mathsf{pk} := (1^\lambda, \mathsf{param}_{\vec{n}}, \widehat{\mathbb{B}}_0, \{\widehat{\mathbb{B}}_t^{\mathsf{KP}}\}_{t=1,\ldots,d^{\mathsf{KP}}}, \{\widehat{\mathbb{B}}_t^{\mathsf{CP}}\}_{t=1,\ldots,d^{\mathsf{CP}}}),$

$\mathsf{sk} := (\widehat{\mathbb{B}}^*_0, \{\widehat{\mathbb{B}}_t^{*\mathsf{KP}}\}_{t=1,\ldots,d^{\mathsf{KP}}}, \{\widehat{\mathbb{B}}_t^{*\mathsf{CP}}\}_{t=1,\ldots,d^{\mathsf{CP}}}),$

return pk, sk.

$\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \mathbb{S}^{\mathsf{KP}} := (M^{\mathsf{KP}}, \rho^{\mathsf{KP}}),$

$\Gamma^{\mathsf{CP}} := \{(t, \vec{x}_t^{\mathsf{CP}} := (x^{\mathsf{CP}}_{t,1}, \ldots, x^{\mathsf{CP}}_{t,n_t^{\mathsf{CP}}}) \in \mathbb{F}_q^{n_t^{\mathsf{CP}}} \setminus \{\vec{0}\}) \mid 1 \le t \le d^{\mathsf{CP}},\ x^{\mathsf{CP}}_{t,1} := 1\})$

$\vec{f}^{\,\mathsf{KP}} \xleftarrow{\mathsf{U}} \mathbb{F}_q^{\ell^{\mathsf{KP}}},\ (\vec{s}^{\,\mathsf{KP}})^{\mathrm{T}} := (s_1^{\mathsf{KP}}, \ldots, s_{\ell^{\mathsf{KP}}}^{\mathsf{KP}})^{\mathrm{T}} := M^{\mathsf{KP}} \cdot (\vec{f}^{\,\mathsf{KP}})^{\mathrm{T}},\ s_0^{\mathsf{KP}} := \vec{1} \cdot (\vec{f}^{\,\mathsf{KP}})^{\mathrm{T}},$

$\delta^{\mathsf{CP}} \xleftarrow{\mathsf{U}} \mathbb{F}_q,\ \vec{\eta}_t^{\,\mathsf{CP}} \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_t^{\mathsf{CP}}}$ such that $(t, \vec{x}_t^{\,\mathsf{CP}}) \in \Gamma^{\mathsf{CP}},\ (\eta_{0,1}, \eta_{0,2}) \xleftarrow{\mathsf{U}} \mathbb{F}_q^2,$

$\boldsymbol{k}^*_0 := (-s_0^{\mathsf{KP}}, \delta^{\mathsf{CP}}, 0, 0, 1, \eta_{0,1}, \eta_{0,2}, 0)_{\mathbb{B}^*_0},$

for $i = 1, \ldots, \ell^{\mathsf{KP}}$,

if $\rho^{\mathsf{KP}}(i) = (t, \vec{v}_i^{\,\mathsf{KP}} := (v^{\mathsf{KP}}_{i,1}, \ldots, v^{\mathsf{KP}}_{i,n_t^{\mathsf{KP}}}) \in \mathbb{F}_q^{n_t^{\mathsf{KP}}} \setminus \{\vec{0}\}), \quad \theta_i^{\mathsf{KP}} \xleftarrow{\mathsf{U}} \mathbb{F}_q, \vec{\eta}_i^{\,\mathsf{KP}} \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_t^{\mathsf{KP}}},$

$$\boldsymbol{k}_i^{*\mathsf{KP}} := (\overbrace{\quad s_i^{\mathsf{KP}}\vec{e}_{t,1}^{\,\mathsf{KP}} + \theta_i^{\mathsf{KP}}\vec{v}_i^{\,\mathsf{KP}}\quad}^{n_t^{\mathsf{KP}}},\ \overbrace{0^{n_t^{\mathsf{KP}}}}^{n_t^{\mathsf{KP}}},\ \overbrace{\vec{\eta}_i^{\,\mathsf{KP}}}^{n_t^{\mathsf{KP}}},\ \overbrace{0}^{1})_{\mathbb{B}_t^{*\mathsf{KP}}},$$

if $\rho^{\mathsf{KP}}(i) = \neg(t, \vec{v}_i^{\,\mathsf{KP}})$, $\vec{\eta}_i^{\,\mathsf{KP}} \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_t^{\mathsf{KP}}}$,

$$\boldsymbol{k}_i^{*\mathsf{KP}} := (\overbrace{\quad s_i^{\mathsf{KP}}\vec{v}_i^{\,\mathsf{KP}}\quad}^{n_t^{\mathsf{KP}}},\ \overbrace{0^{n_t^{\mathsf{KP}}}}^{n_t^{\mathsf{KP}}},\ \overbrace{\vec{\eta}_i^{\,\mathsf{KP}}}^{n_t^{\mathsf{KP}}},\ \overbrace{0}^{1})_{\mathbb{B}_t^{*\mathsf{KP}}},$$

$$\boldsymbol{k}_t^{*\mathsf{CP}} := (\overbrace{\quad \delta^{\mathsf{CP}}\vec{x}_t^{\,\mathsf{CP}}\quad}^{n_t^{\mathsf{CP}}},\ \overbrace{0^{n_t^{\mathsf{CP}}}}^{n_t^{\mathsf{CP}}},\ \overbrace{\vec{\eta}_t^{\,\mathsf{CP}}}^{n_t^{\mathsf{CP}}},\ \overbrace{0}^{1})_{\mathbb{B}_t^{*\mathsf{CP}}} \text{ for } (t, \vec{x}_t^{\,\mathsf{CP}}) \in \Gamma^{\mathsf{CP}},$$

return $\mathsf{sk}_{(\mathbb{S}^{\mathsf{KP}}, \Gamma^{\mathsf{CP}})} := (\boldsymbol{k}_0^*;\ \mathbb{S}^{\mathsf{KP}}, \boldsymbol{k}_1^{*\mathsf{KP}}, \ldots, \boldsymbol{k}_{\ell^{\mathsf{KP}}}^{*\mathsf{KP}};\ \Gamma^{\mathsf{CP}}, \{\boldsymbol{k}_t^{*\mathsf{CP}}\}_{(t,\vec{x}_t^{\,\mathsf{CP}})\in\Gamma^{\mathsf{CP}}})$.

$\mathsf{Enc}(\mathsf{pk},\ m,\ \Gamma^{\mathsf{KP}} := \{(t, \vec{x}_t^{\,\mathsf{KP}} := (x_{t,1}^{\mathsf{KP}}, .., x_{t,n_t^{\mathsf{KP}}}^{\mathsf{KP}}) \in \mathbb{F}_q^{n_t^{\mathsf{KP}}} \setminus \{\vec{0}\}) \mid 1 \le t \le d^{\mathsf{KP}}, x_{t,1}^{\mathsf{KP}} := 1\},$

$\mathbb{S}^{\mathsf{CP}} := (M^{\mathsf{CP}}, \rho^{\mathsf{CP}})) :$

$\omega^{\mathsf{KP}}, \varphi_0, \varphi_t^{\mathsf{KP}}, \zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$ for $(t, \vec{x}_t^{\,\mathsf{KP}}) \in \Gamma^{\mathsf{KP}}$,

$\vec{f}^{\,\mathsf{CP}} \xleftarrow{\mathsf{R}} \mathbb{F}_q^{r^{\mathsf{CP}}}$, $(\vec{s}^{\,\mathsf{CP}})^{\mathrm{T}} := (s_1^{\mathsf{CP}}, \ldots, s_{\ell^{\mathsf{CP}}}^{\mathsf{CP}})^{\mathrm{T}} := M^{\mathsf{CP}} \cdot (\vec{f}^{\,\mathsf{CP}})^{\mathrm{T}}$, $s_0^{\mathsf{CP}} := \vec{1} \cdot (\vec{f}^{\,\mathsf{CP}})^{\mathrm{T}}$,

$\boldsymbol{c}_0 := (\omega^{\mathsf{KP}}, -s_0^{\mathsf{CP}}, 0, 0, \zeta, 0, 0, \varphi_0)_{\mathbb{B}_0}$,

$$\boldsymbol{c}_t^{\mathsf{KP}} := (\overbrace{\quad \omega^{\mathsf{KP}}\vec{x}_t^{\,\mathsf{KP}}\quad}^{n_t^{\mathsf{KP}}},\ \overbrace{0^{n_t^{\mathsf{KP}}}}^{n_t^{\mathsf{KP}}},\ \overbrace{0^{n_t^{\mathsf{KP}}}}^{n_t^{\mathsf{KP}}},\ \overbrace{\varphi_t^{\mathsf{KP}}}^{1})_{\mathbb{B}_t^{\mathsf{KP}}} \quad \text{for } (t, \vec{x}_t^{\,\mathsf{KP}}) \in \Gamma^{\mathsf{KP}},$$

for $i = 1, \ldots, \ell^{\mathsf{CP}}$,

if $\rho^{\mathsf{CP}}(i) = (t, \vec{v}_i^{\,\mathsf{CP}} := (v_{i,1}^{\mathsf{CP}}, \ldots, v_{i,n_t^{\mathsf{CP}}}^{\mathsf{CP}}) \in \mathbb{F}_q^{n_t^{\mathsf{CP}}} \setminus \{\vec{0}\})$ $(v_{i,n_t^{\mathsf{CP}}}^{\mathsf{CP}} := 1)$, $\varphi_i^{\mathsf{CP}}, \theta_i^{\mathsf{CP}} \xleftarrow{\mathsf{U}} \mathbb{F}_q$,

$$\boldsymbol{c}_i^{\mathsf{CP}} := (\overbrace{\quad s_i^{\mathsf{CP}}\vec{e}_{t,1}^{\,\mathsf{CP}} + \theta_i^{\mathsf{CP}}\vec{v}_i^{\,\mathsf{CP}}\quad}^{n_t^{\mathsf{CP}}},\ \overbrace{0^{n_t^{\mathsf{CP}}}}^{n_t^{\mathsf{CP}}},\ \overbrace{0^{n_t^{\mathsf{CP}}}}^{n_t^{\mathsf{CP}}},\ \overbrace{\varphi_i^{\mathsf{CP}}}^{1})_{\mathbb{B}_i^{\mathsf{CP}}},$$

if $\rho^{\mathsf{CP}}(i) = \neg(t, \vec{v}_i^{\,\mathsf{CP}})$, $\varphi_i^{\mathsf{CP}} \xleftarrow{\mathsf{U}} \mathbb{F}_q$,

$$\boldsymbol{c}_i^{\mathsf{CP}} := (\overbrace{\quad s_i^{\mathsf{CP}}\vec{v}_i^{\,\mathsf{CP}}\quad}^{n_t^{\mathsf{CP}}},\ \overbrace{0^{n_t^{\mathsf{CP}}}}^{n_t^{\mathsf{CP}}},\ \overbrace{0^{n_t^{\mathsf{CP}}}}^{n_t^{\mathsf{CP}}},\ \overbrace{\varphi_i^{\mathsf{CP}}}^{1})_{\mathbb{B}_i^{\mathsf{CP}}},$$

$c_{d+1} := g_T^{\zeta} m$,

return $\mathsf{ct}_{(\Gamma^{\mathsf{KP}}, \mathbb{S}^{\mathsf{CP}})} := (\boldsymbol{c}_0;\ \Gamma^{\mathsf{KP}}, \{\boldsymbol{c}_t^{\mathsf{KP}}\}_{(t,\vec{x}_t^{\,\mathsf{KP}})\in\Gamma^{\mathsf{KP}}};\ \mathbb{S}^{\mathsf{CP}}, \boldsymbol{c}_1^{\mathsf{CP}}, \ldots, \boldsymbol{c}_{\ell^{\mathsf{CP}}}^{\mathsf{CP}};\ c_{d+1})$.

$\mathsf{Dec}(\mathsf{pk},\ \mathsf{sk}_{(\mathbb{S}^{\mathsf{KP}}, \Gamma^{\mathsf{CP}})} := (\boldsymbol{k}_0^*;\ \mathbb{S}^{\mathsf{KP}}, \boldsymbol{k}_1^{*\mathsf{KP}}, \ldots, \boldsymbol{k}_{\ell^{\mathsf{KP}}}^{*\mathsf{KP}};\ \Gamma^{\mathsf{CP}}, \{\boldsymbol{k}_t^{*\mathsf{CP}}\}_{(t,\vec{x}_t^{\,\mathsf{CP}})\in\Gamma^{\mathsf{CP}}}),$

$\mathsf{ct}_{(\Gamma^{\mathsf{KP}}, \mathbb{S}^{\mathsf{CP}})} := (\boldsymbol{c}_0;\ \Gamma^{\mathsf{KP}}, \{\boldsymbol{c}_t^{\mathsf{KP}}\}_{(t,\vec{x}_t^{\,\mathsf{KP}})\in\Gamma^{\mathsf{KP}}};\ \mathbb{S}^{\mathsf{CP}}, \boldsymbol{c}_1^{\mathsf{CP}}, \ldots, \boldsymbol{c}_{\ell^{\mathsf{CP}}}^{\mathsf{CP}};\ c_{d+1})) :$

If $\mathbb{S}^{\mathsf{KP}} := (M^{\mathsf{KP}}, \rho^{\mathsf{KP}})$ accepts $\Gamma^{\mathsf{KP}} := \{(t, \vec{x}_t^{\,\mathsf{KP}})\}$

and $\mathbb{S}^{\mathsf{CP}} := (M^{\mathsf{CP}}, \rho^{\mathsf{CP}})$ accepts $\Gamma^{\mathsf{CP}} := \{(t, \vec{x}_t^{\,\mathsf{CP}})\}$,

then compute $(I^{\mathsf{KP}},\ \{\alpha_i^{\mathsf{KP}}\}_{i\in I^{\mathsf{KP}}})$ and $(I^{\mathsf{CP}},\ \{\alpha_i^{\mathsf{CP}}\}_{i\in I^{\mathsf{CP}}})$ such that

$\vec{1} = \sum_{i\in I^{\mathsf{KP}}} \alpha_i^{\mathsf{KP}} M_i^{\mathsf{KP}}$, where M_i^{KP} is the ith row of M^{KP}, and

$I^{\mathsf{KP}} \subseteq \{i \in \{1, \ldots, \ell^{\mathsf{KP}}\} \mid [\rho^{\mathsf{KP}}(i) = (t, \vec{v}_i^{\,\mathsf{KP}}) \wedge (t, \vec{x}_t^{\,\mathsf{KP}}) \in \Gamma^{\mathsf{KP}} \wedge \vec{v}_i^{\,\mathsf{KP}} \cdot \vec{x}_t^{\,\mathsf{KP}} = 0]$

$\vee\ [\rho^{\mathsf{KP}}(i) = \neg(t, \vec{v}_i^{\,\mathsf{KP}}) \wedge (t, \vec{x}_t^{\,\mathsf{KP}}) \in \Gamma^{\mathsf{KP}} \wedge \vec{v}_i^{\,\mathsf{KP}} \cdot \vec{x}_t^{\,\mathsf{KP}} \neq 0]\ \}$, and

$\vec{1} = \sum_{i\in I^{\mathsf{CP}}} \alpha_i^{\mathsf{CP}} M_i^{\mathsf{CP}}$, where M_i^{CP} is the ith row of M^{CP}, and

$I^{\mathsf{CP}} \subseteq \{i \in \{1, \ldots, \ell^{\mathsf{CP}}\} \mid [\rho^{\mathsf{CP}}(i) = (t, \vec{v}_i^{\,\mathsf{CP}}) \wedge (t, \vec{x}_t^{\,\mathsf{CP}}) \in \Gamma^{\mathsf{CP}} \wedge \vec{v}_i^{\,\mathsf{CP}} \cdot \vec{x}_t^{\,\mathsf{CP}} = 0]$

$$\vee\ [\rho^{\mathsf{CP}}(i) = \neg(t, \vec{v}_i^{\mathsf{CP}})\ \wedge\ (t, \vec{x}_t^{\mathsf{CP}})\ \in \Gamma^{\mathsf{CP}}\ \wedge\ \vec{v}_i^{\mathsf{CP}} \cdot \vec{x}_t^{\mathsf{CP}} \neq 0]\ \},$$

$$K := e(\boldsymbol{c}_0, \boldsymbol{k}_0^*) \cdot$$

$$\prod_{i \in I^{\mathsf{KP}}\ \wedge\ \rho^{\mathsf{KP}}(i) = (t, \vec{v}_i^{\mathsf{KP}})} e(\boldsymbol{c}_t^{\mathsf{KP}}, \boldsymbol{k}_i^{*\mathsf{KP}})^{\alpha_i^{\mathsf{KP}}} \prod_{i \in I^{\mathsf{KP}}\ \wedge\ \rho^{\mathsf{KP}}(i) = \neg(t, \vec{v}_i^{\mathsf{KP}})} e(\boldsymbol{c}_t^{\mathsf{KP}}, \boldsymbol{k}_i^{*\mathsf{KP}})^{\alpha_i^{\mathsf{KP}}/(\vec{v}_i^{\mathsf{KP}} \cdot \vec{x}_t^{\mathsf{KP}})} \cdot$$

$$\prod_{i \in I^{\mathsf{CP}}\ \wedge\ \rho^{\mathsf{CP}}(i) = (t, \vec{v}_i^{\mathsf{CP}})} e(\boldsymbol{c}_t^{\mathsf{CP}}, \boldsymbol{k}_i^{*\mathsf{CP}})^{\alpha_i^{\mathsf{CP}}} \prod_{i \in I^{\mathsf{CP}}\ \wedge\ \rho^{\mathsf{CP}}(i) = \neg(t, \vec{v}_i^{\mathsf{CP}})} e(\boldsymbol{c}_t^{\mathsf{CP}}, \boldsymbol{k}_i^{*\mathsf{CP}})^{\alpha_i^{\mathsf{CP}}/(\vec{v}_i^{\mathsf{CP}} \cdot \vec{x}_t^{\mathsf{CP}})},$$

return $m' := c_{d+1}/K$.

[Correctness] If $\mathbb{S}^{\mathsf{KP}} := (M^{\mathsf{KP}}, \rho^{\mathsf{KP}})$ accepts $\Gamma^{\mathsf{KP}} := \{(t, \vec{x}_t^{\mathsf{KP}})\}$ and $\mathbb{S}^{\mathsf{CP}} := (M^{\mathsf{CP}}, \rho^{\mathsf{CP}})$ accepts $\Gamma^{\mathsf{CP}} := \{(t, \vec{x}_t^{\mathsf{CP}})\}$,

$$e(\boldsymbol{c}_0, \boldsymbol{k}_0^*) \cdot$$

$$\prod_{i \in I^{\mathsf{KP}}\ \wedge\ \rho^{\mathsf{KP}}(i) = (t, \vec{v}_i^{\mathsf{KP}})} e(\boldsymbol{c}_t^{\mathsf{KP}}, \boldsymbol{k}_i^{*\mathsf{KP}})^{\alpha_i^{\mathsf{KP}}} \cdot \prod_{i \in I^{\mathsf{KP}}\ \wedge\ \rho^{\mathsf{KP}}(i) = \neg(t, \vec{v}_i^{\mathsf{KP}})} e(\boldsymbol{c}_t^{\mathsf{KP}}, \boldsymbol{k}_i^{*\mathsf{KP}})^{\alpha_i^{\mathsf{KP}}/(\vec{v}_i^{\mathsf{KP}} \cdot \vec{x}_t^{\mathsf{KP}})} \cdot$$

$$\prod_{i \in I^{\mathsf{CP}}\ \wedge\ \rho^{\mathsf{CP}}(i) = (t, \vec{v}_i^{\mathsf{CP}})} e(\boldsymbol{c}_t^{\mathsf{CP}}, \boldsymbol{k}_i^{*\mathsf{CP}})^{\alpha_i^{\mathsf{CP}}} \cdot \prod_{i \in I^{\mathsf{CP}}\ \wedge\ \rho^{\mathsf{CP}}(i) = \neg(t, \vec{v}_i^{\mathsf{CP}})} e(\boldsymbol{c}_t^{\mathsf{CP}}, \boldsymbol{k}_i^{*\mathsf{CP}})^{\alpha_i^{\mathsf{CP}}/(\vec{v}_i^{\mathsf{CP}} \cdot \vec{x}_t^{\mathsf{CP}})}$$

$$= g_T^{-(\omega^{\mathsf{KP}} s_0^{\mathsf{KP}} + \delta^{\mathsf{CP}} s_0^{\mathsf{CP}}) + \zeta} \cdot g_T^{\omega^{\mathsf{KP}} (\sum_{i \in I^{\mathsf{KP}}} \alpha_i^{\mathsf{KP}} s_i^{\mathsf{KP}})} \cdot g_T^{\delta^{\mathsf{CP}} (\sum_{i \in I^{\mathsf{CP}}} \alpha_i^{\mathsf{CP}} s_i^{\mathsf{CP}})} = g_T^{\zeta}.$$

6.2. *Security*

The following theorem can be proved similarly as Theorems 1 and 2 .

Theorem 3. *The proposed UP-FE scheme is adaptively payload-hiding against chosen-plaintext attacks under the DLIN assumption.*

7. CCA-Secure CP-FE Scheme

We can transform the proposed (KP, CP and UP)-FE schemes to CCA-secure (KP, CP and UP)-FE schemes, respectively, by using the Canetti–Halevi–Katz (CHK) transformation [17] or the Boneh–Katz (BK) transformation [13].

This section shows a CCA-secure CP-FE scheme, that is modified from the CP-FE scheme in Sect. 5 through the CHK transformation, in which a strongly unforgeable one-time signature scheme (Gen, Sig, Ver) is employed.

We can similarly apply the CHK transformation to our KP-FE scheme and the BK transformation to the FE schemes.

7.1. *Strongly Unforgeable One-Time Signatures*

Definition 14. (*Signatures*) A signature scheme consists of three algorithms.

Gen This is a randomized algorithm that takes as input the security parameter 1^λ. It outputs a verification key verk and a signing key sigk.

Sig This is a randomized algorithm that takes as input a signing key sigk and a message m (in some implicit message space). It outputs a signature σ.

Ver This takes as input a verification key verk, a message m, and a signature σ, and outputs a boolean value accept := 1 or reject := 0.

A signature scheme should have the following correctness property: for all $(\mathsf{verk}, \mathsf{sigk}) \xleftarrow{\mathsf{R}} \mathsf{Gen}(1^\lambda)$, all messages m, and all signatures $\sigma \xleftarrow{\mathsf{R}} \mathsf{Sig}(\mathsf{sigk}, m)$, it holds that $1 = \mathsf{Ver}(\mathsf{verk}, m, \sigma)$ with probability 1.

Definition 15. (*Strongly unforgeable one-time signatures*) For an adversary, we define $\mathsf{Adv}^{\mathsf{OS,SUF}}_{\mathcal{A}}(\lambda)$ to be the success probability in the following experiment for any security parameter λ. A signature scheme is a strongly unforgeable one-time signature scheme if the success probability of any polynomial-time adversary is negligible:

1. Run $(\mathsf{verk}, \mathsf{sigk}) \xleftarrow{\mathsf{R}} \mathsf{Gen}(1^\lambda)$ and give verk to the adversary.
2. The adversary is given access to signing oracle $\mathsf{Sig}(\mathsf{sigk}, \cdot)$ *at most once*. We denote the pair of message and signature by (m, σ) if the signing oracle is queried.
3. At the end, the adversary outputs (m', σ').

We say the adversary succeeds if $\mathsf{Ver}(\mathsf{verk}, m', \sigma') = 1$ and $(m', \sigma') \neq (m, \sigma)$ (assuming the signing oracle is queried).

7.2. *Construction*

$\widetilde{\rho} : \{1, \ldots, \ell\} \to \{1, \ldots, d\}$ is defined at the start of Sect. 4. In the proposed scheme, we assume that $\widetilde{\rho}$ is injective for $\mathbb{S} := (M, \rho)$.

In the description of the scheme, we assume that an input vector, $\vec{x}_t := (x_{t,1}, \ldots, x_{t,n_t})$, is normalized such that $x_{t,1} := 1$. (If $\vec{x}_t$ is not normalized, change it to a normalized one by $(1/x_{t,1}) \cdot \vec{x}_t$, assuming that $x_{t,1}$ is nonzero). In addition, we assume that input vector $\vec{v}_t := (v_{i,1}, \ldots, v_{i,n_t})$ satisfies that $v_{i,n_t} \neq 0$.

Random dual basis generator $\mathcal{G}_{\mathsf{ob}}(1^\lambda, \vec{n})$ is defined at the end of Sect. 2.1. We refer to Sect. 1.5 for notations on DPVS, e.g., $(x_1, \ldots, x_N)_{\mathbb{B}}$, $(y_1, \ldots, y_N)_{\mathbb{B}^*}$ for $x_i, y_i \in \mathbb{F}_q$, and $\vec{e}_{t,j}$.

For simplicity, we assume verification key verk is an element in $\mathbb{F}_q$. (We can extend the construction to verification key over any distribution D by first hashing verk using a collision resistant hash $H : \mathsf{D} \to \mathbb{F}_q$.)

$\mathsf{Setup}(1^\lambda, \vec{n} := (d; n_1, \ldots, n_d))$:

$n_{d+1} := 2$, $\vec{n}' := (d+1; \{n_t\}_{t=1,\ldots,d+1})$, $(\mathsf{param}_{\vec{n}'}, \{\mathbb{B}_t, \mathbb{B}^*_t\}_{t=0,\ldots,d+1}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}(1^\lambda, \vec{n}')$,

$\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5})$, $\widehat{\mathbb{B}}_t := (\boldsymbol{b}_{t,1}, \ldots, \boldsymbol{b}_{t,n_t}, \boldsymbol{b}_{t,3n_t+1})$ for $t = 1, \ldots, d+1$,

$\widehat{\mathbb{B}}^*_0 := (\boldsymbol{b}^*_{0,1}, \boldsymbol{b}^*_{0,3}, \boldsymbol{b}^*_{0,4})$, $\widehat{\mathbb{B}}^*_t := (\boldsymbol{b}^*_{t,1}, \ldots, \boldsymbol{b}^*_{t,n_t}, \boldsymbol{b}^*_{t,2n_t+1}, \ldots, \boldsymbol{b}^*_{t,3n_t})$ for $t = 1, \ldots, d+1$,

$\mathsf{pk} := (1^\lambda, \mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}_t\}_{t=0,\ldots,d+1})$, $\mathsf{sk} := \{\widehat{\mathbb{B}}^*_t\}_{t=0,\ldots,d+1}$,

return pk, sk.

$\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \Gamma := \{(t, \vec{x}_t := (x_{t,1}, \ldots, x_{t,n_t}) \in \mathbb{F}_q^{n_t} \backslash \{\vec{0}\}) \mid 1 \le t \le d,\ x_{t,1} := 1\})$:

$\delta, \varphi_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $\vec{\varphi}_t \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_t}$ such that $(t, \vec{x}_t) \in \Gamma$, $\vec{\varphi}_{d+1,1}, \vec{\varphi}_{d+1,2} \xleftarrow{\mathsf{U}} \mathbb{F}_q^2$

$\boldsymbol{k}_0 := (\delta,\ 0,\ 1,\ \varphi_0,\ 0)_{\mathbb{B}^*_0}$,

$\boldsymbol{k}^*_t := (\overbrace{\quad \delta\vec{x}_t, \quad}^{n_t} \overbrace{\quad 0^{n_t}, \quad}^{n_t} \overbrace{\quad \vec{\varphi}_t, \quad}^{n_t} \overbrace{\quad 0 \quad}^{1})_{\mathbb{B}^*_t}$ for $(t, \vec{x}_t) \in \Gamma$,

$\boldsymbol{k}^*_{d+1,1} := (\delta(1,0),\ 0^2,\ \vec{\varphi}_{d+1,1},\ 0)_{\mathbb{B}^*_{d+1}}$, $\boldsymbol{k}^*_{d+1,2} := (\delta(0,1),\ 0^2,\ \vec{\varphi}_{d+1,2},\ 0)_{\mathbb{B}^*_{d+1}}$,

$\mathsf{sk}_\Gamma := (\Gamma, \boldsymbol{k}_0^*, \{\boldsymbol{k}_t^*\}_{(t,\vec{x}_t)\in\Gamma}, \boldsymbol{k}_{d+1,1}^*, \boldsymbol{k}_{d+1,2}^*)$,

return sk_Γ.

$\mathsf{Enc}(\mathsf{pk},\ m,\ \mathbb{S} := (M,\rho)):\ \ \vec{f} \xleftarrow{\mathsf{R}} \mathbb{F}_q^r,\ \vec{s}^{\mathrm{T}} := (s_1,\ldots,s_\ell)^{\mathrm{T}} := M\cdot\vec{f}^{\mathrm{T}},\ s_0 := \vec{1}\cdot\vec{f}^{\mathrm{T}},$

$s_{\ell+1}, \eta_0, \eta_i, \theta_i, \zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$ for $i = 1,\ldots,\ell+1$, $(\mathsf{sigk},\mathsf{verk}) \xleftarrow{\mathsf{R}} \mathsf{Gen}(1^\lambda)$,

$\boldsymbol{c}_0 := (-s_0 - s_{\ell+1},\ 0,\ \zeta,\ 0,\ \eta_0)_{\mathbb{B}_0}$,

for $i = 1,\ldots,\ell$,

if $\rho(i) = (t, \vec{v}_i := (v_{i,1},\ldots,v_{i,n_t}) \in \mathbb{F}_q^{n_t}\backslash\{\vec{0}\})$ $(v_{i,n_t} \neq 0)$,

$$\boldsymbol{c}_i := (\overbrace{\quad s_i\vec{e}_{t,1} + \theta_i\vec{v}_i,\quad}^{n_t}\ \overbrace{\quad 0^{n_t},\quad}^{n_t}\ \overbrace{\quad 0^{n_t},\quad}^{n_t}\ \overbrace{\quad \eta_i\quad}^{1})_{\mathbb{B}_t},$$

if $\rho(i) = \neg(t, \vec{v}_i)$,

$$\boldsymbol{c}_i := (\overbrace{\quad s_i\vec{v}_i,\quad}^{n_t}\ \overbrace{\quad 0^{n_t},\quad}^{n_t}\ \overbrace{\quad 0^{n_t},\quad}^{n_t}\ \overbrace{\quad \eta_i\quad}^{1})_{\mathbb{B}_t},$$

$\boldsymbol{c}_{\ell+1} := (s_{\ell+1} - \theta_{\ell+1}\cdot\mathsf{verk},\ \theta_{\ell+1},\ 0^2,\ 0^2,\ \eta_{\ell+1})_{\mathbb{B}_{d+1}}$,

$c_{d+2} := g_T^\zeta m,\ \ C := (\mathbb{S}, \boldsymbol{c}_0,\ldots,\boldsymbol{c}_{\ell+1}, c_{d+2}),\ \ \sigma \xleftarrow{\mathsf{R}} \mathsf{Sig}(\mathsf{sigk}, C)$,

return $\mathsf{ct}_\mathbb{S} := (\mathsf{verk}, C, \sigma)$.

$\mathsf{Dec}(\mathsf{pk},\ \mathsf{sk}_\Gamma := (\Gamma, \boldsymbol{k}_0^*, \{\boldsymbol{k}_t^*\}_{(t,\vec{x}_t)\in\Gamma}, \boldsymbol{k}_{d+1,1}^*, \boldsymbol{k}_{d+1,2}^*),\ \mathsf{ct}_\mathbb{S} := (\mathsf{verk}, (\mathbb{S}, \boldsymbol{c}_0,\ldots,\boldsymbol{c}_{\ell+1}, c_{d+2}), \sigma)):$

if $\mathsf{Ver}(\mathsf{verk}, C, \sigma) \neq 1$, return $\perp$, where $C := (\mathbb{S}, \boldsymbol{c}_0,\ldots,\boldsymbol{c}_{\ell+1}, c_{d+2})$,

if $\mathbb{S} := (M,\rho)$ accepts $\Gamma := \{(t,\vec{x}_t)\}$, then compute I and $\{\alpha_i\}_{i\in I}$ such that

$\vec{1} = \sum_{i\in I}\alpha_i M_i$, where M_i is the ith row of M, and

$I \subseteq \{i \in \{1,\ldots,\ell\} |\ [\rho(i) = (t,\vec{v}_i)\ \wedge\ (t,\vec{x}_t)\in\Gamma\ \wedge\ \vec{v}_i\cdot\vec{x}_t = 0]$

$\qquad\qquad \vee\ [\rho(i) = \neg(t,\vec{v}_i)\ \wedge\ (t,\vec{x}_t)\in\Gamma\ \wedge\ \vec{v}_i\cdot\vec{x}_t \neq 0]\ \}$,

$\boldsymbol{s}_{d+1}^* := \boldsymbol{k}_{d+1,1}^* + \mathsf{verk}\cdot\boldsymbol{k}_{d+1,2}^*$,

$K := e(\boldsymbol{c}_0, \boldsymbol{k}_0^*)\prod_{i\in I\ \wedge\ \rho(i)=(t,\vec{v}_i)} e(\boldsymbol{c}_i, \boldsymbol{k}_t^*)^{\alpha_i}\cdot\prod_{i\in I\ \wedge\ \rho(i)=\neg(t,\vec{v}_i)} e(\boldsymbol{c}_i, \boldsymbol{k}_t^*)^{\alpha_i/(\vec{v}_i\cdot\vec{x}_t)}\cdot e(\boldsymbol{c}_{\ell+1}, \boldsymbol{s}_{d+1}^*)$,

return $m' := c_{d+1}/K$.

[Correctness] If $\mathbb{S} := (M,\rho)$ accepts $\Gamma := \{(t,\vec{x}_t)\}$,

$$\begin{aligned}
&e(\boldsymbol{c}_0, \boldsymbol{k}_0^*)\prod_{i\in I\ \wedge\ \rho(i)=(t,\vec{v}_i)} e(\boldsymbol{c}_i, \boldsymbol{k}_t^*)^{\alpha_i}\cdot\prod_{i\in I\ \wedge\ \rho(i)=\neg(t,\vec{v}_i)} e(\boldsymbol{c}_i, \boldsymbol{k}_t^*)^{\alpha_i/(\vec{v}_i\cdot\vec{x}_t)}\cdot e(\boldsymbol{c}_{\ell+1}, \boldsymbol{s}_{d+1}^*)\\
&= g_T^{\delta(-s_0-s_{\ell+1})+\zeta}\prod_{i\in I\ \wedge\ \rho(i)=(t,\vec{v}_i)} g_T^{\delta\alpha_i s_i}\prod_{i\in I\ \wedge\ \rho(i)=\neg(t,\vec{v}_i)} g_T^{\delta\alpha_i s_i(\vec{v}_i\cdot\vec{x}_t)/(\vec{v}_i\cdot\vec{x}_t)} g_T^{\delta s_{\ell+1}}\\
&= g_T^{\delta(-s_0-s_{\ell+1}+\sum_{i\in I}\alpha_i s_i+s_{\ell+1})+\zeta} = g_T^\zeta.
\end{aligned}$$

7.3. *Security*

Theorem 4. *The proposed CP-FE scheme is adaptively payload-hiding against chosen-ciphertext attacks under the DLIN assumption provided that the underlying signature scheme* (*Gen*, *Sig*, *Ver*) *is a strongly unforgeable one-time signature scheme.*

For any adversary $\mathcal{A}$*, there exist probabilistic machines* $\mathcal{E}_1, \mathcal{E}_2^+, \mathcal{E}_2, \mathcal{E}_3, \mathcal{E}_4$*, whose running times are essentially the same as that of* $\mathcal{A}$*, such that for any security parameter* λ*,*

$$\mathsf{Adv}^{\mathit{CP\text{-}FE,\ CCA\text{-}PH}}_{\mathcal{A}}(\lambda) \le \mathsf{Adv}^{\mathit{DLIN}}_{\mathcal{E}_1}(\lambda) + \sum_{h=0}^{\nu_1-1}\left(\mathsf{Adv}^{\mathit{DLIN}}_{\mathcal{E}^+_{2,h}}(\lambda) + \mathsf{Adv}^{\mathit{DLIN}}_{\mathcal{E}_{2,h+1}}(\lambda)\right) + \sum_{h=1}^{\nu_2}\left(\mathsf{Adv}^{\mathit{DLIN}}_{\mathcal{E}_{3,h}}(\lambda) + \mathsf{Adv}^{\mathsf{OS,SUF}}_{\mathcal{E}_{4,h}}(\lambda)\right) + \epsilon,$$

where $\mathcal{E}^+_{2,h}(\cdot) := \mathcal{E}^+_2(h,\cdot), \mathcal{E}_{2,h+1}(\cdot) := \mathcal{E}_2(h,\cdot)$ $(h = 0,\ldots,\nu_1-1)$, $\mathcal{E}_{3,h}(\cdot) := \mathcal{E}_3(h,\cdot), \mathcal{E}_{4,h}(\cdot) := \mathcal{E}_4(h,\cdot)$ $(h = 1,\ldots,\nu_2)$, *ν_1 is the maximum number of $\mathcal{A}$'s* KeyGen *queries, ν_2 is the maximum number of $\mathcal{A}$'s* Dec *queries, and* $\epsilon := (2d\nu_1 + 16\nu_1 + 8\nu_2 + d + 10)/q$.

Proof Outline of Theorem 4: To prove Theorem 4, we consider the following $(2\nu_1 + \nu_2 + 3)$ 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 which were changed in a game from the previous game.

Game 0 : Original game. That is, the reply to a KeyGen query for $\Gamma := \{(t, \vec{x}_t)\}$ are:

$$\begin{aligned}
&\boldsymbol{k}^*_0 := (\delta,\ \boxed{0},\ 1,\ \varphi_0,\ 0)_{\mathbb{B}^*_0},\\
&\boldsymbol{k}^*_t := (\delta\vec{x}_t,\ \boxed{0^{n_t}},\ \vec{\varphi}_t,\ 0)_{\mathbb{B}_t} \ \text{ for } (t,\vec{x}_t)\in\Gamma,\\
&\boldsymbol{k}^*_{d+1,1} := (\delta(1,0),\ 0^2,\ \vec{\varphi}_{d+1,1},\ 0)_{\mathbb{B}^*_{d+1}},\quad \boldsymbol{k}^*_{d+1,2} := (\delta(0,1),\ 0^2,\ \vec{\varphi}_{d+1,2},\ 0)_{\mathbb{B}^*_{d+1}},
\end{aligned}$$

where $\delta \xleftarrow{\cup} \mathbb{F}_q^\times$, $\varphi_0 \xleftarrow{\cup} \mathbb{F}_q$, $\vec{\varphi}_t \xleftarrow{\cup} \mathbb{F}_q^{n_t}$ for $(t,\vec{x}_t)\in\Gamma, \vec{\varphi}_{d+1,1}, \vec{\varphi}_{d+1,2} \xleftarrow{\cup} \mathbb{F}_q^2$. In answering Dec query for $\mathsf{ct}_{\mathbb{S}} := (\mathsf{verk}, (\mathbb{S}, \boldsymbol{c}_0, \ldots, \boldsymbol{c}_{\ell+1}, c_{d+2}), \sigma)$ when $\mathsf{Ver}(\mathsf{verk}, C, \sigma) = 1$, where $C := (\mathbb{S}, \boldsymbol{c}_0, \ldots, \boldsymbol{c}_{\ell+1}, c_{d+2})$, the used key for $\Gamma := \{(t,\vec{x}_t)\}$ such that $\mathbb{S}$ accepts Γ are:

$$\begin{aligned}
&\boldsymbol{k}^*_0 := (\widetilde{\delta},\ \boxed{0},\ 1,\ \widetilde{\varphi}_0,\ 0)_{\mathbb{B}^*_0},\\
&\boldsymbol{k}^*_t := (\widetilde{\delta}\vec{x}_t,\ 0^{n_t},\ \vec{\widetilde{\varphi}}_t,\ 0)_{\mathbb{B}_t} \ \text{ for } (t,\vec{x}_t)\in\Gamma,\\
&\boldsymbol{s}^*_{d+1} := (\widetilde{\delta}(1,\mathsf{verk}),\ \boxed{0^2},\ \vec{\widetilde{\varphi}}_{d+1},\ 0)_{\mathbb{B}^*_{d+1}},
\end{aligned}$$

where $\widetilde{\delta} \xleftarrow{\cup} \mathbb{F}_q^\times$, $\widetilde{\varphi}_0 \xleftarrow{\cup} \mathbb{F}_q$, $\vec{\widetilde{\varphi}}_t \xleftarrow{\cup} \mathbb{F}_q^{n_t}$ for $(t,\vec{x}_t)\in\Gamma, \vec{\widetilde{\varphi}}_{d+1} \xleftarrow{\cup} \mathbb{F}_q^2$.

The challenge ciphertext for challenge plaintexts $(m^{(0)}, m^{(1)})$ and access structure $\mathbb{S} := (M, \rho)$ is:

$$\begin{aligned}
&\boldsymbol{c}_0 := (-s_0 - s_{\ell+1},\ \boxed{0},\ \boxed{\zeta},\ 0,\ \eta_0)_{\mathbb{B}_0},\\
&\text{for } i = 1,\ldots,\ell,\\
&\quad \text{if } \rho(i) = (t,\vec{v}_i),\quad \boldsymbol{c}_i := (s_i\vec{e}_{t,1} + \theta_i\vec{v}_i,\ \boxed{0^{n_t}},\ 0^{n_t},\ \eta_i)_{\mathbb{B}_t},\\
&\quad \text{if } \rho(i) = \neg(t,\vec{v}_i),\quad \boldsymbol{c}_i := (s_i\vec{v}_i,\ \boxed{0^{n_t}},\ 0^{n_t},\ \eta_i)_{\mathbb{B}_t},\\
&\boldsymbol{c}_{\ell+1} := (s_{\ell+1} - \theta_{\ell+1}\cdot\mathsf{verk},\ \theta_{\ell+1},\ \boxed{0^2},\ 0^2,\ \eta_{\ell+1})_{\mathbb{B}_{d+1}},\\
&c_{d+2} := g_T^{\zeta} m^{(b)},
\end{aligned}$$

where $\vec{f} \xleftarrow{\mathsf{R}} \mathbb{F}_q^r$, $\vec{s}^{\mathrm{T}} := (s_1, \ldots, s_\ell)^{\mathrm{T}} := M \cdot \vec{f}^{\mathrm{T}}$, $s_0 := \vec{1} \cdot \vec{f}^{\mathrm{T}}$, $s_{\ell+1}, \zeta, \eta_0, \eta_i, \theta_i \xleftarrow{\mathsf{U}} \mathbb{F}_q$ for $i = 1, \ldots, \ell+1$, and $\vec{e}_{t,1} := (1, 0, \ldots, 0) \in \mathbb{F}_q^{n_t}$.

Game 1 : Same as Game 0 except that the challenge ciphertext for challenge plaintexts $(m^{(0)}, m^{(1)})$ and access structure $\mathbb{S} := (M, \rho)$ is:

$$
\begin{aligned}
&\boldsymbol{c}_0 := (-s_0 - s_{\ell+1}, \boxed{w_0}, \zeta, 0, \eta_0)_{\mathbb{B}_0},\\
&\text{for } i = 1, \ldots, \ell,\\
&\quad \text{if } \rho(i) = (t, \vec{v}_i), \quad \boldsymbol{c}_i := (s_i \vec{e}_{t,1} + \theta_i \vec{v}_i, \boxed{\vec{w}_i}, 0^{n_t}, \eta_i)_{\mathbb{B}_t},\\
&\quad \text{if } \rho(i) = \neg(t, \vec{v}_i), \quad \boldsymbol{c}_i := (s_i \vec{v}_i, \boxed{\vec{\overline{w}}_i}, 0^{n_t}, \eta_i)_{\mathbb{B}_t},\\
&\boldsymbol{c}_{\ell+1} := (s_{\ell+1} - \theta_{\ell+1} \cdot \mathsf{verk}, \theta_{\ell+1}, \boxed{\vec{w}_{\ell+1}}, 0^2, \eta_{\ell+1})_{\mathbb{B}_{d+1}},
\end{aligned}
$$

where $w_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $\vec{w}_i, \vec{\overline{w}}_i \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_t}$ for $i = 1, \ldots, \ell$, $\vec{w}_{\ell+1} \xleftarrow{\mathsf{U}} \mathbb{F}_q^2$, and all the other variables are generated as in Game 0.

Game 2-h^+ ($h = 0, \ldots, v_1 - 1$) and **Game 2-($h+1$) ($h = 0, \ldots, v_1 - 1$)** are the same as **Game 2-h^+** and **Game 2-($h+1$)** in the proof of Theorem 2, respectively.

Game 3-h ($h = 1, \ldots, v_2$) : Game 3-0 is Game 2-ν_1. Game 3-h is the same as Game 3-$(h-1)$ except that $\boldsymbol{k}_0^*, s_{d+1}^*$ of the key used in answering the hth Dec query when $\mathsf{Ver}(\mathsf{verk}, C, \sigma) = 1$ are:

$$
\begin{aligned}
&\boldsymbol{k}_0^* := (\widetilde{\delta}, \boxed{\widetilde{r}_0}, 1, \widetilde{\varphi}_0, 0)_{\mathbb{B}_0^*},\\
&\boldsymbol{s}_{d+1}^* := (\widetilde{\delta}(1, \mathsf{verk}), \boxed{\vec{\widetilde{r}}_{d+1}}, \vec{\widetilde{\varphi}}_{d+1}, 0)_{\mathbb{B}_{d+1}^*},
\end{aligned}
$$

where $\widetilde{r}_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $\vec{\widetilde{r}}_{d+1} \xleftarrow{\mathsf{U}} \mathbb{F}_q^2$, and all the other variables are generated as in Game 3-$(h-1)$.

Game 4 : Same as Game 3-ν_2 except that $\boldsymbol{c}_0$ in the challenge ciphertext is:

$$
\boldsymbol{c}_0 := (-s_0 - s_{\ell+1}, w_0, \boxed{\zeta'}, 0, \eta_0)_{\mathbb{B}_0},
$$

where $\zeta' \xleftarrow{\mathsf{U}} \mathbb{F}_q$ (i.e., independent from all the other variables), and all the other variables are generated as in Game 3-ν_2.

We follow the argument in [17] used for the chosen-ciphertext security, and the rest of the proof of Theorem 4 is similar to that of Theorem 2.

Let $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$ be $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{CP\text{-}FE},\,\mathsf{CCA\text{-}PH}}(\lambda)$ in Game 0, and $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2-h^+)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2-h)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(3-h)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(4)}(\lambda)$ be the advantage of $\mathcal{A}$ in Game $1, 2-h, 2-h^+, 3-h, 4$, respectively. ($\mathsf{Adv}_{\mathcal{A}}^{(4)}(\lambda) = 0$.) We can evaluate the gaps between pairs of $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2-h)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2-h^+)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2-(h+1))}(\lambda)$ for $h = 0, \ldots, \nu_1 - 1$ using Problems 3 and 4 (given in "Appendix D") as in the proof of Theorem 2.

Moreover, we can evaluate the gaps between pairs of $\mathsf{Adv}_{\mathcal{A}}^{(3-h)}(\lambda)$ and $\mathsf{Adv}_{\mathcal{A}}^{(3-(h+1))}(\lambda)$ for $h = 0, \ldots, \nu_2 - 1$ using Problem 5 in "Appendix D". □

A. Dual Pairing Vector Spaces (DPVS)

A.1. *Summary*

We now briefly explain our approach, DPVS, constructed on symmetric pairing groups $(q, \mathbb{G}, \mathbb{G}_T, G, e)$, where q is a prime, $\mathbb{G}$ and $\mathbb{G}_T$ are cyclic groups of order q, G is a generator of $\mathbb{G}$, $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is a non-degenerate bilinear pairing operation, and $e(G, G) \neq 1$. Here we denote the group operation of $\mathbb{G}$ by addition and $\mathbb{G}_T$ by multiplication, respectively. Note that this construction also works on *asymmetric* pairing groups (in this paper, we use symmetric pairing groups for simplicity of description).

Vector space $\mathbb{V}$: $\mathbb{V} := \overbrace{\mathbb{G} \times \cdots \times \mathbb{G}}^{N}$, whose element is expressed by N-dimensional vector, $\boldsymbol{x} := (x_1 G, \ldots, x_N G)$ $(x_i \in \mathbb{F}_q$ for $i = 1, \ldots, N)$.

Canonical base $\mathbb{A}$: $\mathbb{A} := (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N)$ of $\mathbb{V}$, where $\boldsymbol{a}_1 := (G, 0, \ldots, 0), \boldsymbol{a}_2 := (0, G, 0, \ldots, 0), \ldots, \boldsymbol{a}_N := (0, \ldots, 0, G)$.

Pairing operation: $e(\boldsymbol{x}, \boldsymbol{y}) := \prod_{i=1}^{N} e(x_i G, y_i G) = e(G, G)^{\sum_{i=1}^{N} x_i y_i} = e(G, G)^{\vec{x} \cdot \vec{y}} \in \mathbb{G}_T$, where $\boldsymbol{x} := (x_1 G, \ldots, x_N G) = x_1 \boldsymbol{a}_1 + \cdots + x_N \boldsymbol{a}_N \in \mathbb{V}$, $\boldsymbol{y} := (y_1 G, \ldots, y_N G) = y_1 \boldsymbol{a}_1 + \cdots + y_N \boldsymbol{a}_N \in \mathbb{V}$, $\vec{x} := (x_1, \ldots, x_N)$ and $\vec{y} := (y_1, \ldots, y_N)$. Here, $\boldsymbol{x}$ and $\boldsymbol{y}$ can be expressed by coefficient vector over basis $\mathbb{A}$ such that $(x_1, \ldots, x_N)_{\mathbb{A}} = (\vec{x})_{\mathbb{A}} := \boldsymbol{x}$ and $(y_1, \ldots, y_N)_{\mathbb{A}} = (\vec{y})_{\mathbb{A}} := \boldsymbol{y}$.

Base change: Canonical basis $\mathbb{A}$ is changed to basis $\mathbb{B} := (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_N)$ of $\mathbb{V}$ using a uniformly chosen (regular) linear transformation, $X := (\chi_{i,j}) \xleftarrow{\cup} GL(N, \mathbb{F}_q)$, such that $\boldsymbol{b}_i = \sum_{j=1}^{N} \chi_{i,j} \boldsymbol{a}_j$, $(i = 1, \ldots, N)$. $\mathbb{A}$ is also changed to basis $\mathbb{B}^* := (\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_N^*)$ of $\mathbb{V}$, such that $(\vartheta_{i,j}) := (X^T)^{-1}$, $\boldsymbol{b}_i^* = \sum_{j=1}^{N} \vartheta_{i,j} \boldsymbol{a}_j$, $(i = 1, \ldots, N)$. We see that $e(\boldsymbol{b}_i, \boldsymbol{b}_j^*) = e(G, G)^{\delta_{i,j}}$, $(\delta_{i,j} = 1$ if $i = j$, and $\delta_{i,j} = 0$ if $i \neq j)$, i.e., $\mathbb{B}$ and $\mathbb{B}^*$ are dual orthonormal bases of $\mathbb{V}$.

Here, $\boldsymbol{x} := x_1 \boldsymbol{b}_1 + \cdots + x_N \boldsymbol{b}_N \in \mathbb{V}$ and $\boldsymbol{y} := y_1 \boldsymbol{b}_1^* + \cdots + y_N \boldsymbol{b}_N^* \in \mathbb{V}$ can be expressed by coefficient vectors over $\mathbb{B}$ and $\mathbb{B}^*$ such that $(x_1, \ldots, x_N)_{\mathbb{B}} = (\vec{x})_{\mathbb{B}} := \boldsymbol{x}$ and $(y_1, \ldots, y_N)_{\mathbb{B}^*} = (\vec{y})_{\mathbb{B}^*} := \boldsymbol{y}$, and $e(\boldsymbol{x}, \boldsymbol{y}) = e(G, G)^{\sum_{i=1}^{N} x_i y_i} = e(G, G)^{\vec{x} \cdot \vec{y}} \in \mathbb{G}_T$.

Intractable problem: One of the most natural decisional problems in this approach is the decisional subspace problem [35]. It is to tell $\boldsymbol{v} := v_{N_2+1} \boldsymbol{b}_{N_2+1} + \cdots + v_{N_1} \boldsymbol{b}_{N_1}$ $(= (0, \ldots, 0, v_{N_2+1}, \ldots, v_{N_1})_{\mathbb{B}})$, from $\boldsymbol{u} := v_1 \boldsymbol{b}_1 + \cdots + v_{N_1} \boldsymbol{b}_{N_1}$ $(= (v_1, \ldots, v_{N_1})_{\mathbb{B}})$, where $(v_1, \ldots, v_{N_1}) \xleftarrow{\cup} \mathbb{F}_q^{N_1}$ and $N_2 + 1 < N_1$.

Trapdoor: Although the decisional subspace problem is assumed to be intractable, it can be efficiently solved using *trapdoor* $\boldsymbol{t}^* \in \mathsf{span}\langle \boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_{N_2}^* \rangle$. Given $\boldsymbol{v} := v_{N_2+1}\boldsymbol{b}_{N_2+1} + \cdots + v_{N_1}\boldsymbol{b}_{N_1}$ or $\boldsymbol{u} := v_1\boldsymbol{b}_1 + \cdots + v_{N_1}\boldsymbol{b}_{N_1}$, we can tell $\boldsymbol{v}$ from $\boldsymbol{u}$ using $\boldsymbol{t}^*$ since $e(\boldsymbol{v}, \boldsymbol{t}^*) = 1$ and $e(\boldsymbol{u}, \boldsymbol{t}^*) \neq 1$ with high probability.

Advantage of this approach: Higher-dimensional vector treatment of bilinear pairing groups have been already employed in literature especially in the areas of IBE, ABE and BE (e.g., [5,8,12,16,28,44]). For example, in a typical vector treatment, two vector forms of $P := (x_1 G, \ldots, x_N G)$ and $Q := (y_1 G, \ldots, y_N G)$ are set and pairing for P and Q is operated as $e(P, Q) := \prod_{i=1}^{N} e(x_i G, y_i G)$. Such treatment can be rephrased in this approach such that $P = x_1\boldsymbol{a}_1 + \cdots + x_N\boldsymbol{a}_N$ $(= (x_1, \ldots, x_N)_{\mathbb{A}})$, and $Q = y_1\boldsymbol{a}_1 + \cdots + y_N\boldsymbol{a}_N$ $(= (y_1, \ldots, y_N)_{\mathbb{A}})$ over canonical basis $\mathbb{A}$.

The major drawback of this approach is the easily *decomposable* property over $\mathbb{A}$ (i.e., the decisional subspace problem is easily solved). That is, it is easy to decompose $x_i\boldsymbol{a}_i = (0, \ldots, 0, x_i G, 0, \ldots, 0)$ from $P := x_1\boldsymbol{a}_1 + \cdots x_N\boldsymbol{a}_N = (x_1 G, \ldots, x_N G)$.

In contrast, our approach employs basis $\mathbb{B}$, which is linearly transformed from $\mathbb{A}$ using a secret random matrix $X \in \mathbb{F}_q^{n \times n}$. A remarkable property over $\mathbb{B}$ is that it seems hard to decompose $x_i\boldsymbol{b}_i$ from $P' := x_1\boldsymbol{b}_1 + \cdots x_N\boldsymbol{b}_N$ (and the decisional subspace problem seems intractable). In addition, the secret matrix X (and the dual orthonormal basis $\mathbb{B}^*$ of $\mathbb{V}$) can be used as a source of the trapdoors to the decomposability (and distinguishability for the decisional subspace problem through the pairing operation over $\mathbb{B}$ and $\mathbb{B}^*$ as mentioned above). The hard decomposability (and indistinguishability) and its trapdoors are ones of the key tricks in this paper. Note that composite-order pairing groups are often employed with similar tricks such as hard decomposability (and indistinguishability) of a composite-order group to the prime-order subgroups and its trapdoors through factoring (e.g., [30,45]).

A.2. *Dual Pairing Vector Spaces by Direct Product of Asymmetric Pairing Groups*

Definition 16. "Asymmetric bilinear pairing groups" $(q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, G_1, G_2, e)$ are a tuple of a prime q, cyclic additive groups $\mathbb{G}_1, \mathbb{G}_2$ and multiplicative group $\mathbb{G}_T$ of order q, $G_1 \neq 0 \in \mathbb{G}_1, G_2 \neq 0 \in \mathbb{G}_2$, and a polynomial-time computable non-degenerate bilinear pairing $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$, i.e., $e(sG_1, tG_2) = e(G_1, G_2)^{st}$ and $e(G_1, G_2) \neq 1$.

Let $\mathcal{G}_{\mathsf{bpg}}$ be an algorithm that takes input 1^λ and outputs a description of bilinear pairing groups $\mathsf{param}_{\mathbb{G}} := (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, G_1, G_2, e)$ with security parameter λ.

Definition 17. "Dual pairing vector spaces (DPVS)" $(q, \mathbb{V}, \mathbb{V}^*, \mathbb{G}_T, \mathbb{A}, \mathbb{A}^*, e)$ by direct product of asymmetric pairing groups $\mathsf{param}_{\mathbb{G}} := (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, G_1, G_2, e)$ are a tuple of a prime q, two N-dimensional vector spaces $\mathbb{V} := \overbrace{\mathbb{G}_1 \times \cdots \times \mathbb{G}_1}^{N}$ and $\mathbb{V}^* := \overbrace{\mathbb{G}_2 \times \cdots \times \mathbb{G}_2}^{N}$ over $\mathbb{F}_q$, a cyclic group $\mathbb{G}_T$ of order q, and their canonical bases, i.e., $\mathbb{A} :=$

$(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N)$ of $\mathbb{V}$ and $\mathbb{A}^* := (\boldsymbol{a}_1^*, \ldots, \boldsymbol{a}_N^*)$ of $\mathbb{V}^*$, where $\boldsymbol{a}_i := (\overbrace{0, \ldots, 0}^{i-1}, G_1, \overbrace{0, \ldots, 0}^{N-i})$ and $\boldsymbol{a}_i^* := (\overbrace{0, \ldots, 0}^{i-1}, G_2, \overbrace{0, \ldots, 0}^{N-i})$, and pairing $e : \mathbb{V} \times \mathbb{V}^* \to \mathbb{G}_T$.

The pairing is defined by $e(\boldsymbol{x}, \boldsymbol{y}) := \prod_{i=1}^{N} e(D_i, H_i) \in \mathbb{G}_T$ where $\boldsymbol{x} := (D_1, \ldots, D_N) \in \mathbb{V}$ and $\boldsymbol{y} := (H_1, \ldots, H_N) \in \mathbb{V}^*$. This is non-degenerate 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, $e(\boldsymbol{a}_i, \boldsymbol{a}_j^*) = g_T^{\delta_{i,j}}$ where $\delta_{i,j} = 1$ if $i = j$, and 0 otherwise, and $e(G_1, G_2) \neq 1 \in \mathbb{G}_T$.

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

Right multiplication by $W \in GL(N, \mathbb{F}_q)$ is defined as in Remark 1 in Sect. 2.1.

B. Hierarchical Reduction to DLIN: Proofs of Main Lemmas (Lemmas 1 and 2)

B.1. *Outline*

The DLIN Problem is reduced to (complicated) Problems 1 and 2 through several intermediate steps, or intermediate problems, as indicated below (See Fig. 1 in Sect. 4.3):

1. DLIN Problem (in Definition 3)
2. Basic Problem 0 with three-dimensional DPVS (in Definition 18)
3. Basic Problems 1 and 2 with $\vec{n} := (d; n_1, \ldots, n_d)$ (in Definitions 19 and 20)
4. Problems 1 and 2 with $\vec{n}$ (in Definitions 4, 5)

We will explain how the simplest problem, DLIN, is sequentially transformed to more complicated ones according to parameter $\vec{n}$, which indicates degree of complexity.

DLIN → Basic Problem 0 : Basic Problem 0 uses three-dimensional DPVS. In this first reduction step, a DLIN instance on (symmetric) pairing group is transformed to a Basic Problem 0 instance on the DPVS, i.e., higher-level concept. It is proven in Lemma 14.

Basic Problem 0 → Basic Problems 1 and 2 : Format $\vec{n} := (d; n_1, \ldots, n_d)$ corresponds to $d + 1$ DPVSs, $\mathbb{V}_t$ ($t = 0, \ldots, d$). The dimension of $\mathbb{V}_0$ is 5, and the dimensions of $\mathbb{V}_t$ are $3n_t + 1$ for $t = 1, \ldots, d$. In this reduction step, vector elements (and additional group elements) in a Basic Problem 0 instance are transformed to the corresponding elements in $\mathbb{V}_t$ for $t = 0, \ldots, d$. They are proven in Lemmas 15 and 17.

Basic Problem 1 → Problem 1 : The proof is given in Lemmas 16.

Basic Problem 2 → Problem 2 : The proof is given in Lemma 18.

B.2. *Preliminary Lemma*

We will use the following lemma (Lemma 14) in the proofs of Lemmas 1 and 2.

Definition 18. (*Basic Problem 0*) Basic Problem 0 is to guess $\beta \in \{0,1\}$, given $(\mathsf{param}_{\mathsf{BP0}}, \widehat{\mathbb{B}}, \mathbb{B}^*, \boldsymbol{y}^*_\beta, \boldsymbol{f}, \kappa G, \xi G, \delta\xi G) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{BP0}}_\beta(1^\lambda)$, where

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

for $\beta \xleftarrow{\mathsf{U}} \{0,1\}$. For a probabilistic machine $\mathcal{D}$, we define the advantage of $\mathcal{D}$ for Basic Problem 0, $\mathsf{Adv}^{\mathsf{BP0}}_{\mathcal{D}}(\lambda)$, is similarly defined as in Definition 4.

Lemma 14. *For any adversary $\mathcal{D}$, 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, $\mathsf{Adv}^{\mathsf{BP0}}_{\mathcal{D}}(\lambda) \leq \mathsf{Adv}^{\mathit{DLIN}}_{\mathcal{E}}(\lambda) + 5/q$.*

Proof. Given a DLIN instance

$$(\mathsf{param}_{\mathbb{G}},\ G, \xi G, \kappa G, \delta\xi G, \sigma\kappa G, Y_\beta),$$

$\mathcal{E}$ calculates

$$
\begin{aligned}
&\mathsf{param}_{\mathbb{V}} := (q, \mathbb{V}, \mathbb{G}_T, \mathbb{A}, e) := \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, 3, \mathsf{param}_{\mathbb{G}}),\\
&g_T := e(\kappa G, \xi G)\ \ \left(= e(G,G)^{\kappa\xi}\right),\ \ \mathsf{param}_{\mathsf{BP0}} := (\mathsf{param}_{\mathbb{V}},\ g_T).
\end{aligned}
$$

$\mathcal{E}$ sets 3×3 matrices Π^*, Π as follows:

$$\Pi^* := \begin{pmatrix} \xi & & 1 \\ & 1 & \\ \kappa & & 1 \end{pmatrix}, \quad \Pi := \begin{pmatrix} \kappa & & \\ -\kappa & -\xi & \kappa\xi \\ & \xi & \end{pmatrix},$$

Then, $\Pi \cdot (\Pi^*)^{\mathrm{T}} = \kappa\xi \cdot I_3$. By using matrices Π and Π^*, $\mathcal{E}$ sets

$$
\begin{aligned}
&\boldsymbol{u}^*_1 := (\xi, 0, 1)_{\mathbb{A}},\ \ \boldsymbol{u}^*_2 := (0,0,1)_{\mathbb{A}},\ \ \boldsymbol{u}^*_3 := (0, \kappa, 1)_{\mathbb{A}},\\
&\boldsymbol{u}_1 := (\kappa, 0, 0)_{\mathbb{A}},\ \ \boldsymbol{u}_2 := (-\kappa, -\xi, \kappa\xi)_{\mathbb{A}},\ \ \boldsymbol{u}_3 := (0, \xi, 0)_{\mathbb{A}},
\end{aligned}
$$

$\mathcal{E}$ can compute $\boldsymbol{u}_i^*$ for $i = 1, 2, 3$ and $\boldsymbol{u}_i$ for $i = 1, 3$ from the above DLIN instance. Let bases $\mathbb{U} := (\boldsymbol{u}_i)_{i=1,2,3}$, $\mathbb{U}^* := (\boldsymbol{u}_i^*)_{i=1,2,3}$ of $\mathbb{V}$. $\mathcal{E}$ then generates $\eta, \varphi \xleftarrow{\mathsf{U}} \mathbb{F}_q$ such that $\eta \neq 0$, and sets

$$\boldsymbol{v} := (\varphi G, -\eta G, \eta(\kappa G)) \quad (= (\varphi, -\eta, \eta\kappa)_{\mathbb{A}}) \quad \text{and} \quad \boldsymbol{w}_\beta^* := (\delta\xi G, \sigma\kappa G, Y_\beta).$$

$\mathcal{E}$ generates a random matrix $W \xleftarrow{\mathsf{U}} GL(3, \mathbb{F}_q)$, then calculates

$$\begin{aligned}
&\boldsymbol{b}_i := \boldsymbol{u}_i W \ \text{ for } i = 1, 3, \quad \boldsymbol{b}_i^* := \boldsymbol{u}_i^* (W^{-1})^{\mathrm{T}} \ \text{ for } i = 1, 2, 3,\\
&\widehat{\mathbb{B}} := (\boldsymbol{b}_1, \boldsymbol{b}_3). \quad \mathbb{B}^* := (\boldsymbol{b}_1^*, \boldsymbol{b}_2^*, \boldsymbol{b}_3^*),\\
&\boldsymbol{f} = \boldsymbol{v} W, \quad \boldsymbol{y}_\beta^* = \boldsymbol{w}_\beta^* (W^{-1})^{\mathrm{T}},
\end{aligned}$$

where right multiplication by W (and $(W^{-1})^{\mathrm{T}}$) is given as in Remark 1 in Sect. 2.1. $\mathcal{E}$ then gives $(\mathsf{param}_{\mathsf{BP0}}, \widehat{\mathbb{B}}, \mathbb{B}^*, \boldsymbol{y}_\beta^*, \boldsymbol{f}, \kappa G, \xi G, \delta\xi G)$ to $\mathcal{D}$, where $\delta\xi G$ is contained in the DLIN instance, and outputs $\beta' \in \{0, 1\}$ if $\mathcal{D}$ outputs β'.

If we set

$$\tau := \xi^{-1}\eta, \quad \omega := \tau + \kappa^{-1}\varphi,$$

then $\tau \neq 0$ (since $\eta \neq 0$),

$$\begin{aligned}
&\boldsymbol{v} = (\varphi, -\eta, \eta\kappa)_{\mathbb{A}} = ((\omega - \tau)\kappa, -\tau\xi, \tau\kappa\xi)_{\mathbb{A}} = \omega\boldsymbol{u}_1 + \tau\boldsymbol{u}_2 = (\omega, \tau, 0)_{\mathbb{U}}, \ \text{ and}\\
&\boldsymbol{f} = \boldsymbol{v} W = ((\omega, \tau, 0)_{\mathbb{U}})\, W = (\omega, \tau, 0)_{\mathbb{B}}.
\end{aligned}$$

If $\beta = 0$, i.e., $Y_\beta = Y_0 = (\delta + \sigma)G$, then

$$\begin{aligned}
&\boldsymbol{w}_0^* = (\delta\xi G, \sigma\kappa G, (\delta + \sigma)G) = (\delta\xi, \sigma\kappa, \delta + \sigma)_{\mathbb{A}} = \delta\boldsymbol{u}_1^* + \sigma\boldsymbol{u}_3^* = (\delta, 0, \sigma)_{\mathbb{U}^*} \ \text{ and}\\
&\boldsymbol{y}_0^* = \boldsymbol{w}_0^* (W^{-1})^{\mathrm{T}} = ((\delta, 0, \sigma)_{\mathbb{U}^*})\, (W^{-1})^{\mathrm{T}} = (\delta, 0, \sigma)_{\mathbb{B}^*}.
\end{aligned}$$

Therefore, the distribution of $(\mathsf{param}_{\mathsf{BP0}}, \widehat{\mathbb{B}}, \mathbb{B}^*, \boldsymbol{y}_0^*, \boldsymbol{f}, \kappa G, \xi G, \delta\xi G)$ is exactly the same as $\left\{ \varrho \;\middle|\; \varrho \xleftarrow{\mathsf{R}} \mathcal{G}_0^{\mathsf{BP0}}(1^\lambda) \right\}$ when $\kappa \neq 0$ and $\xi \neq 0$, i.e., except with probability $2/q$.

If $\beta = 1$, i.e., $Y_\beta = Y_1$ $(= \psi G)$ is uniformly distributed in $\mathbb{G}$, we set $\rho := \psi - \delta - \sigma$. Then

$$\begin{aligned}
\boldsymbol{w}_1^* &= (\delta\xi G, \sigma\kappa G, (\delta + \rho + \sigma)G) = (\delta\xi, \sigma\kappa, \delta + \rho + \sigma)_{\mathbb{A}}\\
&= \delta\boldsymbol{u}_1^* + \rho\boldsymbol{u}_2^* + \sigma\boldsymbol{u}_3^* = (\delta, \rho, \sigma)_{\mathbb{U}^*}, \ \text{ and}\\
\boldsymbol{y}_1^* &= \boldsymbol{w}_1^* (W^{-1})^{\mathrm{T}} = ((\delta, \rho, \sigma)_{\mathbb{U}^*})\, (W^{-1})^{\mathrm{T}} = (\delta, \rho, \sigma)_{\mathbb{B}^*},
\end{aligned}$$

where ρ is also uniformly distributed. Therefore, the distribution of $(\mathsf{param}_{\mathsf{BP0}}, \widehat{\mathbb{B}}, \mathbb{B}^*, \boldsymbol{y}_1^*, \boldsymbol{f}, \kappa G, \xi G, \delta\xi G)$ is exactly the same as $\left\{ \varrho \;\middle|\; \varrho \xleftarrow{\mathsf{R}} \mathcal{G}_1^{\mathsf{BP0}}(1^\lambda) \right\}$ when $\kappa \neq 0$, $\xi \neq 0$ and $\rho \neq 0$, i.e., except with probability $3/q$.

Therefore, $\mathsf{Adv}_{\mathcal{D}}^{\mathsf{BP0}}(\lambda) \leq \mathsf{Adv}_{\mathcal{E}}^{\mathsf{DLIN}}(\lambda) + 2/q + 3/q = \mathsf{Adv}_{\mathcal{E}}^{\mathsf{DLIN}}(\lambda) + 5/q$. □

B.3. *Proof of Lemma 1*

Combining Lemmas 14, 15 and 16, we obtain Lemma 1.

Definition 19. (*Basic Problem 1*) Basic Problem 1 is to guess $\beta \in \{0,1\}$, given $(\mathsf{param}_{\vec{n}}, \{\mathbb{B}_t, \widehat{\mathbb{B}}^*_t\}_{t=0,\ldots,d}, \boldsymbol{f}_{\beta,0}, \{\boldsymbol{f}_{\beta,t,1}, \boldsymbol{f}_{t,i}\}_{t=1,\ldots,d;i=2,\ldots,n_t}) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{BP1}}_{\beta}(1^\lambda, \vec{n})$, where

$$
\begin{aligned}
&\mathcal{G}^{\mathsf{BP1}}_{\beta}(1^\lambda, \vec{n} := (d; n_1, \ldots, n_d)) : \quad (\mathsf{param}_{\vec{n}}, \{\mathbb{B}_t, \mathbb{B}^*_t\}_{t=0,\ldots,d}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}(1^\lambda, \vec{n}),\\
&\quad \omega, \gamma \xleftarrow{\mathsf{U}} \mathbb{F}_q, \ \tau \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times,\\
&\quad \widehat{\mathbb{B}}^*_0 := (\boldsymbol{b}^*_{0,1}, \boldsymbol{b}^*_{0,3}, .., \boldsymbol{b}^*_{0,5}), \quad \boldsymbol{f}_{0,0} := (\omega, 0, 0, 0, \gamma)_{\mathbb{B}_0}, \quad \boldsymbol{f}_{1,0} := (\omega, \tau, 0, 0, \gamma)_{\mathbb{B}_0},\\
&\quad \text{for } t = 1, \ldots, d,\\
&\qquad \vec{e}_{t,1} := (1, 0^{n_t - 1}) \in \mathbb{F}_q^{n_t}, \quad \widehat{\mathbb{B}}^*_t := (\boldsymbol{b}^*_{t,1}, \ldots, \boldsymbol{b}^*_{t,n_t}, \boldsymbol{b}^*_{t,n_t+2}, \ldots, \boldsymbol{b}^*_{t,3n_t+1}),\\
&\qquad \boldsymbol{f}_{0,t,1} := (\ \overbrace{\omega\vec{e}_{t,1},}^{n_t} \ \overbrace{0^{n_t},}^{n_t} \ \overbrace{0^{n_t},}^{n_t} \ \overbrace{\gamma}^{1}\ )_{\mathbb{B}_t},\\
&\qquad \boldsymbol{f}_{1,t,1} := (\ \omega\vec{e}_{t,1}, \quad \tau\vec{e}_{t,1}, \quad 0^{n_t}, \quad \gamma\ )_{\mathbb{B}_t},\\
&\qquad \boldsymbol{f}_{t,i} := \omega \boldsymbol{b}_{t,i} \quad i = 2, \ldots, n_t,\\
&\quad \text{return } (\mathsf{param}_{\vec{n}}, \{\mathbb{B}_t, \widehat{\mathbb{B}}^*_t\}_{t=0,\ldots,d}, \boldsymbol{f}_{\beta,0}, \{\boldsymbol{f}_{\beta,t,1}, \boldsymbol{f}_{t,i}\}_{t=1,\ldots,d;i=2,\ldots,n_t}).
\end{aligned}
$$

for $\beta \xleftarrow{\mathsf{U}} \{0,1\}$. For a probabilistic adversary $\mathcal{C}$, the advantage of $\mathcal{C}$ for Basic Problem 1, $\mathsf{Adv}^{\mathsf{BP1}}_{\mathcal{C}}(\lambda)$, is similarly defined as in Definition 4.

Lemma 15. *For any adversary $\mathcal{C}$, 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, $\mathsf{Adv}^{\mathsf{BP1}}_{\mathcal{C}}(\lambda) \le \mathsf{Adv}^{BP\hat{0}}_{\mathcal{D}}(\lambda)$ for any $\vec{n} := (d; \{n_t\}) := (d; n_1, \ldots, n_d)$.*

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

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

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

$$
\begin{aligned}
&\mathsf{param}_0 := (q, \mathbb{V}_0, \mathbb{G}_T, \mathbb{A}_0, e) := \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, 5, \mathsf{param}_{\mathbb{G}}),\\
&\mathsf{param}_t := (q, \mathbb{V}_t, \mathbb{G}_T, \mathbb{A}_t, e) := \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, 3n_t + 1, \mathsf{param}_{\mathbb{G}}) \quad \text{for } t = 1, \ldots, d,\\
&\mathsf{param}_{\vec{n}} := (\{\mathsf{param}_t\}_{t=0,\ldots,d}, g_T),
\end{aligned}
$$

where g_T is contained in $\mathsf{param}_{\mathsf{BP0}}$. $\mathcal{D}$ generates random matrices $W_t \xleftarrow{\mathsf{U}} GL(N_t, \mathbb{F}_q)$ with $N_0 := 5$, $N_t := 3n_t + 1$ for $t = 1, \ldots, d$, then sets

$$
\begin{aligned}
&\boldsymbol{d}_{0,\iota} := (\boldsymbol{b}^*_\iota, 0, 0)\, W_0 \quad \text{for } \iota = 1, 2, \quad \boldsymbol{d}_{0,3} := (0, 0, 0, \xi G, 0)\, W_0,\\
&\boldsymbol{d}_{0,4} := (\boldsymbol{b}^*_3, 0, 0)\, W_0, \quad \boldsymbol{d}_{0,5} := (0, 0, 0, 0, \xi G)\, W_0,\\
&\boldsymbol{d}^*_\iota := (\boldsymbol{b}_\iota, 0, 0)\, (W_0^{-1})^{\mathrm{T}} \quad \text{for } \iota = 1, 2, \quad \boldsymbol{d}^*_{0,3} := (0, 0, 0, \kappa G, 0)\, (W_0^{-1})^{\mathrm{T}},
\end{aligned}
$$

$$\boldsymbol{d}^*_{0,4} := (\boldsymbol{b}_3, 0, 0)\,(W_0^{-1})^{\mathrm{T}} \quad \boldsymbol{d}^*_{0,5} := (0, 0, 0, 0, \kappa G)\,(W_0^{-1})^{\mathrm{T}},$$
$$\boldsymbol{g}_{\beta,0} := (\boldsymbol{y}^*_\beta, 0, 0)\,W_0,$$
for $t = 1, \ldots, d$,
$$\boldsymbol{d}_{t,1} := (\boldsymbol{b}^*_1, 0^{N_t-3})\,W_t,\ \boldsymbol{d}_{t,n_t+1} := (\boldsymbol{b}^*_2, 0^{N_t-3})\,W_t,\ \boldsymbol{d}_{t,N_t} := (\boldsymbol{b}^*_3, 0^{N_t-3})\,W_t,$$
otherwise, $\boldsymbol{d}_{t,i} := (0^\iota, \xi G, 0^{N_t-\iota-1})\,W_t$ where $\begin{cases} \iota := i+1 \text{ if } i \in \{2, \ldots, n_t\}, \\ \iota := i \text{ if } i \in \{n_t+2, \ldots, N_t-1\}, \end{cases}$
$$\boldsymbol{d}^*_{t,1} := (\boldsymbol{b}_1, 0^{N_t-3})\,(W_t^{-1})^{\mathrm{T}},\ \boldsymbol{d}^*_{t,n_t+1} := (\boldsymbol{b}_2, 0^{N_t-3})\,(W_t^{-1})^{\mathrm{T}},\ \boldsymbol{d}^*_{t,N_t} := (\boldsymbol{b}_3, 0^{N_t-3})\,(W_t^{-1})^{\mathrm{T}},$$
otherwise, $\boldsymbol{d}^*_{t,i} := (0^\iota, \kappa G, 0^{N_t-\iota-1})\,(W_t^{-1})^{\mathrm{T}}$ where $\begin{cases} \iota := i+1 \text{ if } i \in \{2, \ldots, n_t\}, \\ \iota := i \text{ if } i \in \{n_t+2, \ldots, N_t-1\}, \end{cases}$
$$\boldsymbol{g}_{\beta,t,1} := (\boldsymbol{y}^*_\beta, 0^{N_t-3})\,W_t,\ \boldsymbol{g}_{t,i} := (0^{i+1}, \delta\xi G, 0^{N_t-i-2})\,W_t \ \text{ for } i = 2, \ldots, n_t,$$

where $(\boldsymbol{v}, 0^{N_t-3}) := (\widetilde{G}_1, \widetilde{G}_2, \widetilde{G}_3, , 0^{N_t-3})$ for any $\boldsymbol{v} := (\widetilde{G}_1, \widetilde{G}_2, \widetilde{G}_3) \in \mathbb{V} = \mathbb{G}^3$ and right multiplication by W_t (and $(W_t^{-1})^{\mathrm{T}}$) for $t \geq 0$ is given as in Remark 1 in Sect. 2.1. Then, $\mathbb{D}_0 := (\boldsymbol{d}_{0,i})_{i=1,\ldots,5}$ and $\mathbb{D}^*_0 := (\boldsymbol{d}^*_{0,i})_{i=1,\ldots,5}$, $\mathbb{D}_t := (\boldsymbol{d}_{t,i})_{i=1,\ldots,3n_t+1}$ and $\mathbb{D}^*_t := (\boldsymbol{d}^*_{t,i})_{i=1,\ldots,3n_t+1}$ are dual orthonormal bases. $\mathcal{D}$ can compute $\mathbb{D}_t$ for $t = 0, \ldots, d$, $\widehat{\mathbb{D}}^*_0 := (\boldsymbol{d}^*_{0,1}, \boldsymbol{d}^*_{0,3}, \ldots, \boldsymbol{d}^*_{0,5})$, $\widehat{\mathbb{D}}^*_t := (\boldsymbol{d}^*_{t,1}, \ldots, \boldsymbol{d}^*_{t,n_t}, \boldsymbol{d}^*_{t,n_t+2}, \ldots, \boldsymbol{d}^*_{t,3n_t+1})$ for $t = 1, \ldots, d$ from $\widehat{\mathbb{B}} := (\boldsymbol{b}_1, \boldsymbol{b}_3)$, $\mathbb{B}^*$, κG, and ξG. $\mathcal{D}$ then gives $(\mathsf{param}_{\vec{n}}, \{\mathbb{D}_t, \widehat{\mathbb{D}}^*_t\}_{t=0,\ldots,d}, \boldsymbol{g}_{\beta,0}, \{\boldsymbol{g}_{\beta,t,1}, \boldsymbol{g}_{t,i}\}_{t=1,\ldots,d;i=2,\ldots,n_t})$ to $\mathcal{C}$, and outputs $\beta' \in \{0, 1\}$ if $\mathcal{C}$ outputs β'.

We can see that

$$\boldsymbol{g}_{0,0} := (\omega', 0, 0, 0, \gamma')_{\mathbb{D}_0},\quad \boldsymbol{g}_{1,0} := (\omega', \tau', 0, 0, \gamma')_{\mathbb{D}_0},$$
for $t = 1, \ldots, d$,
$$\boldsymbol{g}_{0,t,1} := (\ \overbrace{\omega'\vec{e}_{t,1}}^{n_t},\ \overbrace{0^{n_t}}^{n_t},\ \overbrace{0^{n_t}}^{n_t},\ \overbrace{\gamma'}^{1}\ )_{\mathbb{D}_t},$$
$$\boldsymbol{g}_{1,t,1} := (\ \overbrace{\omega'\vec{e}_{t,1}}^{n_t},\ \overbrace{\tau'\vec{e}_{t,1}}^{n_t},\ \overbrace{0^{n_t}}^{n_t},\ \overbrace{\gamma'}^{1}\ )_{\mathbb{D}_t},$$
$$\boldsymbol{g}_{t,i} := \omega'\boldsymbol{d}_{t,i} \ \text{ for } i = 2, \ldots, n_t,$$

where $\omega' := \delta$, $\gamma' := \sigma$, and $\tau' := \rho$ which are distributed uniformly in $\mathbb{F}_q$. Therefore, the distribution of $(\mathsf{param}_{\vec{n}}, \{\mathbb{D}_t, \widehat{\mathbb{D}}^*_t\}_{t=0,\ldots,d}, \boldsymbol{g}_{\beta,0}, \{\boldsymbol{g}_{\beta,t,1}, \boldsymbol{g}_{t,i}\}_{t=1,\ldots,d;i=2,\ldots,n_t})$ is exactly the same as $\left\{ \varrho \;\middle|\; \varrho \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{BP1}}_\beta(1^\lambda, \vec{n}) \right\}$. □

Lemma 16. *For any adversary $\mathcal{B}$, 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, $\mathsf{Adv}^{\mathsf{P1}}_{\mathcal{B}}(\lambda) \leq \mathsf{Adv}^{BP1}_{\mathcal{C}}(\lambda) + (d+1)/q$.*

Proof. Given a Basic Problem 1 instance

$$(\mathsf{param}_{\vec{n}}, \{\mathbb{B}_t, \widehat{\mathbb{B}}^*_t\}_{t=0,\ldots,d}, \boldsymbol{f}_{\beta,0}, \{\boldsymbol{f}_{\beta,t,1}, \boldsymbol{f}_{t,i}\}_{t=1,\ldots,d;i=2,\ldots,n_t}),$$

$\mathcal{C}$ calculates

$$\boldsymbol{r}_t \xleftarrow{\mathsf{U}} \mathsf{span}\langle \boldsymbol{b}_{t,3n_t+1} \rangle,\quad \boldsymbol{e}_{\beta,t,1} := \boldsymbol{f}_{\beta,t,1} + \boldsymbol{r}_t \ \text{ for } t = 1, \ldots, d.$$

$\mathcal{C}$ generates $u_0 \xleftarrow{\cup} \mathbb{F}_q^{\times}$, $\begin{pmatrix} \vec{u}_{t,1} \\ \vdots \\ \vec{u}_{t,n_t} \end{pmatrix} := U_t \xleftarrow{\cup} GL(n_t, \mathbb{F}_q)$ for $t = 1, \ldots, d$. $\mathcal{C}$ then calculates

$$\boldsymbol{d}_{0,2} := (0, u_0, 0, 0, 0)_{\mathbb{B}_0},$$

$$\boldsymbol{d}_{t,n_t+i} := (\overbrace{\quad 0^{n_t}, \quad}^{n_t} \overbrace{\quad \vec{u}_{t,i}, \quad}^{n_t} \overbrace{\quad 0^{n_t}, \quad}^{n_t} \overbrace{\quad 0 \quad}^{1})_{\mathbb{B}_t} \text{ for } t = 1, \ldots, d;\ i = 1, \ldots, n_t,$$

$\mathcal{C}$ then sets dual orthonormal basis vectors

$$\boldsymbol{d}^*_{0,2} := (0, u_0^{-1}, 0, 0, 0)_{\mathbb{B}^*_0},$$

$$\boldsymbol{d}^*_{t,n_t+i} := (\overbrace{\quad 0^{n_t}, \quad}^{n_t} \overbrace{\quad \vec{z}_{t,i}, \quad}^{n_t} \overbrace{\quad 0^{n_t}, \quad}^{n_t} \overbrace{\quad 0 \quad}^{1})_{\mathbb{B}^*_t} \text{ for } t = 1, \ldots, d;\ i = 1 \ldots, n_t,$$

where $\begin{pmatrix} \vec{z}_{t,1} \\ \vdots \\ \vec{z}_{t,n_t} \end{pmatrix} := (U_t^{-1})^{\mathrm{T}}$. $\mathcal{C}$ cannot calculate above $\boldsymbol{d}^*_{0,2}$ and $\boldsymbol{d}^*_{t,i}$ for $i = n_t + 1, \ldots, 2n_t$ because of lack of $\boldsymbol{b}^*_{0,2}$ and $\boldsymbol{b}^*_{t,n_t+1}$. $\mathcal{C}$ then sets

$$\mathbb{D}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{d}_{0,2}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,4}, \boldsymbol{b}_{0,5}),\ \mathbb{D}^*_0 := (\boldsymbol{b}^*_{0,1}, \boldsymbol{d}^*_{0,2}, \boldsymbol{b}^*_{0,3}, \boldsymbol{b}^*_{0,4}, \boldsymbol{b}^*_{0,5}),\ \widehat{\mathbb{D}}^*_0 := (\boldsymbol{b}^*_{0,1}, \boldsymbol{b}^*_{0,3}, \boldsymbol{b}^*_{0,4}, \boldsymbol{b}^*_{0,5}),$$
$$\mathbb{D}_t := (\boldsymbol{b}_{t,1}, \ldots, \boldsymbol{b}_{t,n_t}, \boldsymbol{d}_{t,n_t+1}, \ldots, \boldsymbol{d}_{t,2n_t}, \boldsymbol{b}_{t,2n_t+1}, \ldots, \boldsymbol{b}_{t,3n_t+1}),$$
$$\mathbb{D}^*_t := (\boldsymbol{b}^*_{t,1}, \ldots, \boldsymbol{b}^*_{t,n_t}, \boldsymbol{d}^*_{t,n_t+1}, \ldots, \boldsymbol{d}^*_{t,2n_t}, \boldsymbol{b}^*_{t,2n_t+1}, \ldots, \boldsymbol{b}^*_{t,3n_t+1})$$
$$\widehat{\mathbb{D}}^*_t := (\boldsymbol{b}^*_{t,1}, \ldots, \boldsymbol{b}^*_{t,n_t}, \boldsymbol{b}^*_{t,2n_t+1}, \ldots, \boldsymbol{b}^*_{t,3n_t+1}).$$

$\mathcal{C}$ gives $(\mathsf{param}_{\vec{n}}, \{\mathbb{D}_t, \widehat{\mathbb{D}}^*_t\}_{t=0,\ldots,d}, \boldsymbol{f}_{\beta,0}, \{\boldsymbol{e}_{\beta,t,1}, \boldsymbol{f}_{t,i}\}_{t=1,\ldots,d;i=2,\ldots,n_t})$ to $\mathcal{B}$, and outputs $\beta' \in \{0,1\}$ if $\mathcal{B}$ outputs β'.

Then, with respect to $\mathbb{D}_t, \mathbb{D}^*_t$ (instead of $\mathbb{B}_t, \mathbb{B}^*_t$, respectively), 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 $\mathcal{G}^{\mathsf{P1}}_{\beta}(1^{\lambda}, \vec{n})$ if z_0 in Problem 1 is not equal to 0 and $(z_{t,1}, \ldots, z_{t,n_t})$ in Problem 1 is not equal to $\vec{0}$ for any $t = 1, \ldots, d$, i.e., except with probability $(d+1)/q$ for $\beta = 1$. □

B.4. *Proof of Lemma 2*

Combining Lemmas 14, 17 and 18 , we obtain Lemma 2.

Definition 20. (*Basic Problem 2*) Basic Problem 2 is to guess $\beta \in \{0,1\}$, given $(\mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}_t, \mathbb{B}^*_t\}_{t=0,\ldots,d}, \boldsymbol{y}^*_{\beta,0}, \boldsymbol{f}_0, \{\boldsymbol{y}^*_{\beta,t,i}, \boldsymbol{f}_{t,i}\}_{t=1,\ldots,d;i=1,\ldots,n_t}) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{BP2}}_{\beta}(1^{\lambda}, \vec{n})$, where

$$\mathcal{G}^{\mathsf{BP2}}_{\beta}(1^{\lambda}, \vec{n} := (d; n_1, \ldots, n_d)) : \quad (\mathsf{param}_{\vec{n}}, \{\mathbb{B}_t, \mathbb{B}^*_t\}_{t=0,\ldots,d}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}(1^{\lambda}, \vec{n}),$$

$$\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \ldots, \boldsymbol{b}_{0,5}),$$
$$\widehat{\mathbb{B}}_t := (\boldsymbol{b}_{t,1}, \ldots, \boldsymbol{b}_{t,n_t}, \boldsymbol{b}_{t,2n_t+1}, \ldots, \boldsymbol{b}_{t,3n_t+1}) \text{ for } t = 1, \ldots, d,$$
$$\delta, \delta_0, \omega \xleftarrow{\cup} \mathbb{F}_q, \quad \rho, \tau \xleftarrow{\cup} \mathbb{F}_q^{\times},$$
$$\boldsymbol{y}^*_{0,0} := (\delta, 0, 0, \delta_0, 0)_{\mathbb{B}^*_0}, \quad \boldsymbol{y}^*_{1,0} := (\delta, \rho, 0, \delta_0, 0)_{\mathbb{B}^*_0}, \quad \boldsymbol{f}_0 := (\omega, \tau, 0, 0, 0)_{\mathbb{B}_0},$$
$$\text{for } t = 1, \ldots, d, \quad i = 1, \ldots, n_t;$$
$$\vec{e}_{t,i} := (0^{i-1}, 1, 0^{n_t - i}) \in \mathbb{F}_q^{n_t},$$
$$\boldsymbol{y}^*_{0,t,i} := (\overbrace{\delta\vec{e}_{t,i},}^{n_t} \overbrace{0^{n_t},}^{n_t} \overbrace{\delta_0\vec{e}_{t,i},}^{n_t} \overbrace{0}^{1})_{\mathbb{B}^*_t}$$
$$\boldsymbol{y}^*_{1,t,i} := (\delta\vec{e}_{t,i}, \rho\vec{e}_{t,i}, \delta_0\vec{e}_{t,i}, 0)_{\mathbb{B}^*_t}$$
$$\boldsymbol{f}_{t,i} := (\omega\vec{e}_{t,i}, \tau\vec{e}_{t,i}, 0^{n_t}, 0)_{\mathbb{B}_t},$$
$$\text{return } (\mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}_t, \mathbb{B}^*_t\}_{t=0,..,d}, \boldsymbol{y}^*_{\beta,0}, \boldsymbol{f}_0, \{\boldsymbol{y}^*_{\beta,t,i}, \boldsymbol{f}_{t,i}\}_{t=1,..,d;i=1,..,n_t}).$$

for $\beta \xleftarrow{\cup} \{0, 1\}$. 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 4.

Lemma 17. *For any adversary $\mathcal{C}$, 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, $\mathsf{Adv}^{\mathsf{BP2}}_{\mathcal{C}}(\lambda) = \mathsf{Adv}^{BP0}_{\mathcal{D}}(\lambda)$ for any $\vec{n} := (d; \{n_t\}) := (d; n_1, \ldots, n_d)$.*

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

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

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

$$\mathsf{param}_0 := (q, \mathbb{V}_0, \mathbb{G}_T, \mathbb{A}_0, e) := \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, 5, \mathsf{param}_{\mathbb{G}}),$$
$$\mathsf{param}_t := (q, \mathbb{V}_t, \mathbb{G}_T, \mathbb{A}_t, e) := \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, 3n_t + 1, \mathsf{param}_{\mathbb{G}}) \quad \text{for } t = 1, \ldots, d,$$
$$\mathsf{param}_{\vec{n}} := (\{\mathsf{param}_t\}_{t=0,\ldots,d}, g_T),$$

where g_T is contained in $\mathsf{param}_{\mathsf{BP0}}$. $\mathcal{D}$ generates random matrices $W_t \xleftarrow{\cup} GL(N_t, \mathbb{F}_q)$ with $N_0 := 5$, $N_t := 3n_t + 1$ for $t = 1, \ldots, d$, then sets

$$\boldsymbol{d}_{0,\iota} := (\boldsymbol{b}_\iota, 0, 0)\, W_0 \quad \text{for } \iota = 1, 2, \quad \boldsymbol{d}_{0,3} := (0, 0, 0, \kappa G, 0)\, W_0,$$
$$\boldsymbol{d}_{0,4} := (\boldsymbol{b}_3, 0, 0)\, W_0, \quad \boldsymbol{d}_{0,5} := (0, 0, 0, 0, \kappa G)\, W_0,$$
$$\boldsymbol{d}^*_\iota := (\boldsymbol{b}^*_\iota, 0, 0)\, (W_0^{-1})^{\mathrm{T}} \quad \text{for } \iota = 1, 2, \quad \boldsymbol{d}^*_{0,3} := (0, 0, 0, \xi G, 0)\, (W_0^{-1})^{\mathrm{T}},$$
$$\boldsymbol{d}^*_{0,4} := (\boldsymbol{b}^*_3, 0, 0)\, (W_0^{-1})^{\mathrm{T}} \quad \boldsymbol{d}^*_{0,5} := (0, 0, 0, 0, \xi G)\, (W_0^{-1})^{\mathrm{T}},$$
$$\boldsymbol{p}^*_{\beta,0} := (\boldsymbol{y}^*_\beta, 0, 0)\, (W_0^{-1})^{\mathrm{T}}, \qquad \boldsymbol{g}_0 := (\boldsymbol{f}, 0, 0)\, W_0,$$
$$\text{for } t = 1, \ldots, d,$$

$$
\begin{aligned}
\boldsymbol{d}_{t,(\iota-1)n_t+i} &:= (0^{3(i-1)}, \boldsymbol{b}_\iota, 0^{3(n_t-i)}, 0)\, W_t \quad \text{for } \iota = 1,2,3;\ i = 1,\ldots,n_t,\\
\boldsymbol{d}_{t,3n_t+1} &:= (0^{3n_t}, \kappa G)\, W_t,\\
\boldsymbol{d}^*_{t,(\iota-1)n_t+i} &:= (0^{3(i-1)}, \boldsymbol{b}^*_\iota, 0^{3(n_t-i)}, 0)\, (W_t^{-1})^{\mathrm{T}} \quad \text{for } \iota = 1,2,3;\ i = 1,\ldots,n_t,\\
\boldsymbol{d}^*_{t,3n_t+1} &:= (0^{3n_t}, \xi G)\, (W_t^{-1})^{\mathrm{T}},\\
\boldsymbol{p}^*_{\beta,t,i} &:= (0^{3(i-1)}, \boldsymbol{y}^*_\beta, 0^{3(n_t-i)}, 0)\, (W_t^{-1})^{\mathrm{T}} \quad \text{for } i = 1,\ldots,n_t,\\
\boldsymbol{g}_{t,i} &:= (0^{3(i-1)}, \boldsymbol{f}, 0^{3(n_t-i)}, 0)\, W_t \quad \text{for } i = 1,\ldots,n_t.
\end{aligned}
$$

where $(0^{l_1}, \boldsymbol{v}, 0^{l_2}) := (0^{l_1}, \widetilde{G}_1, \widetilde{G}_2, \widetilde{G}_3, 0^{l_2})$ for any $\boldsymbol{v} := (\widetilde{G}_1, \widetilde{G}_2, \widetilde{G}_3) \in \mathbb{V} = \mathbb{G}^3$ and $l_1, l_2 \in \mathbb{Z}_{\geq 0}$, and right multiplication by W_t (and $(W_t^{-1})^{\mathrm{T}}$) for $t \geq 0$ is given as in Remark 1 in Sect. 2.1. Then, $\mathbb{D}_0 := (\boldsymbol{d}_{0,i})_{i=1,\ldots,5}$ and $\mathbb{D}^*_0 := (\boldsymbol{d}^*_{0,i})_{i=1,\ldots,5}$, $\mathbb{D}_t := (\boldsymbol{d}_{t,i})_{i=1,\ldots,3n_t+1}$ and $\mathbb{D}^*_t := (\boldsymbol{d}^*_{t,i})_{i=1,\ldots,3n_t+1}$ for $t = 1,\ldots,d$ are dual orthonormal bases. $\mathcal{D}$ can compute

$$
\begin{aligned}
&\widehat{\mathbb{D}}_0 := (\boldsymbol{d}_{0,1}, \boldsymbol{d}_{0,3}, \ldots, \boldsymbol{d}_{0,5}), \quad \mathbb{D}^*_0 := (\boldsymbol{d}^*_{0,1}, \ldots, \boldsymbol{d}^*_{0,5}),\\
&\text{for } t = 1,\ldots,d, \quad \widehat{\mathbb{D}}_t := (\boldsymbol{d}_{t,1}, \ldots, \boldsymbol{d}_{t,n_t}, \boldsymbol{d}_{t,2n_t+1}, \ldots, \boldsymbol{d}_{t,3n_t+1}), \mathbb{D}^*_t := (\boldsymbol{d}^*_{t,1}, \ldots, \boldsymbol{d}^*_{t,3n_t+1}),
\end{aligned}
$$

from $\widehat{\mathbb{B}} := (\boldsymbol{b}_1, \boldsymbol{b}_3)$, $\mathbb{B}^*$, κG, and ξG. $\mathcal{D}$ then gives $(\mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{D}}_t, \mathbb{D}^*_t\}_{t=0,\ldots,d}, \boldsymbol{p}^*_{\beta,0}, \boldsymbol{g}_0, \{\boldsymbol{p}^*_{\beta,t,i}, \boldsymbol{g}_{t,i}\}_{t=1,\ldots,d;i=1,\ldots,n_t})$ to $\mathcal{C}$, and outputs $\beta' \in \{0,1\}$ if $\mathcal{C}$ outputs β'.

We can see that

$$
\boldsymbol{p}^*_{0,0} = (\delta, 0, 0, \delta_0, 0)_{\mathbb{D}^*_0}, \quad \boldsymbol{p}^*_{1,0} = (\delta, \rho, 0, \delta_0, 0)_{\mathbb{D}^*_0}, \quad \boldsymbol{g}_0 = (\omega, \tau, 0, 0, 0)_{\mathbb{D}_0},
$$

$$
\begin{aligned}
\boldsymbol{p}^*_{0,t,i} &= (\ \overbrace{\delta\vec{e}_{t,i},}^{n_t}\ \overbrace{0^{n_t},}^{n_t}\ \overbrace{\delta_0\vec{e}_{t,i},}^{n_t}\ \overbrace{0}^{1}\ )_{\mathbb{D}^*_t}\\
\boldsymbol{p}^*_{1,t,i} &= (\ \delta\vec{e}_{t,i},\ \rho\vec{e}_{t,i},\ \delta_0\vec{e}_{t,i},\ 0\ )_{\mathbb{D}^*_t}\\
\boldsymbol{g}_{t,i} &= (\ \omega\vec{e}_{t,i},\ \tau\vec{e}_{t,i},\ 0^{n_t},\ 0\ )_{\mathbb{D}_t},\\
&\qquad t = 1,\ldots,d; i = 1,\ldots,n_t,
\end{aligned}
$$

Therefore, the distribution of $(\mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{D}}_t, \mathbb{D}^*_t\}_{t=0,\ldots,d}, \boldsymbol{p}^*_{\beta,0}, \boldsymbol{g}_0, \{\boldsymbol{p}^*_{\beta,t,i}, \boldsymbol{g}_{t,i}\}_{t=1,\ldots,d;i=1,\ldots,n_t})$ is exactly the same as $\left\{ \varrho \;\middle|\; \varrho \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{BP2}}_\beta(1^\lambda, (d, \{n_t\})) \right\}$. □

Lemma 18. *For any adversary $\mathcal{B}$, 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, $\mathsf{Adv}^{\mathsf{P2}}_{\mathcal{B}}(\lambda) = \mathsf{Adv}^{BP2}_{\mathcal{C}}(\lambda)$.*

Proof. Given a Basic Problem 2 instance

$$
(\mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}_t, \mathbb{B}^*_t\}_{t=0,..,d}, \boldsymbol{y}^*_{\beta,0}, \boldsymbol{f}_0, \{\boldsymbol{y}^*_{\beta,t,i}, \boldsymbol{f}_{t,i}\}_{t=1,..,d;i=1,..,n_t}),
$$

$\mathcal{C}$ calculates

$$
\boldsymbol{r}^*_{t,i} \xleftarrow{\mathsf{U}} \mathsf{span}\langle \boldsymbol{b}^*_{t,2n_t+1}, \ldots, \boldsymbol{b}^*_{t,3n_t} \rangle, \quad \boldsymbol{h}^*_{\beta,t,i} := \boldsymbol{y}^*_{\beta,t,i} + \boldsymbol{r}^*_{t,i}.
$$

$\mathcal{C}$ then generates $z'_0 \xleftarrow{\cup} \mathbb{F}_q^\times$ and $\begin{pmatrix} \vec{z}'_{t,1} \\ \vdots \\ \vec{z}'_{t,n_t} \end{pmatrix} := Z'_t \xleftarrow{\cup} GL(n_t, \mathbb{F}_q)$, for $t = 1, \ldots, d$, and calculates

$$\boldsymbol{d}^*_{0,2} := (0, z'_0, 0, 0, 0)_{\mathbb{B}^*_0},$$
$$\boldsymbol{d}^*_{t,n_t+i} := (\overbrace{0^{n_t}}^{n_t}, \overbrace{\vec{z}'_{t,i}}^{n_t}, \overbrace{0^{n_t}}^{n_t}, \overbrace{0}^{1})_{\mathbb{B}^*_t} \text{ for } t = 1, \ldots, d;\ i = 1 \ldots, n_t.$$

$\mathcal{C}$ then sets $z_0 := \rho^{-1} z'_0, u_0 := z_0^{-1}$, $\begin{pmatrix} \vec{z}_{t,1} \\ \vdots \\ \vec{z}_{t,n_t} \end{pmatrix} := Z_t := \rho^{-1} Z'_t$ and $\begin{pmatrix} \vec{u}_{t,1} \\ \vdots \\ \vec{u}_{t,n_t} \end{pmatrix} := (Z_t^{-1})^{\mathrm{T}}$, where ρ is defined in Basic Problem 2. Then,

$$\boldsymbol{d}^*_{0,2} = (0, \rho z_0, 0, 0, 0)_{\mathbb{B}^*_0},$$
$$\boldsymbol{d}^*_{t,n_t+i} = (\overbrace{0^{n_t}}^{n_t}, \overbrace{\rho\vec{z}_{t,i}}^{n_t}, \overbrace{0^{n_t}}^{n_t}, \overbrace{0}^{1})_{\mathbb{B}^*_t} \text{ for } t = 1, \ldots, d;\ i = 1 \ldots, n_t.$$

$\mathcal{C}$ then sets dual orthonormal basis vectors

$$\boldsymbol{d}_{0,2} := (0, \rho^{-1} u_0, 0, 0, 0)_{\mathbb{B}_0},$$
$$\boldsymbol{d}_{t,n_t+i} := (\overbrace{0^{n_t}}^{n_t}, \overbrace{\rho^{-1}\vec{u}_{t,i}}^{n_t}, \overbrace{0^{n_t}}^{n_t}, \overbrace{0}^{1})_{\mathbb{B}_t} \text{ for } t = 1, \ldots, d;\ i = 1 \ldots, n_t.$$

$\mathcal{C}$ cannot calculate above $\boldsymbol{d}_{0,2}$ and $\boldsymbol{d}_{t,i}$ for $i = n_t + 1, \ldots, 2n_t$. $\mathcal{C}$ then sets

$$\mathbb{D}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{d}_{0,2}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,4}, \boldsymbol{b}_{0,5}),\ \widehat{\mathbb{D}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,4}, \boldsymbol{b}_{0,5}), \mathbb{D}^*_0 := (\boldsymbol{b}^*_{0,1}, \boldsymbol{d}^*_{0,2}, \boldsymbol{b}^*_{0,3}, \boldsymbol{b}^*_{0,4}, \boldsymbol{b}^*_{0,5}),$$
$$\mathbb{D}_t := (\boldsymbol{b}_{t,1}, \ldots, \boldsymbol{b}_{t,n_t}, \boldsymbol{d}_{t,n_t+1}, \ldots, \boldsymbol{d}_{t,2n_t}, \boldsymbol{b}_{t,2n_t+1}, \ldots, \boldsymbol{b}_{t,3n_t+1}),$$
$$\widehat{\mathbb{D}}_t := (\boldsymbol{b}_{t,1}, \ldots, \boldsymbol{b}_{t,n_t}, \boldsymbol{b}_{t,2n_t+1}, \ldots, \boldsymbol{b}_{t,3n_t+1}),$$
$$\mathbb{D}^*_t := (\boldsymbol{b}^*_{t,1}, \ldots, \boldsymbol{b}^*_{t,n_t}, \boldsymbol{d}^*_{t,n_t+1}, \ldots, \boldsymbol{d}^*_{t,2n_t}, \boldsymbol{b}^*_{t,2n_t+1}, \ldots, \boldsymbol{b}^*_{t,3n_t+1}).$$

$\mathcal{C}$ gives $(\mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{D}}_t, \mathbb{D}^*_t\}_{t=0,\ldots,d}, \boldsymbol{y}^*_{\beta,0}, \boldsymbol{f}_0, \{\boldsymbol{h}^*_{\beta,t,i}, \boldsymbol{f}_{t,i}\}_{t=1,\ldots,d;i=1,\ldots,n_t})$ to $\mathcal{B}$, and outputs $\beta' \in \{0, 1\}$ if $\mathcal{B}$ outputs β'.

For τ in Basic Problem 2, let $\tau' := \rho\tau$. Then, with respect to τ', $\mathbb{D}_t$, $\mathbb{D}^*_t$ (instead of τ, $\mathbb{B}_t$, $\mathbb{B}^*_t$), 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 $\mathcal{G}^{\mathsf{P2}}_\beta(1^\lambda, \vec{n})$. □

C. Proof of Main Lemma (Lemma 3)

Proof. We first remind the definition of cofactor (and cofactor matrix). When $n \geq 2$, for $n \times n$ matrix $Z := (z_{i,j})$, let $\Delta_{i,j}$ the minor obtained by removing the ith row and the jth column from Z. Cofactors $\widetilde{z}_{i,j}$ are defined by $(-1)^{i+j}\Delta_{i,j}$. The determinant of Z is given as $\det Z = \sum_{j=1}^{n} z_{i,j}\widetilde{z}_{i,j}$. In particular, when $i = 1$, we obtain

$$\det Z = \sum_{j=1}^{n} z_{1,j}\widetilde{z}_{1,j}. \tag{17}$$

In addition, when $\det Z \neq 0$, we have

$$U := (Z^{-1})^{\mathrm{T}} = \frac{1}{\det Z}\begin{pmatrix} \widetilde{z}_{1,1} & \cdots & \widetilde{z}_{1,n} \\ \vdots & & \vdots \\ \widetilde{z}_{n,1} & \cdots & \widetilde{z}_{n,n} \end{pmatrix}. \tag{18}$$

Below, we denote row vectors $\vec{z}_i := (z_{i,1}, \ldots, z_{i,n})$ of Z and $\vec{\widetilde{z}}_i := (\widetilde{z}_{i,1}, \ldots, \widetilde{z}_{i,n})$ of $(\det Z) \cdot (Z^{-1})^{\mathrm{T}}$ for $i = 1, \ldots, n$.

Case that $\vec{x} \cdot \vec{v} = p \neq 0$: For normalized pair of vectors

$$\vec{x} := (p, 0, \ldots, 0), \quad \vec{v} := (1, 0, \ldots, 0), \tag{19}$$

we will show that $(\vec{x}U, \vec{v}Z)$ is uniformly distributed on C_p for $Z \xleftarrow{\cup} GL(n, \mathbb{F}_q)$, $U := (Z^{-1})^{\mathrm{T}}$. By that, for any pair $(\vec{x}, \vec{v}) \in C_p$, we see that $(\vec{x}U, \vec{v}Z)$ is uniformly distributed on C_p for $Z \xleftarrow{\cup} GL(n, \mathbb{F}_q)$, $U := (Z^{-1})^{\mathrm{T}}$. Therefore, we consider $(\vec{x}, \vec{v})$ given by Eq. (19) in the following.

Since $Z = (z_{i,j})$ and Eq. (18) holds, we have

$$\vec{x}U = \frac{p \cdot \vec{\widetilde{z}}_1}{\det Z} = \frac{p}{\det Z}(\widetilde{z}_{1,1}, \ldots, \widetilde{z}_{1,n}), \quad \vec{v}Z = \vec{z}_1 = (z_{1,1}, \ldots, z_{1,n}). \tag{20}$$

(Eqs. (17) and (20) give that $(\vec{x}U) \cdot (\vec{v}Z) = p$.) Cofactors $\widetilde{z}_{1,j}$ are determined by $n - 1$ row vectors, $\vec{z}_2, \ldots, \vec{z}_n$ of Z. That is, $\vec{\widetilde{z}}_1$ is orthogonal to hyperplane $\mathsf{span}\langle \vec{z}_2, \ldots, \vec{z}_n \rangle$ which is uniformly distributed in the spaces of hyperplanes H with the condition $\vec{z}_1 \notin H$ when $Z \xleftarrow{\cup} GL(n, \mathbb{F}_q)$ with fixed (random) $\vec{z}_1$. Hence, from one-to-one correspondence of hyperplanes and their normal vectors (up to scalars), we see that $(\vec{x}U, \vec{v}Z)$ is uniformly distributed in C_p when $Z \xleftarrow{\cup} GL(n, \mathbb{F}_q)$.

Case that $\vec{x} \cdot \vec{v} = 0$: For normalized pair

$$\vec{x} := (0, 1, \ldots, 0), \quad \vec{v} := (1, 0, \ldots, 0),$$

we will show that $(\vec{x}U, \vec{v}Z)$ is uniformly distributed on C_0 for $Z \xleftarrow{\mathsf{U}} GL(n, \mathbb{F}_q)$, $U := (Z^{-1})^{\mathrm{T}}$ because of the similar reason as above.

Since $Z = (z_{i,j})$ and Eq. (18) holds, we have

$$\vec{x}U = \frac{\tilde{\vec{z}}_2}{\det Z} = \frac{1}{\det Z}(\tilde{z}_{2,1}, \ldots, \tilde{z}_{2,n}), \quad \vec{v}Z = \vec{z}_1 = (z_{1,1}, \ldots, z_{1,n}).$$

Cofactors $\tilde{z}_{2,j}$ are determined by $n-1$ row vectors, $\vec{z}_1, \vec{z}_3, \ldots, \vec{z}_n$ of Z. In particular, $\vec{z}_2$ is related to only term $\det Z$ in $(\vec{x}U, \vec{v}Z)$.

First, since $\tilde{\vec{z}}_2$ is orthogonal to hyperplane $\mathsf{span}\langle \vec{z}_1, \vec{z}_3, \ldots, \vec{z}_n \rangle$ which is uniformly distributed in the spaces of hyperplanes H with the condition $\vec{z}_1 \in H$ when $Z \xleftarrow{\mathsf{U}} GL(n, \mathbb{F}_q)$ with fixed (random) $\vec{z}_1$, we see that $\tilde{\vec{z}}_2$ is distributed uniformly on the orthogonal space to $\vec{z}_1 (= \vec{v}Z)$ up to scalar multiplication. Moreover, since $\det Z$ is uniformly distributed in $\mathbb{F}_q^{\times}$ when $\vec{z}_2$ is uniformly distributed in $\mathbb{F}_q^n \backslash \{\vec{0}\}$ such that $\det Z \neq 0$ and $\vec{z}_2$ is related to only $\det Z$ in $(\vec{x}U, \vec{v}Z)$, we see that $\vec{x}U = \frac{\tilde{\vec{z}}_2}{\det Z}$ is distributed uniformly on the space orthogonal to $\vec{z}_1$, i.e., $(\vec{x}U, \vec{v}Z)$ is uniformly distributed on C_0 when $Z \xleftarrow{\mathsf{U}} GL(n, \mathbb{F}_q)$. □

D. Problems 3, 4 and 5 for CCA-Secure CP-FE

We will show Problems 3–5 and Lemmas 19–21 for the proof of Theorem 4. The proofs of Lemmas 19–21 are similar to those of Lemmas 1 and 2

Definition 21. (*Problem 3*) Problem 3 is to guess $\beta \in \{0, 1\}$, given $(\mathsf{param}_{\vec{n}}, \mathbb{B}_0, \widehat{\mathbb{B}}_0^*, \boldsymbol{e}_{\beta,0}, \{\mathbb{B}_t, \widehat{\mathbb{B}}_t^*, \boldsymbol{e}_{\beta,t,1}, \boldsymbol{e}_{t,i}\}_{t=1,\ldots,d+1;i=2,\ldots,n_t}) \xleftarrow{\mathsf{R}} \mathcal{G}_\beta^{\mathsf{P3}}(1^\lambda, \vec{n})$, where

$$\begin{aligned}
&\mathcal{G}_\beta^{\mathsf{P3}}(1^\lambda, \vec{n}): \quad n_{d+1} := 2, \quad \vec{n}' := (d+1; \{n_t\}_{t=1,\ldots,d+1}),\\
&(\mathsf{param}_{\vec{n}'}, \mathbb{B}_0, \widehat{\mathbb{B}}_0^*, \boldsymbol{e}_{\beta,0}, \{\mathbb{B}_t, \widehat{\mathbb{B}}_t^*, \boldsymbol{e}_{\beta,t,1}, \boldsymbol{e}_{t,i}\}_{t=1,\ldots,d+1;i=2,\ldots,n_t}) \xleftarrow{\mathsf{R}} \mathcal{G}_\beta^{\mathsf{P1}}(1^\lambda, \vec{n}'),\\
&\text{return } (\mathsf{param}_{\vec{n}}, \mathbb{B}_0, \widehat{\mathbb{B}}_0^*, \boldsymbol{e}_{\beta,0}, \{\mathbb{B}_t, \widehat{\mathbb{B}}_t^*, \boldsymbol{e}_{\beta,t,1}, \boldsymbol{e}_{t,i}\}_{t=1,\ldots,d+1;i=2,\ldots,n_t}).
\end{aligned}$$

for $\beta \xleftarrow{\mathsf{U}} \{0, 1\}$. For a probabilistic machine $\mathcal{B}$, the advantage of $\mathcal{B}$ for Problem 3, $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{P3}}(\lambda)$, is similarly defined as in Definition 4.

Lemma 19. *For any adversary $\mathcal{B}$, there exist probabilistic machine $\mathcal{E}$, whose running times are essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{P3}}(\lambda) \leq \mathsf{Adv}_{\mathcal{E}}^{DLIN}(\lambda) + (d+7)/q$.*

Definition 22. (*Problem 4*) Problem 4 is to guess $\beta \in \{0, 1\}$, given $(\mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}_t, \mathbb{B}_t^*\}_{t=0,..,d}, \quad \mathbb{B}_{d+1}, \mathbb{B}_{d+1}^*, \boldsymbol{h}*_{\beta,0}, \boldsymbol{e}_0, \{\boldsymbol{h}*_{\beta,t,i}, \boldsymbol{e}_{t,i}\}_{t=1,..,d;i=1,..,n_t}, \{\boldsymbol{h}*_{d+1,i}\}_{i=1,2})$ $\xleftarrow{\mathsf{R}} \mathcal{G}_\beta^{\mathsf{P4}}(1^\lambda, \vec{n})$, where

$\mathcal{G}_{\beta}^{\mathsf{P4}}(1^{\lambda}, \vec{n})$: $n_{d+1} := 2$, $\vec{n}' := (d+1; \{n_t\}_{t=1,\ldots,d+1})$,

$(\mathsf{param}_{\vec{n}'}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0,\ldots,d+1}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}(1^{\lambda}, \vec{n}')$,

$\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \ldots, \boldsymbol{b}_{0,5})$,

$\widehat{\mathbb{B}}_t := (\boldsymbol{b}_{t,1}, \ldots, \boldsymbol{b}_{t,n_t}, \boldsymbol{b}_{t,2n_t+1}, \ldots, \boldsymbol{b}_{t,3n_t+1})$ for $t = 1, \ldots, d$, $\delta, \delta_0, \omega \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $u_0, \tau \xleftarrow{\mathsf{U}} \mathbb{F}_q^{\times}$, $z_0 := u_0^{-1}$,

$\begin{pmatrix} \vec{z}_{t,1} \\ \vdots \\ \vec{z}_{t,n_t} \end{pmatrix} := Z_t \xleftarrow{\mathsf{U}} GL(n_t, \mathbb{F}_q)$, $\begin{pmatrix} \vec{u}_{t,1} \\ \vdots \\ \vec{u}_{t,n_t} \end{pmatrix} := (Z_t^{-1})^{\mathrm{T}}$ for $t = 1, \ldots, d$,

$\boldsymbol{h}_{0,0}^* := (\delta, 0, 0, \delta_0, 0)_{\mathbb{B}_0^*}$, $\boldsymbol{h}_{1,0}^* := (\delta, u_0, 0, \delta_0, 0)_{\mathbb{B}_0^*}$, $\boldsymbol{e}_0 := (\omega, \tau z_0, 0, 0, 0)_{\mathbb{B}_0}$,

for $t = 1, \ldots, d$; $i = 1, \ldots, n_t$;

$\vec{e}_{t,i} := (0^{i-1}, 1, 0^{n_t-i}) \in \mathbb{F}_q^{n_t}$, $\vec{\delta}_{t,i} \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_t}$,

$$\begin{aligned}
\boldsymbol{h}_{0,t,i}^* &:= (\overbrace{\delta\vec{e}_{t,i},}^{n_t} \overbrace{0^{n_t},}^{n_t} \overbrace{\vec{\delta}_{t,i},}^{n_t} \overbrace{0}^{1})_{\mathbb{B}_t^*}, \\
\boldsymbol{h}_{1,t,i}^* &:= (\delta\vec{e}_{t,i}, \vec{u}_{t,i}, \vec{\delta}_{t,i}, 0)_{\mathbb{B}_t^*}, \\
\boldsymbol{e}_{t,i} &:= (\omega\vec{e}_{t,i}, \tau\vec{z}_{t,i}, 0^{n_t}, 0)_{\mathbb{B}_t}
\end{aligned}$$

$\boldsymbol{h}_{d+1,i}^* := \delta \boldsymbol{b}_{d+1,i}^*$ for $i = 1, 2$,

return $(\mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}_t, \mathbb{B}_t^*\}_{t=0,\ldots,d}, \mathbb{B}_{d+1}, \mathbb{B}_{d+1}^*, \boldsymbol{h}_{\beta,0}^*, \boldsymbol{e}_0, \{\boldsymbol{h}_{\beta,t,i}^*, \boldsymbol{e}_{t,i}\}_{t=1,\ldots,d;i=1,\ldots,n_t}, \{\boldsymbol{h}_{d+1,i}^*\}_{i=1,2})$.

for $\beta \xleftarrow{\mathsf{U}} \{0, 1\}$. For a probabilistic machine $\mathcal{B}$, the advantage of $\mathcal{B}$ for Problem 4, $\mathsf{Adv}_{\mathcal{B}}^{\mathsf{P4}}(\lambda)$, is similarly defined as in Definition 4.

Lemma 20. *For any adversary $\mathcal{B}$, 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, $\mathsf{Adv}_{\mathcal{B}}^{P4}(\lambda) \leq \mathsf{Adv}_{\mathcal{E}}^{DLIN}(\lambda) + 5/q$.*

Definition 23. (*Problem 5*) Problem 5 is to guess $\beta \in \{0, 1\}$, given $(\mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}_t, \mathbb{B}_t^*\}_{t=0,d+1}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=1,..,d}, \boldsymbol{h}_{\beta,0}^*, \boldsymbol{e}_0, \{\boldsymbol{h}_{t,i}^*\}_{t=1,..,d;i=1,..,n_t}, \{\boldsymbol{h}_{\beta,d+1,i}^*, \boldsymbol{e}_{d+1,i}\}_{i=1,2})$ $\xleftarrow{\mathsf{R}} \mathcal{G}_{\beta}^{\mathsf{P5}}(1^{\lambda}, \vec{n})$, where

$\mathcal{G}_{\beta}^{\mathsf{P5}}(1^{\lambda}, \vec{n})$: $n_{d+1} := 2$, $\vec{n}' := (d+1; \{n_t\}_{t=1,\ldots,d+1})$,

$(\mathsf{param}_{\vec{n}'}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0,\ldots,d+1}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}(1^{\lambda}, \vec{n}')$,

$\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \ldots, \boldsymbol{b}_{0,5})$, $\widehat{\mathbb{B}}_{d+1} := (\boldsymbol{b}_{d+1,1}, \boldsymbol{b}_{d+1,2}, \boldsymbol{b}_{d+1,5}, \ldots, \boldsymbol{b}_{d+1,7})$,

$\delta, \delta_0, \omega \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $u_0, \tau \xleftarrow{\mathsf{U}} \mathbb{F}_q^{\times}$, $z_0 := u_0^{-1}$,

$\boldsymbol{h}_{0,0}^* := (\delta, 0, 0, \delta_0, 0)_{\mathbb{B}_0^*}$, $\boldsymbol{h}_{1,0}^* := (\delta, u_0, 0, \delta_0, 0)_{\mathbb{B}_0^*}$, $\boldsymbol{e}_0 := (\omega, \tau z_0, 0, 0, 0)_{\mathbb{B}_0}$,

$\boldsymbol{h}_{t,i}^* := \delta \boldsymbol{b}_{t,i}^*$ for $t = 1, \ldots, d$; $i = 1, \ldots, n_t$,

$\begin{pmatrix} \vec{z}_{d+1,1} \\ \vec{z}_{d+1,2} \end{pmatrix} := Z_{d+1} \xleftarrow{\mathsf{U}} GL(2, \mathbb{F}_q)$, $\begin{pmatrix} \vec{u}_{d+1,1} \\ \vec{u}_{d+1,2} \end{pmatrix} := (Z_{d+1}^{-1})^{\mathrm{T}}$,

for $i = 1, 2$,

$\vec{e}_{d+1,i} := (0^{i-1}, 1, 0^{2-i}) \in \mathbb{F}_q^2$, $\vec{\delta}_{d+1,i} \xleftarrow{\mathsf{U}} \mathbb{F}_q^2$,

$$\begin{aligned}
\boldsymbol{h}^*_{0,d+1,i} &:= (\overbrace{\delta\vec{\boldsymbol{e}}_{d+1,i}}^{2}, \overbrace{0^2}^{2}, \overbrace{\vec{\delta}_{d+1,i}}^{2}, \overbrace{0}^{1})_{\mathbb{B}^*_{d+1}},\\
\boldsymbol{h}^*_{1,d+1,i} &:= (\delta\vec{\boldsymbol{e}}_{d+1,i}, \vec{\boldsymbol{u}}_{d+1,i}, \vec{\delta}_{d+1,i}, 0)_{\mathbb{B}^*_{d+1}},\\
\boldsymbol{e}_{d+1,i} &:= (\omega\vec{\boldsymbol{e}}_{d+1,i}, \tau\vec{z}_{d+1,i}, 0^2, 0)_{\mathbb{B}_{d+1}},\\
\text{return } &(\mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}_t, \mathbb{B}^*_t\}_{t=0,d+1}, \{\mathbb{B}_t, \mathbb{B}^*_t\}_{t=1,..,d},\\
&\quad \boldsymbol{h}^*_{\beta,0}, \boldsymbol{e}_0, \{\boldsymbol{h}^*_{t,i}\}_{t=1,..,d;i=1,..,n_t}, \{\boldsymbol{h}^*_{\beta,d+1,i}, \boldsymbol{e}_{d+1,i}\}_{i=1,2}),
\end{aligned}$$

for $\beta \xleftarrow{\cup}\{0, 1\}$. For a probabilistic machine $\mathcal{B}$, the advantage of $\mathcal{B}$ for Problem 5, $\mathsf{Adv}^{\mathsf{P5}}_{\mathcal{B}}(\lambda)$, is similarly defined as in Definition 4.

Lemma 21. *For any adversary $\mathcal{B}$, there is a probabilistic machine $\mathcal{E}$, whose running time is essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}^{\mathsf{P5}}_{\mathcal{B}}(\lambda) \leq \mathsf{Adv}^{DLIN}_{\mathcal{E}}(\lambda) + 5/q$.*

E. How to Relax the Restriction that $\widetilde{\rho}$ Is Injective

We assume that $\phi \in \mathbb{N}$ is given in the system. For any access structure $\mathbb{S} := (M, \rho)$ for ciphertext in the CP-FE scheme, $\phi \geq \max_{t=1}^{d} \sharp\{i \mid \widetilde{\rho}(i) = t\}$. (In the proposed CP-FE scheme in Sect. 5, we assume that $\phi := 1$.)

We will show how to modify the CP-FE scheme to allow $\phi > 1$ with preserving the security of the CP-FE scheme in Sect. 5. We can also show the similar modification of the KP-FE scheme to allow $\phi > 1$.

In a recent work [47], another technique to allow $\phi > 1$ in an efficient manner was proposed.

E.1. *The Modified CP-FE Scheme*

1. As for Setup, given $(1^\lambda, \vec{n} := (d; n_1, \ldots, n_d))$, execute $\mathsf{Setup}(1^\lambda, \vec{n}' := (d; n'_1, \ldots, n'_d))$ such that $n'_t := n_t + \phi$ for $t = 1, \ldots, d$.
2. As for KeyGen, given $(\mathsf{pk}, \mathsf{sk}, \Gamma := \{(t, \vec{x}_t := (x_{t,1}, \ldots, x_{t,n_t}) \in \mathbb{F}_q^{n_t}) \mid 1 \leq t \leq d\})$ execute the same procedure as KeyGen except that:

$$\boldsymbol{k}^*_t := (\overbrace{\delta\vec{x}_t, 0^\phi}^{n'_t}, \overbrace{0^{n'_t}}^{n'_t}, \overbrace{\vec{\varphi}_t}^{n'_t}, \overbrace{0}^{1})_{\mathbb{B}^*_t} \quad \text{for } (t, \vec{x}_t) \in \Gamma,$$

3. As for Enc, given $(\mathsf{pk}, m, \mathbb{S} := (M, \rho))$, execute the same procedure as Enc except that:

$$\begin{aligned}
&\text{if } \rho(i) = (t, \vec{v}_i := (v_{i,1}, \ldots, v_{i,n_t}) \in \mathbb{F}_q^{n_t}) \qquad \eta_i, \tau_i \xleftarrow{\cup} \mathbb{F}_q,\\
&\boldsymbol{c}_i := (\overbrace{s_i\vec{\boldsymbol{e}}_{t,1} + \theta_t\vec{v}_i, 0^{\kappa-1}, \tau_i, 0^{\varphi-\kappa}}^{n'_t}, \overbrace{0^{n'_t}}^{n'_t}, \overbrace{0^{n'_t}}^{n'_t}, \overbrace{\eta_i}^{1})_{\mathbb{B}_t}
\end{aligned}$$

$$\text{if } \rho(i) = \neg(t, \vec{v}_i), \qquad \eta_i, \tau_i \xleftarrow{\cup} \mathbb{F}_q,$$

$$\boldsymbol{c}_i := (\ s_i \vec{v}_i, \overbrace{0^{\kappa-1}}^{n'_t}, \tau_i, \overbrace{0^{\phi-\kappa}}^{n'_t}, \overbrace{0^{n'_t}}^{n'_t}, \overbrace{0^{n'_t}}^{n'_t}, \overbrace{\eta_i}^{1}\ )_{\mathbb{B}_t},$$

where i is the κth index such that $\widetilde{\rho}(i) = t$ and $\sharp\{j < i \mid \widetilde{\rho}(j) = t\} = \kappa - 1$.

E.2. *Generalized Version of Lemma 3*

For $\vec{p} := (p_1, \ldots, p_s) \in \mathbb{F}_q^s$, let

$$C_{\vec{p}} := \left\{ (\vec{x}, \vec{v}_1, \ldots, \vec{v}_s) \;\middle|\; \begin{array}{l} \vec{x} \neq \vec{0},\ \vec{v}_i \neq \vec{0},\ \vec{x} \cdot \vec{v}_i = p_i \text{ for } i = 1, \ldots, s \\ \{\vec{v}_i\}_{i=1,\ldots,s} \text{ are linearly independent over } \mathbb{F}_q, \end{array} \right\} \subset \mathbb{F}_q^n \times (\mathbb{F}_q^n)^s.$$

Lemma 22. *For all $\vec{p}$ such that $C_{\vec{p}} \neq \emptyset$, for all $(\vec{x}, \vec{v}_1, \ldots, \vec{v}_s) \in C_{\vec{p}}$, and $(\vec{r}, \vec{w}_1, \ldots, \vec{w}_s) \in C_{\vec{p}}$,*

$$\Pr_{Z \xleftarrow{\cup} GL(n, \mathbb{F}_q),} \left[\vec{x}U = \vec{r} \ \wedge\ \vec{v}_i Z = \vec{w}_i \text{ for } i = 1, \ldots, s\right] = \frac{1}{\sharp C_{\vec{p}}},$$

where $U := (Z^{-1})^{\mathrm{T}}$.

Proof. **Case that there exists an i such that $\vec{x} \cdot \vec{v}_i = p_i \neq 0$:** We can assume that $p_i \neq 0$ for $i = 1, \ldots, t$, $p_i = 0$ for $i = t+1, \ldots, s$ through an appropriate change of order of coordinates. Then $t \geq 1$.

For normalized tuple of vectors

$$\vec{x} = (p_1, \ldots, p_t, 0, \ldots, 0), \qquad \vec{v}_i := (\overbrace{0, \ldots, 0}^{i-1}, 1, \overbrace{0, \ldots, 0}^{n-i}) \quad \text{for } i = 1, \ldots, s, \quad (21)$$

we will show that $(\vec{x}U, \vec{v}_1 Z, \ldots, \vec{v}_s Z)$ is uniformly distributed on $C_{\vec{p}}$ for $Z \xleftarrow{\cup} GL(n, \mathbb{F}_q)$, $U := (Z^{-1})^{\mathrm{T}}$. By that, for any pair $(\vec{x}, \vec{v}_1, \ldots, \vec{v}_s) \in C_{\vec{p}}$, we see that $(\vec{x}U, \vec{v}_1 Z, \ldots, \vec{v}_s Z)$ is uniformly distributed on $C_{\vec{p}}$ for $Z \xleftarrow{\cup} GL(n, \mathbb{F}_q)$, $U := (Z^{-1})^{\mathrm{T}}$. Therefore, we consider $(\vec{x}, \vec{v}_1, \ldots, \vec{v}_s)$ given by Eq. (21) in the following.

For the proof, we define

$$C^{(i)}_{p_i} := \left\{ (\vec{x}, \vec{v}_1, \ldots, \vec{v}_s) \;\middle|\; \begin{array}{l} \vec{x} \neq \vec{0},\ \vec{v}_i \neq \vec{0},\ \vec{x} \cdot \vec{v}_i = p_i, \\ \{\vec{v}_i\}_{i=1,\ldots,s} \text{ are linearly independent over } \mathbb{F}_q, \end{array} \right\} \subset \mathbb{F}_q^n \times (\mathbb{F}_q^n)^s,$$

for $i = 1, \ldots, s$, then $C_{\vec{p}} = \cap_{i=1}^s C^{(i)}_{p_i}$. Since $Z = (z_{i,j})$ and Eq. (18) holds, we have

$$\vec{x}U = \frac{\sum_{i=1}^t p_i \cdot \widetilde{\vec{z}}_i}{\det Z} = \frac{1}{\det Z} \sum_{i=1}^t p_i (\widetilde{z}_{i,1}, \ldots, \widetilde{z}_{i,n}), \vec{v}_i Z = \vec{z}_i = (z_{i,1}, \ldots, z_{i,n}) \quad \text{for } i = 1, \ldots, s.$$

From Lemma 3, we see that $(\vec{x}U, \vec{v}_1 Z, \ldots, \vec{v}_s Z)$ is in $C_{p_i}^{(i)}$ for $i = 1, \ldots, s$ and moreover it is uniformly distributed in $C_{\vec{p}} = \cap_{i=1}^{s} C_{p_i}^{(i)}$ when $Z \xleftarrow{\cup} GL(n, \mathbb{F}_q)$, i.e., $(\vec{x}U, \vec{v}_1 Z, \ldots, \vec{v}_s Z)$ is uniformly distributed in $C_{\vec{p}}$ when $Z \xleftarrow{\cup} GL(n, \mathbb{F}_q)$.

Case that $p_i = \vec{x} \cdot \vec{v}_i = 0$ **for all** $1 \le i \le s$: For normalized tuple

$$\vec{x} = (\overbrace{0, \ldots, 0}^{s}, 1, \overbrace{0, \ldots, 0}^{n-s-1}), \quad \vec{v}_i := (\overbrace{0, \ldots, 0}^{i-1}, 1, \overbrace{0, \ldots, 0}^{n-i}) \quad \text{for } i = 1, \ldots, s,$$

we will show that $(\vec{x}U, \vec{v}_1 Z, \ldots, \vec{v}_s Z)$ is uniformly distributed on $C_{\vec{0}}$ for $Z \xleftarrow{\cup} GL(n, \mathbb{F}_q)$, $U := (Z^{-1})^{\mathrm{T}}$ because of the similar reason as above (where $\vec{0} := (0, \ldots, 0)$).

Since $Z = (z_{i,j})$ and Eq. (18) holds, we have

$$\vec{x}U = \frac{\tilde{\vec{z}}_{s+1}}{\det Z} = \frac{1}{\det Z}(\tilde{z}_{s+1,1}, \ldots, \tilde{z}_{s+1,n}), \quad \vec{v}_i Z = \vec{z}_i = (z_{i,1}, \ldots, z_{i,n}) \quad \text{for } i = 1, \ldots, s.$$

Cofactors $\tilde{z}_{s+1,j}$ are determined by $n-1$ row vectors, $\vec{z}_1, \ldots, \vec{z}_s, \vec{z}_{s+2}, \ldots, \vec{z}_n$ of Z. In particular, $\vec{z}_{s+1}$ is related to only term $\det Z$ in $(\vec{x}U, \vec{v}Z)$.

First, since $\tilde{\vec{z}}_{s+1}$ is orthogonal to hyperplane $\mathsf{span}\langle \vec{z}_1, \ldots, \vec{z}_s, \vec{z}_{s+2}, \ldots, \vec{z}_n \rangle$ which is uniformly distributed in the spaces of hyperplanes H with the condition $\vec{z}_1, \ldots, \vec{z}_s \in H$ when $Z \xleftarrow{\cup} GL(n, \mathbb{F}_q)$ with fixed (random) $\vec{z}_1, \ldots, \vec{z}_s$, we see that $\tilde{\vec{z}}_{s+1}$ is distributed uniformly on the orthogonal space to $\mathsf{span}\langle \vec{z}_1, \ldots, \vec{z}_s \rangle (= \mathsf{span}\langle \vec{v}_1 Z, \ldots, \vec{v}_s Z \rangle)$ up to scalar multiplication. Moreover, since $\det Z$ is uniformly distributed in $\mathbb{F}_q^{\times}$ when $\vec{z}_{s+1}$ is uniformly distributed in $\mathbb{F}_q^n \setminus \{\vec{0}\}$ such that $\det Z \neq 0$ and $\vec{z}_{s+1}$ is related to only $\det Z$ in $(\vec{x}U, \vec{v}Z)$, we see that $\vec{x}U = \frac{\tilde{\vec{z}}_{s+1}}{\det Z}$ is distributed uniformly on the space orthogonal to $\mathsf{span}\langle \vec{z}_1, \ldots, \vec{z}_s \rangle$, i.e., $(\vec{x}U, \vec{v}Z)$ is uniformly distributed on C_0 when $Z \xleftarrow{\cup} GL(n, \mathbb{F}_q)$. □

E.3. *Security*

We can prove the security of the modified CP-FE scheme in a manner similar to that of Theorem 2 except that Problem 2 is changed to Modified Problem 2, Lemma 10 is changed, where $\mathcal{B}_2^+$'s simulation is executed on Modified Problem 2, Game 2-h^+ is changed to Modified Game 2-h^+, and Claim 2 is proven based on Lemma 22 in place of Lemma 3.

Here we only show the essence of the change by using Modified Game 2-h^+. The Modified Game 2-h^+ is the same as Game 2-h^+ except that $Z_t \xleftarrow{\cup} GL(n'_t, \mathbb{F}_q)$, $U_t := (Z_t^{-1})^{\mathrm{T}}$ for $t = 1, \ldots, d$, where for each t such that $\{i_\kappa \mid \widetilde{\rho}(i_\kappa) = t,\ 1 \le \kappa \le \phi\}\}$ is not empty, and for $\kappa = 1, \ldots, \phi$, the framed part by a box in $\boldsymbol{k}_t^*$ in Eq. (14) is $(\vec{x}_t, 0^\phi) \cdot U_t$, and the framed parts by a box in $\boldsymbol{c}_i$ $\left(:= \boldsymbol{c}_{i_\kappa}\right)$ in Eq. (15) are $(a_i \vec{e}_{t,1} + \pi_i \vec{v}_i, 0^{\kappa-1}, \tau'_i, 0^{\phi-\kappa}) \cdot Z_t$ and $(a_i \vec{v}_i, 0^{\kappa-1}, \tau'_i, 0^{\phi-\kappa}) \cdot Z_t$, where $\tau'_i \xleftarrow{\cup} \mathbb{F}_q$ for $i = 1, \ldots, \ell$. By using Modified Problem 2, $\mathcal{B}_2^+$ can simulate the ciphertexts, $\boldsymbol{c}_{i_\kappa}$. By applying Lemma 22, we can prove Claim 2 for the changed simulation by $\mathcal{B}_2^+$ in a manner similar to the proof of Claim 2.

F. Special Cases

This section describes special cases, KP-ABE and CP-ABE, of the proposed FE schemes given in Sects. 4 and 5. Here, the underlying attribute vectors, $\{\vec{x}_t\}_{t\in\{1,\ldots,d\}}$ and $\{\vec{v}_i\}_{i\in\{1,\ldots,\ell\}}$, are specialized to two-dimensional vectors for the equality relation, e.g., $\vec{x}_t := (1, x_t)$ and $\vec{v}_i := (v_i, -1)$, where $\vec{x}_t \cdot \vec{v}_i = 0$ iff $x_t = v_i$. These schemes are also adaptively payload-hiding under the DLIN assumption.

F.1. *KP-ABE with Non-Monotone Access Structures*

$\mathsf{Setup}(1^\lambda,\ \vec{n} := (d; 2, \ldots, 2)) :\quad (\mathsf{param}_{\vec{n}}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0,\ldots,d}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}(1^\lambda, \vec{n}),$

$\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5}),\ \ \widehat{\mathbb{B}}_t := (\boldsymbol{b}_{t,1}, \boldsymbol{b}_{t,2}, \boldsymbol{b}_{t,7})\ \ \text{for } t = 1, .., d,$

$\widehat{\mathbb{B}}_0^* := (\boldsymbol{b}_{0,1}^*, \boldsymbol{b}_{0,3}^*, \boldsymbol{b}_{0,4}^*),\ \ \widehat{\mathbb{B}}_t^* := (\boldsymbol{b}_{t,1}^*, \boldsymbol{b}_{t,2}^*, \boldsymbol{b}_{t,5}^*, \boldsymbol{b}_{t,6}^*)\ \ \text{for } t = 1, .., d,$

return $\mathsf{pk} := (1^\lambda, \mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}_t\}_{t=0,\ldots,d}),\ \ \mathsf{sk} := \{\widehat{\mathbb{B}}_t^*\}_{t=0,\ldots,d},$

$\mathsf{KeyGen}(\mathsf{pk},\ \mathsf{sk},\ \mathbb{S} := (M, \rho)) :$

$\vec{f} \xleftarrow{\mathsf{U}} \mathbb{F}_q^r,\ \vec{s}^{\mathrm{T}} := (s_1, \ldots, s_\ell)^{\mathrm{T}} := M \cdot \vec{f}^{\mathrm{T}},\ s_0 := \vec{1} \cdot \vec{f}^{\mathrm{T}},\ \eta_0 \xleftarrow{\mathsf{U}} \mathbb{F}_q,$

$\boldsymbol{k}_0^* := (-s_0,\ 0,\ 1,\ \eta_0,\ 0)_{\mathbb{B}_0^*},$

for $i = 1, \ldots, \ell,$

if $\rho(i) = (t, \vec{v}_i := (v_i, -1) \in \mathbb{F}_q^2 \backslash \{\vec{0}\}),\qquad \theta_i \xleftarrow{\mathsf{U}} \mathbb{F}_q,\ \vec{\eta}_i \xleftarrow{\mathsf{U}} \mathbb{F}_q^2,$

$$\boldsymbol{k}_i^* := (\overbrace{\ s_i + \theta_i v_i,\quad -\theta_i,\ }^{2}\ \overbrace{\ 0,\quad 0,\ }^{2}\ \overbrace{\ \vec{\eta}_i,\ }^{2}\ \overbrace{\ 0\ }^{1})_{\mathbb{B}_t^*},$$

if $\rho(i) = \neg(t, \vec{v}_i),\qquad \vec{\eta}_i \xleftarrow{\mathsf{U}} \mathbb{F}_q^2,$

$$\boldsymbol{k}_i^* := (\overbrace{\ s_i v_i,\quad -s_i,\ }^{2}\ \overbrace{\ 0,\quad 0,\ }^{2}\ \overbrace{\ \vec{\eta}_i,\ }^{2}\ \overbrace{\ 0\ }^{1})_{\mathbb{B}_t^*},$$

return $\mathsf{sk}_{\mathbb{S}} := (\mathbb{S}, \boldsymbol{k}_0^*, \boldsymbol{k}_1^*, \ldots, \boldsymbol{k}_\ell^*).$

$\mathsf{Enc}(\mathsf{pk},\ m,\ \Gamma := \{(t, \vec{x}_t := (1, x_t) \in \mathbb{F}_q^2 \backslash \{\vec{0}\}) \mid 1 \le t \le d\}) :$

$\omega, \varphi_0, \varphi_t, \zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q \text{ for } (t, \vec{x}_t) \in \Gamma,$

$\boldsymbol{c}_0 := (\omega,\ 0,\ \zeta,\ 0,\ \varphi_0)_{\mathbb{B}_0},$

$$\boldsymbol{c}_t := (\overbrace{\ \omega,\quad \omega x_t,\ }^{2}\ \overbrace{\ 0,\quad 0,\ }^{2}\ \overbrace{\ 0,\quad 0,\ }^{2}\ \overbrace{\ \varphi_t\ }^{1})_{\mathbb{B}_t} \qquad \text{for } (t, \vec{x}_t) \in \Gamma,$$

$c_{d+1} := g_T^{\zeta} m,\ \ \mathsf{ct}_\Gamma := (\Gamma, \boldsymbol{c}_0, \{\boldsymbol{c}_t\}_{(t,\vec{x}_t)\in\Gamma}, c_{d+1}),$

return $\mathsf{ct}_\Gamma.$

$\mathsf{Dec}(\mathsf{pk},\ \mathsf{sk}_{\mathbb{S}} := (\mathbb{S}, \boldsymbol{k}_0^*, \boldsymbol{k}_1^*, \ldots, \boldsymbol{k}_\ell^*),\ \mathsf{ct}_\Gamma := (\Gamma, \boldsymbol{c}_0, \{\boldsymbol{c}_t\}_{(t,\vec{x}_t)\in\Gamma}, c_{d+1})) :$

If $\mathbb{S} := (M, \rho)$ accepts $\Gamma := \{(t, \vec{x}_t)\}$, then compute I and $\{\alpha_i\}_{i\in I}$ such that

$\vec{1} = \sum_{i\in I} \alpha_i M_i$, where M_i is the ith row of M, and

$I \subseteq \{i \in \{1, \ldots, \ell\} \ \mid\ [\rho(i) = (t, \vec{v}_i) \ \wedge\ (t, \vec{x}_t) \in \Gamma \ \wedge\ v_i = x_t]$

$$\vee\ [\rho(i) = \neg(t, \vec{v}_i)\ \wedge\ (t, \vec{x}_t) \in \Gamma\ \wedge\ v_i \neq x_t]\ \},$$

$$K := e(\boldsymbol{c}_0, \boldsymbol{k}_0^*) \prod_{i \in I\ \wedge\ \rho(i) = (t, \vec{v}_i)} e(\boldsymbol{c}_t, \boldsymbol{k}_i^*)^{\alpha_i} \prod_{i \in I\ \wedge\ \rho(i) = \neg(t, \vec{v}_i)} e(\boldsymbol{c}_t, \boldsymbol{k}_i^*)^{\alpha_i / (v_i - x_t)},$$

return $m' := c_{d+1}/K$.

F.2. *PE for Inner-Products*

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

$\mathcal{G}_{\mathsf{ob}}{}'(1^\lambda, N)$: $\mathsf{param}'_{\mathbb{V}} := (q, \mathbb{V}, \mathbb{G}_T, \mathbb{A}, e) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{dpvs}}(1^\lambda, N)$, $\psi \xleftarrow{\mathsf{U}} \mathbb{F}_q^\times$,
$X := (\chi_{i,j}) \xleftarrow{\mathsf{U}} GL(N, \mathbb{F}_q)$, $(\vartheta_{i,j}) := \psi \cdot (X^{\mathrm{T}})^{-1}$, $g_T := e(G, G)^\psi$, $\mathsf{param}_{\mathbb{V}} := (\mathsf{param}'_{\mathbb{V}}, g_T)$,
$\boldsymbol{b}_i := \sum_{j=1}^N \chi_{i,j} \boldsymbol{a}_j$, $\mathbb{B} := (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_N)$, $\boldsymbol{b}_i^* := \sum_{j=1}^N \vartheta_{i,j} \boldsymbol{a}_j$, $\mathbb{B}^* := (\boldsymbol{b}_1^*, \ldots, \boldsymbol{b}_N^*)$,
return $(\mathsf{param}_{\mathbb{V}}, \mathbb{B}, \mathbb{B}^*)$.

F.2.1. *Construction*

In order to make a ciphertext shorter, we modify $t = 1$ space $\mathbb{V} := \mathbb{V}_1$ by adding one more dimension instead of using $t = 0$ space $\mathbb{V}_0$. This construction is similar to the IPE construction in Section 3.5 in [32].

Here, we assume that the first coordinate, x_1, of input vector, $\vec{x}$, is nonzero. We refer to Sect. 1.5 for notations on DPVS.

$\mathsf{Setup}(1^\lambda, n)$: $(\mathsf{param}_{\mathbb{V}}, \mathbb{B} := (\boldsymbol{b}_0, \ldots, \boldsymbol{b}_{3n+1}), \mathbb{B}^* := (\boldsymbol{b}_0^*, \ldots, \boldsymbol{b}_{3n+1}^*)) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}{}'(1^\lambda, 3n+2)$,
$\widehat{\mathbb{B}} := (\boldsymbol{b}_0, \ldots, \boldsymbol{b}_n, \boldsymbol{b}_{3n+1})$, $\widehat{\mathbb{B}}^* := (\boldsymbol{b}_0^*, \ldots, \boldsymbol{b}_n^*, \boldsymbol{b}_{2n+1}^*, \ldots, \boldsymbol{b}_{3n}^*)$,
return $\mathsf{pk} := (1^\lambda, \mathsf{param}_{\mathbb{V}}, \widehat{\mathbb{B}})$, $\mathsf{sk} := \widehat{\mathbb{B}}^*$.

$\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, \vec{v} \in \mathbb{F}_q^n)$: $\sigma \xleftarrow{\mathsf{U}} \mathbb{F}_q$, $\vec{\eta} \xleftarrow{\mathsf{U}} \mathbb{F}_q^n$,

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

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

$\mathsf{Enc}(\mathsf{pk}, m, \vec{x} \in \mathbb{F}_q^n)$: $\omega, \varphi, \zeta \xleftarrow{\mathsf{U}} \mathbb{F}_q$,

$$\boldsymbol{c}_1 := (\ \overbrace{\zeta}^{1},\ \overbrace{\omega\vec{x}}^{n},\ \overbrace{0^n}^{n},\ \overbrace{0^n}^{n},\ \overbrace{\varphi}^{1}\ )_{\mathbb{B}}, \quad c_2 := g_T^\zeta m,$$

return $\mathsf{ct}_{\vec{x}} := (\boldsymbol{c}_1, c_2)$.

$\mathsf{Dec}(\mathsf{pk}, \mathsf{sk}_{\vec{v}} := \boldsymbol{k}^*, \mathsf{ct}_{\vec{x}} := (\boldsymbol{c}_1, c_2))$: $m' := c_2 / e(\boldsymbol{c}_1, \boldsymbol{k}^*)$,
return m'.

[Correctness] If $\vec{x} \cdot \vec{v} = 0$, then $e(\boldsymbol{c}_1, \boldsymbol{k}^*) = g_T^{\zeta + \omega\sigma\vec{x}\cdot\vec{v}} = g_T^\zeta$.

Remark 5. The differences between the proposed IPE scheme and the IPE scheme in [32] are:

1. While the scheme in [32] employed a $(2n+3)$-dimensional vector space, the proposed scheme employs a $(3n+2)$-dimensional one. The keys in [32] have only one-dimensional randomness space, but those in our construction have n-dimensional randomness space. The security assumption in [32] is the n-eDDH, a non-standard (and non-static) assumption, while it is a standard (and static) assumption, the DLIN, in our scheme. More precisely, the security of Problems 1 and 2 on a $(2n+3)$-dimensional space is reduced to the n-eDDH assumption in [32], while in the proposed scheme, the security of these problems on a $(3n+2)$-dimensional space is reduced to the DLIN assumption. In other words, we achieve the DLIN-based security (higher security) at the cost of increasing $(n-1)$ dimensions for the randomness space of keys (less efficiency).
2. While scalar ζ in a ciphertext $\boldsymbol{c}_1$ is a coefficient of the $(2n+1)$th basis vector $\boldsymbol{b}_{2n+1}$ in [32], it is that of 0th basis vector $\boldsymbol{b}_0$ here. It is just a change of notation, i.e., not essential one.

F.2.2. *(Weakly) Attribute-Hiding Security*

The notion of *adaptively weakly attribute-hiding* security, where a type of key queries are not allowed, and the advantage $\mathsf{Adv}^{\mathsf{IPE,\ wAH}}_{\mathcal{A}}(\lambda)$ of adversary $\mathcal{A}$ are defined in Definition 17 of [32].

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

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,

$$\mathsf{Adv}^{\mathit{IPE,\ wAH}}_{\mathcal{A}}(\lambda) \le \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_1}(\lambda) + \sum_{h=1}^{\nu} \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_{2,h}}(\lambda) + \epsilon,$$

where $\mathcal{E}_{2,h}(\cdot) := \mathcal{E}_2(h,\cdot)$, ν is the maximum number of $\mathcal{A}$'s key queries and $\epsilon := (6\nu+5)/q$.

We will employ Problem 1' and Problem 2' for the proof of Theorem 5, which are almost the same as Problem 1 (in Definition 4) and Problem 2 (in Definition 5), respectively. For completeness, we describe them and the security lemmas here.

Definition 24. (*Problem 1'*) Problem 1' is to guess β, given $(\mathsf{param}_{\mathbb{V}}, \mathbb{B}, \widehat{\mathbb{B}}^*, \boldsymbol{e}_{\beta,1}, \{\boldsymbol{e}_i\}_{i=2,\ldots,n}) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{P1'}}_{\beta}(1^\lambda, n)$, where

$$\begin{aligned}
&\mathcal{G}^{\mathsf{P1'}}_{\beta}(1^\lambda, n): \quad (\mathsf{param}_{\mathbb{V}}, \mathbb{B}, \mathbb{B}^*) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}{}'(1^\lambda, 3n+2),\\
&\widehat{\mathbb{B}}^* := (\boldsymbol{b}^*_0, \ldots, \boldsymbol{b}^*_n, \boldsymbol{b}^*_{2n+1}, \ldots, \boldsymbol{b}^*_{3n+1}),\\
&\omega \xleftarrow{\mathsf{U}} \mathbb{F}_q, \quad \vec{e}_1 := (1, 0^{n-1}) \in \mathbb{F}_q^n, \ \vec{z} \xleftarrow{\mathsf{U}} \mathbb{F}_q^n, \ \gamma \xleftarrow{\mathsf{U}} \mathbb{F}_q,
\end{aligned}$$

$$
\begin{aligned}
&\overbrace{\quad}^{1}\ \overbrace{\quad\quad}^{n}\ \overbrace{\quad\quad}^{n}\ \overbrace{\quad\quad}^{n}\ \overbrace{\quad}^{1}\\
&\boldsymbol{e}_{0,1} := (\ 0,\ \ \omega\vec{e}_1,\ \ 0^n,\ \ 0^n,\ \ \gamma\ )_{\mathbb{B}},\\
&\boldsymbol{e}_{1,1} := (\ 0,\ \ \omega\vec{e}_1,\ \ \vec{z},\ \ 0^n,\ \ \gamma\ )_{\mathbb{B}},\\
&\boldsymbol{e}_i := \omega\boldsymbol{b}_i \ \text{ for } i = 2,\ldots,n,\\
&\text{return } (\mathsf{param}_{\mathbb{V}}, \mathbb{B}, \widehat{\mathbb{B}}^*, \boldsymbol{e}_{\beta,1}, \{\boldsymbol{e}_i\}_{i=2,\ldots,n}),
\end{aligned}
$$

for $\beta \xleftarrow{\mathsf{U}} \{0,1\}$. For a probabilistic machine $\mathcal{B}$, the advantage of $\mathcal{B}$ for Problem 1', $\mathsf{Adv}^{\mathsf{P1'}}_{\mathcal{B}}(\lambda)$, is similarly defined as in Definition 4.

Lemma 23. *For any adversary $\mathcal{B}$, there exist probabilistic machines $\mathcal{E}$, whose running times are essentially the same as that of $\mathcal{B}$, such that for any security parameter λ, $\mathsf{Adv}^{\mathsf{P1'}}_{\mathcal{B}}(\lambda) \le \mathsf{Adv}^{\mathit{DLIN}}_{\mathcal{E}}(\lambda) + 5/q$.*

The proof of Lemma 23 is similar to that of Lemma 1.

Definition 25. (*Problem 2'*) Problem 2' is to guess β, given $(\mathsf{param}_{\mathbb{V}}, \widehat{\mathbb{B}}, \mathbb{B}^*, \{\boldsymbol{h}^*_{\beta,i}, \boldsymbol{e}_i\}_{i=1,\ldots,n}) \xleftarrow{\mathsf{R}} \mathcal{G}^{\mathsf{P2'}}_{\beta}(1^\lambda, n)$, where

$$
\begin{aligned}
&\mathcal{G}^{\mathsf{P2'}}_{\beta}(1^\lambda, n): \quad (\mathsf{param}_{\mathbb{V}}, \mathbb{B}, \mathbb{B}^*) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}'}(1^\lambda, 3n+2),\\
&\widehat{\mathbb{B}} := (\boldsymbol{b}_0, \ldots, \boldsymbol{b}_n, \boldsymbol{b}_{2n+1}, \ldots, \boldsymbol{b}_{3n+1}),\\
&\delta, \omega \xleftarrow{\mathsf{U}} \mathbb{F}_q,\ \tau \xleftarrow{\mathsf{U}} \mathbb{F}_q^{\times},\ \begin{pmatrix}\vec{z}_1\\ \vdots\\ \vec{z}_n\end{pmatrix} := Z \xleftarrow{\mathsf{U}} GL(n, \mathbb{F}_q),\ \begin{pmatrix}\vec{u}_1\\ \vdots\\ \vec{u}_n\end{pmatrix} := (Z^{-1})^{\mathrm{T}},\\
&\text{for } i = 1,\ldots,n;\\
&\vec{e}_i := (0^{i-1}, 1, 0^{n-i}) \in \mathbb{F}_q^n,\ \vec{\delta}_i \xleftarrow{\mathsf{U}} \mathbb{F}_q^n,\\
&\qquad\qquad \overbrace{\quad}^{1}\ \overbrace{\quad\quad}^{n}\ \overbrace{\quad\quad}^{n}\ \overbrace{\quad\quad}^{n}\ \overbrace{\quad}^{1}\\
&\boldsymbol{h}^*_{0,i} := (\ 0,\ \ \delta\vec{e}_i,\ \ 0^n,\ \ \vec{\delta}_i,\ \ 0\ )_{\mathbb{B}^*}\\
&\boldsymbol{h}^*_{1,i} := (\ 0,\ \ \delta\vec{e}_i,\ \ \vec{u}_i,\ \ \vec{\delta}_i,\ \ 0\ )_{\mathbb{B}^*}\\
&\boldsymbol{e}_i := \ (\ 0,\ \ \omega\vec{e}_i,\ \ \tau\vec{z}_i,\ \ 0^n,\ \ 0\ )_{\mathbb{B}},\\
&\text{return } (\mathsf{param}_{\mathbb{V}}, \widehat{\mathbb{B}}, \mathbb{B}^*, \{\boldsymbol{h}^*_{\beta,i}, \boldsymbol{e}_i\}_{i=1,\ldots,n}),
\end{aligned}
$$

for $\beta \xleftarrow{\mathsf{U}} \{0,1\}$. 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 4.

Lemma 24. *For any adversary $\mathcal{B}$, 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, $\mathsf{Adv}^{\mathit{P2'}}_{\mathcal{B}}(\lambda) \le \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}}(\lambda) + 5/q$.*

The proof of Lemma 24 is similar to that of Lemma 2.

Proof of Theorem 5. To prove Theorem 5, 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 which 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:

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

where $\sigma \xleftarrow{\cup} \mathbb{F}_q$ and $\vec{\eta} \in \mathbb{F}_q^n$. The challenge ciphertext for challenge plaintexts $(m^{(0)}, m^{(1)})$ and vectors $(\vec{x}^{(0)}, \vec{x}^{(1)})$ is:

$$\boldsymbol{c}_1 := (\ \boxed{\zeta},\ \boxed{\omega\vec{x}^{(b)}},\ \boxed{0^n},\ 0^n,\ \varphi\ )_{\mathbb{B}},\quad c_2 := g_T^{\zeta} m^{(b)},$$

where $b \xleftarrow{\cup} \{0, 1\}$; $\zeta, \omega, \varphi \xleftarrow{\cup} \mathbb{F}_q$.

Game 1 : Game 1 is the same as Game 0 except that $\boldsymbol{c}_1$ of the challenge ciphertext is:

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

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

Game 2-$\boldsymbol{h}$ ($\boldsymbol{h} = \mathbf{1}, \ldots, \boldsymbol{\nu}$): Game 2-0 is Game 1. Game 2-h is the same as Game 2-$(h-1)$ except the reply to the hth key query for $\vec{v}$ is:

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

where $\vec{w} \xleftarrow{\cup} \mathbb{F}_q^n$, and all the other variables are generated as in Game 2-$(h-1)$.

Game 3 : Game 3 is the same as Game 2-ν except that $\boldsymbol{c}_1$ of the challenge ciphertext is

$$\boldsymbol{c}_1 := (\ \boxed{\zeta'},\ \boxed{\vec{x}'},\ \vec{r},\ 0^n,\ \varphi\ )_{\mathbb{B}},\quad c_2 := g_T^{\zeta} m^{(b)},$$

where $\zeta' \xleftarrow{\cup} \mathbb{F}_q$, $\vec{x}' \xleftarrow{\cup} \mathbb{F}_q^n$ (i.e., independent from $b \xleftarrow{\cup} \{0, 1\}$), and all the other variables are generated as in Game 2-ν.

Let $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2-h)}(\lambda)$ 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 $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{IPE,wAH}}(\lambda)$ and it is obtained that $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda) = 0$ by Lemma 28.

We will show three lemmas (Lemmas 25–27) that evaluate the gaps between pairs of $\mathsf{Adv}_{\mathcal{A}}^{(0)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(1)}(\lambda)$, $\mathsf{Adv}_{\mathcal{A}}^{(2-h)}(\lambda)$ for $h = 1, \ldots, \nu$ and $\mathsf{Adv}_{\mathcal{A}}^{(3)}(\lambda)$. From these lemmas and Lemmas 23 and 24, we obtain

$$\begin{aligned}\mathsf{Adv}^{\mathsf{IPE,\,wAH}}_{\mathcal{A}}(\lambda) = \mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda) &\le \left|\mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(1)}_{\mathcal{A}}(\lambda)\right| + \sum_{h=1}^{\nu}\left|\mathsf{Adv}^{(2-(h-1))}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(2-h)}_{\mathcal{A}}(\lambda)\right| \\ &+ \left|\mathsf{Adv}^{(2-\nu)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda)\right| + \mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda) \\ &\le \mathsf{Adv}^{\mathsf{P1'}}_{\mathcal{B}_1}(\lambda) + \sum_{h=1}^{\nu}\mathsf{Adv}^{\mathsf{P2'}}_{\mathcal{B}_{2,h}}(\lambda) + \nu/q \le \mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_1}(\lambda) + \sum_{h=1}^{\nu}\mathsf{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_{2,h}}(\lambda) + (6\nu+5)/q.\end{aligned}$$

This completes the proof of Theorem 5. □

Lemma 25. *For any adversary $\mathcal{A}$, 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, $|\mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(1)}_{\mathcal{A}}(\lambda)| = \mathsf{Adv}^{\mathsf{P1'}}_{\mathcal{B}_1}(\lambda)$.*

Proof. In order to prove Lemma 25, we construct a probabilistic machine $\mathcal{B}_1$ against Problem 1' by using any adversary $\mathcal{A}$ in a security game (Game 0 or 1) as a black box.

The construction of $\mathcal{B}_1$ is the same as the machine $\mathcal{B}_0$ in the proof of Lemma 24 in [32] except for step 5. In the step, when $\mathcal{B}_1$ gets challenge plaintexts $(m^{(0)}, m^{(1)})$ and challenge attributes $(\vec{x}^{(0)}, \vec{x}^{(1)})$ (from $\mathcal{A}$), $\mathcal{B}_1$ calculates and returns $(\boldsymbol{c}_1, c_2)$ such that $\boldsymbol{c}_1 := \zeta \boldsymbol{b}_0 + x_1^{(b)}\boldsymbol{e}_{\beta,1} + \sum_{i=2}^{n} x_i^{(b)}\boldsymbol{e}_i$ and $c_2 := g_T^{\zeta} m^{(b)}$ where $\boldsymbol{e}_1$ and $\{\boldsymbol{e}_i\}_{i=2,\ldots,n}$ are from the Problem 1' instance $\mathcal{B}_1$ obtained, $\zeta \xleftarrow{\cup} \mathbb{F}_q$ and $b \xleftarrow{\cup} \{0, 1\}$.

Similar to Lemma 24 in [32], if $\beta = 0$, the distribution of $(\boldsymbol{c}_1, c_2)$ generated in step 5 is the same as that in Game 0. If $\beta = 1$, the distribution of $(\boldsymbol{c}_1, c_2)$ generated in step 5 is the same as that in Game 1.

Therefore, $|\mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(1)}_{\mathcal{A}}(\lambda)| = \mathsf{Adv}^{\mathsf{P1'}}_{\mathcal{B}_1}(\lambda)$. This completes the proof of Lemma 25. □

Lemma 26. *For any adversary $\mathcal{A}$, there exists a probabilistic machine $\mathcal{B}_2$, whose running time is essentially the same as that of $\mathcal{A}$, such that for any security parameter λ, $|\mathsf{Adv}^{(2-(h-1))}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(2-h)}_{\mathcal{A}}(\lambda)| \le \mathsf{Adv}^{\mathsf{P2'}}_{\mathcal{B}_{2,h}}(\lambda) + \frac{1}{q}$, where $\mathcal{B}_{2,h}(\cdot) := \mathcal{B}_2(h, \cdot)$.*

Proof. In order to prove Lemma 26, we construct a probabilistic machine $\mathcal{B}_2$ against Problem 2' by using any adversary $\mathcal{A}$ in a security game (Game 2-$(h-1)$ or 2-h) as a black box.

The construction of $\mathcal{B}_2$ is the same as the machine $\mathcal{B}(:= \mathcal{B}_k)$ in the proof of Lemma 25 in [32] except for the order of basis vectors. That is, while in the IPE scheme in [32], scalar ζ is a coefficient of the $(2n+1)$th basis vector $\boldsymbol{b}_{2n+1}$ in $\boldsymbol{c}_1$, in our IPE scheme, the scalar ζ is that of the 0th basis vector $\boldsymbol{b}_0$ in $\boldsymbol{c}_1$ (item 2 of Remark 5). Except for such a notational difference, the simulation of $\mathcal{B}_2$ is the same as that of $\mathcal{B}$ in the proof of Lemma 25 in [32].

Similar to Lemma 25 in [32], the pair of secret key $\boldsymbol{k}^*$ generated in case (b) of step 4 or 6 and ciphertext $\boldsymbol{c}_1$ generated in step 5 has the same distribution as that in Game 2-$(h-1)$ (resp. Game 2-h) when $\beta = 0$ (resp. $\beta = 1$) except with probability $\frac{1}{q}$.

Therefore, $|\mathsf{Adv}^{(2-(h-1))}_{\mathcal{A}}(\lambda) - \mathsf{Adv}^{(2-h)}_{\mathcal{A}}(\lambda)| \le \mathsf{Adv}^{\mathsf{P2'}}_{\mathcal{B}_{2,h}}(\lambda) + \frac{1}{q}$. This completes the proof of Lemma 26. □

Lemma 27. *For any adversary* $\mathcal{A}$, $\mathsf{Adv}^{(2-\nu)}_{\mathcal{A}}(\lambda) = \mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda)$.

Proof. To prove Lemma 27, we will show distribution $(\mathsf{param}_{\mathbb{V}}, \widehat{\mathbb{B}}, \{\boldsymbol{k}^{(j)*}\}_{j=1,\ldots,\nu}, \boldsymbol{c}_1, c_2)$ in Game 2-ν and that in Game 3 are equivalent. The proof is the same as that of Lemma 26 in [32] (except for a notational difference in item 2 of Remark 5). □

Lemma 28. *For any adversary* $\mathcal{A}$, $\mathsf{Adv}^{(3)}_{\mathcal{A}}(\lambda) = 0$.

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

G. HIPE Schemes

We will show two hierarchical IPE (HIPE) schemes in this appendix. The first one is more efficient but payload-hiding ("Appendix G.3"), and the second one is (weakly) attribute-hiding but less efficient ("Appendix G.4"), where these two schemes employ different delegation mechanisms.

G.1. *Key Idea in Constructing the Proposed HIPEs*

Both schemes (without delegations) are constructed from the KP-FE scheme in Sect. 4 by specializing the policy to be ℓ-out-of-ℓ threshold access structures.

Let $N := 2(\sum_{t=1}^{d} n_t) + 3$. The HIPE scheme in [32] employs one (large) vector space of dimension N, where public basis $\mathbb{B}$ and (master) secret basis $\mathbb{B}^*$ consists of N^2 elements in the pairing group $\mathbb{G}$. It leads to that KeyGen and Enc require $O(N^2)$ scalar multiplications, i.e., they become relatively slow. Our schemes are constructed using separated spaces $\mathbb{V}_0, \mathbb{V}_1, \ldots \mathbb{V}_d$ with dimensions 5 and $3n_t + 1$ for $t = 1, \ldots, d$ (see Sect. 4). Hence, the data sizes of dual bases $\{\mathbb{B}_t, \mathbb{B}^*_t\}_{t=0,1,\ldots,d}$ are $O(\sum_{t=1}^{d} n_t^2)$, and then functions KeyGen and Enc become more efficient than those in [32], where the sizes of the dual bases are $O((\sum_{t=1}^{d} n_t)^2)$.

The HIPE scheme in "Appendix G.3" makes $\{\mathbb{B}^*_t\}_{t=0,\ldots,d}$ public except $\boldsymbol{b}^*_{0,3}$, which is denoted by $\{\widehat{\mathbb{B}}^*_t\}_{t=0,\ldots,d}$, and $\mathsf{Delegate}$ uses $\{\widehat{\mathbb{B}}^*_t\}_{t=0,\ldots,d}$. Master secret key is only one vector $\boldsymbol{b}^*_{0,3}$. Since most of keys for delegation are public, secret key sk_ℓ can be small (compared to those in [32]). The scheme, however, cannot be attribute-hiding for $\vec{x}_t$ for any level $t = 1, \ldots, d$, because $\{\boldsymbol{b}^*_{t,1}, \ldots, \boldsymbol{b}^*_{t,n_t}\}_{t=1,\ldots,d} \subset \widehat{\mathbb{B}}^*_t$ are public.

To achieve both attribute-hiding and key delegatability, a (level-ℓ) secret key of the HIPE scheme in "Appendix G.4", sk_ℓ, consists of 3 types of vector elements, $\boldsymbol{k}^*_{\ell,\mathsf{dec}}, \boldsymbol{k}^*_{\ell,\mathsf{del},\cdot}, \boldsymbol{k}^*_{\ell,\mathsf{ran},\cdot}$ as in the HIPE scheme [32]. Element $\boldsymbol{k}^*_{\ell,\mathsf{dec}}$ is used for decryption, $\boldsymbol{k}^*_{\ell,\mathsf{del},\cdot}$ is used for delegation, i.e., for embedding any level-$(\ell+1)$ vector $\vec{v}_{\ell+1}$ in delegated key $\mathsf{sk}_{\ell+1}$, and $\boldsymbol{k}^*_{\ell,\mathsf{ran},\cdot}$ is used for re-randomization of a level-$(\ell+1)$ key, i.e., for making the distribution of a delegated key equal to that of a freshly generated key (see "Appendix G.4.2"). The secret key size is larger than that in "Appendix G.3" due to the additional elements, $\boldsymbol{k}^*_{\ell,\mathsf{del},\cdot}$ and $\boldsymbol{k}^*_{\ell,\mathsf{ran},\cdot}$.

G.2. *Special Notations for the Proposed HIPEs*

To express our delegation mechanisms in the HIPEs compactly, we will introduce new notations, here.

Since we use dual orthonormal basis generator $\mathcal{G}_{\mathsf{ob}}$ given in Sect. 2.1, $X_0 \xleftarrow{\mathsf{U}} GL(5, \mathbb{F}_q)$ and $X_t \xleftarrow{\mathsf{U}} GL(3n_t + 1, \mathbb{F}_q)$ for $t = 1, \ldots, d$. By arranging the matrices $X_0, X_1, \ldots X_d$ diagonally and other off-diagonal parts are zero, we consider a special from of bases generation matrix $X \in \mathbb{F}_q^{N \times N}$ with $N := 5 + \sum_{t=1}^{d}(3n_t + 1)$, where

$$X := \begin{pmatrix} X_0 & & & \\ & X_1 & & \\ & & \ddots & \\ & & & X_d \end{pmatrix},$$

and our HIPEs are constructed on the one vector space $\mathbb{V}$ ($\cong \mathbb{G}^N$) with special bases induced by X. In other words, the matrix X gives direct sum decomposition $\mathbb{V} \cong \mathbb{V}_0 \oplus \mathbb{V}_1 \oplus \cdots \oplus \mathbb{V}_d$ (resp. $\mathbb{V}^* \cong \mathbb{V}_0^* \oplus \mathbb{V}_1^* \oplus \cdots \oplus \mathbb{V}_d^*$), where $\mathbb{V}_t := \mathsf{span}\langle \mathbb{B}_t \rangle$ (resp. $\mathbb{V}_t^* := \mathsf{span}\langle \mathbb{B}_t^* \rangle$) for $t = 0, \ldots, d$. Based on this isomorphism, i.e., embedding of $\mathbb{V}_t$ (resp. $\mathbb{V}_t^*$) in $\mathbb{V}$ (resp. $\mathbb{V}^*$), we define the following notations as:

$$\begin{aligned}
&((\vec{x}_0)_{\mathbb{B}_0}, \ldots, (\vec{x}_d)_{\mathbb{B}_d}) + ((\vec{y}_0)_{\mathbb{B}_0}, \ldots, (\vec{y}_d)_{\mathbb{B}_d}) := ((\vec{x}_0 + \vec{y}_0)_{\mathbb{B}_0}, \ldots, (\vec{x}_d + \vec{y}_d)_{\mathbb{B}_d}) \\
&\quad \text{where } ((\vec{x}_0)_{\mathbb{B}_0}, \ldots, (\vec{x}_d)_{\mathbb{B}_d}),\ ((\vec{y}_0)_{\mathbb{B}_0}, \ldots, (\vec{y}_d)_{\mathbb{B}_d}) \in \mathbb{V} \cong \mathbb{V}_0 \oplus \mathbb{V}_1 \oplus \cdots \oplus \mathbb{V}_d, \\
&(\vec{x})_{\mathbb{B}_t} := ((\vec{0})_{\mathbb{B}_0}, \cdots, (\vec{0})_{\mathbb{B}_{t-1}}, (\vec{x})_{\mathbb{B}_t}, (\vec{0})_{\mathbb{B}_{t+1}}, \cdots, (\vec{0})_{\mathbb{B}_d}) \in \mathbb{V}, \\
&((\vec{x}_0)_{\mathbb{B}_0}, (\vec{x}_t)_{\mathbb{B}_t} : t = 1, \ldots, \ell) := ((\vec{x}_0)_{\mathbb{B}_0}, \ldots, (\vec{x}_\ell)_{\mathbb{B}_\ell}) := \textstyle\sum_{t=0}^{\ell} (\vec{x}_t)_{\mathbb{B}_t} \in \mathbb{V}, \\
&((\vec{x}_0)_{\mathbb{B}_0}, (\vec{x}_t)_{\mathbb{B}_t} : t = 1, \ldots, \ell, (\vec{x}_\tau)_{\mathbb{B}_\tau}) := ((\vec{x}_0)_{\mathbb{B}_0}, \ldots, (\vec{x}_\ell)_{\mathbb{B}_\ell}, (\vec{x}_\tau)_{\mathbb{B}_\tau}) \\
&\qquad\qquad := \textstyle\sum_{t=0,\ldots,\ell,\tau} (\vec{x}_t)_{\mathbb{B}_t} \in \mathbb{V}, \\
&e(\boldsymbol{c}, \boldsymbol{k}^*) := \textstyle\prod_{t=0}^{d} e(\boldsymbol{c}_t, \boldsymbol{k}_t^*) \quad \text{where } \boldsymbol{c} := (\boldsymbol{c}_0, \ldots, \boldsymbol{c}_d) \in \mathbb{V}_0 \oplus \cdots \oplus \mathbb{V}_d, \\
&\qquad\qquad \boldsymbol{k}^* := (\boldsymbol{k}_0^*, \ldots, \boldsymbol{k}_d^*) \in \mathbb{V}_0^* \oplus \cdots \oplus \mathbb{V}_d^*, \\
&\text{and } \vec{e}_{t,j} := (\overbrace{0, \ldots, 0}^{j-1}, 1, \overbrace{0, \ldots, 0}^{n_t - j}) \in \mathbb{F}_q^{n_t},
\end{aligned}$$

and all the above notations are applied to the case with $\{\mathbb{B}_t^*\}_{t=0,\ldots,d}$ instead of $\{\mathbb{B}_t\}_{t=0,\ldots,d}$

G.3. *Efficient Payload-Hiding HIPE Scheme*

G.3.1. *Construction*

$\mathsf{Setup}(1^\lambda,\ \vec{n} := (d; n_1, \ldots, n_d)) :\quad (\mathsf{param}_{\vec{n}}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0,\ldots,d}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}(1^\lambda, \vec{n}),$

$\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5}),\ \widehat{\mathbb{B}}_t := (\boldsymbol{b}_{t,1}, .., \boldsymbol{b}_{t,n_t}, \boldsymbol{b}_{t,3n_t+1})$ for $t = 1, .., d,$

$\widehat{\mathbb{B}}_0^* := (\boldsymbol{b}_{0,1}^*, \boldsymbol{b}_{0,4}^*),\ \widehat{\mathbb{B}}_t^* := (\boldsymbol{b}_{t,1}^*, .., \boldsymbol{b}_{t,n_t}^*, \boldsymbol{b}_{t,2n_t+1}^*, .., \boldsymbol{b}_{t,3n_t}^*)$ for $t = 1, .., d,$

return $\mathsf{pk} := (1^\lambda, \mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}_t, \widehat{\mathbb{B}}_t^*\}_{t=0,\ldots,d}),\ \mathsf{sk} := \boldsymbol{b}_{0,3}^*.$

$\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, (\vec{v}_1, \ldots, \vec{v}_\ell) \in \mathbb{F}_q^{n_1} \times \cdots \times \mathbb{F}_q^{n_\ell}) :$

$s_t, \theta_t \xleftarrow{\cup} \mathbb{F}_q$ for $t = 1, \ldots, \ell$, $s_0 := \sum_{t=1}^{\ell} s_t$, $\vec{\eta}_t \xleftarrow{\cup} \mathbb{F}_q^{n_t}$ for $t = 0, \ldots, \ell$,

$\boldsymbol{k}_\ell^* := ((-s_0,\ 0,\ 1,\ \eta_0,\ 0)_{\mathbb{B}_0^*},\ (s_t\vec{e}_{t,1} + \theta_t\vec{v}_t,\ 0^{n_t},\ \vec{\eta}_t,\ 0)_{\mathbb{B}_t^*} : t = 1, \ldots, \ell)$,

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

$\mathsf{Enc}(\mathsf{pk}, m \in \mathbb{G}_T, (\vec{x}_1, \ldots, \vec{x}_\ell) \in \mathbb{F}_q^{n_1} \times \cdots \times \mathbb{F}_q^{n_\ell}) :$

$\omega, \varphi_0, \ldots, \varphi_\ell \xleftarrow{\cup} \mathbb{F}_q$, $\boldsymbol{c}_1 := ((\omega, 0, \zeta, 0, \varphi_0)_{\mathbb{B}_0},\ (\omega\vec{x}_t, 0^{n_t}, 0^{n_t}, \varphi_t)_{\mathbb{B}_t} : t = 1, \ldots, \ell)$,

$c_2 := g_T^{\zeta} m$, $\mathsf{ct} := (\boldsymbol{c}_1, c_2)$, return ct.

$\mathsf{Dec}(\mathsf{pk}, \boldsymbol{k}_{\ell,\mathsf{dec}}^*, \mathsf{ct}) :\ m' := c_2 / e(\boldsymbol{c}_1, \boldsymbol{k}_{\ell,\mathsf{dec}}^*)$, return m'.

$\mathsf{Delegate}_\ell(\mathsf{pk}, \mathsf{sk}_\ell, \vec{v}_{\ell+1} \in \mathbb{F}_q^{n_{\ell+1}}) :$

$s_{\mathsf{del},t}, \theta_{\mathsf{del},t} \xleftarrow{\cup} \mathbb{F}_q$ for $t = 1, \ldots, \ell+1$, $s_{\mathsf{del},0} := \sum_{t=1}^{\ell+1} s_{\mathsf{del},t}$,

$\vec{\eta}_{\mathsf{del},t} \xleftarrow{\cup} \mathbb{F}_q^{n_t}$ for $t = 0, \ldots, \ell+1$,

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

$\quad (s_{\mathsf{del},t}\vec{e}_{t,1} + \theta_{\mathsf{del},t}\vec{v}_t,\ 0^{n_t},\ \vec{\eta}_{\mathsf{del},t},\ 0)_{\mathbb{B}_t^*} : t = 1, \ldots, \ell+1)$,

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

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

G.3.2. *Security*

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., [25], 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 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,

$$\mathsf{Adv}_{\mathcal{A}}^{\mathit{HIPE,\ PH}}(\lambda) \le \mathit{Adv}_{\mathcal{E}_1}^{\mathsf{DLIN}}(\lambda) + \sum_{h=1}^{\nu} \mathit{Adv}_{\mathcal{E}_{2,h}}^{\mathsf{DLIN}}(\lambda) + \epsilon,$$

where $\mathcal{E}_{2,h}(\cdot) := \mathcal{E}_2(h, \cdot)$, ν is the maximum number of adversary $\mathcal{A}$'s key queries, and $\epsilon = (d\nu + 8\nu + d + 7)/q$.

Proof Outline of Theorem 6: To prove Theorem 6, 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 which were changed in a game from the previous game.

Game 0 : Original game. That is, the reply to a key query consists of:

$$\boldsymbol{k}^*_\ell := (\,(\,-s_0,\ \boxed{0},\ 1,\ \eta_0,\ 0\,)_{\mathbb{B}^*_0},\ (\,s_t\vec{e}_{t,1}+\theta_t\vec{v}_t,\ \boxed{0^{n_t}},\ \vec{\eta}_t,\ 0\,)_{\mathbb{B}^*_t} : t=1,\ldots,\ell).$$

The challenge ciphertext consists of:

$$\boldsymbol{c}_1 := (\,(\omega, \boxed{0}, \boxed{\zeta}, 0, \varphi_0)_{\mathbb{B}_0},\ (\omega\vec{x}_t, \boxed{0^{n_t}}, 0^{n_t}, \varphi_t)_{\mathbb{B}_t} : t=1,\ldots,\ell\,), \qquad c_2 := g_T^{\zeta} m.$$

Game 1 : Game 1 is the same as Game 0 except the following procedures.

1. When a create key query is issued by $\mathcal{A}$, the challenger of the game only records the specified predicates, and when a create delegated key query is issued, the challenger only records the specified keys and predicates. In this step, just the query is recorded, but no corresponding key is created.
2. When a reveal key query is issued for a hierarchical (level-ℓ) predicate $(\vec{v}_1,\ldots,\vec{v}_\ell)$ which has been already recorded, the challenger creates the queried key by using KeyGen.

Game 2 : Same as Game 1 except that the challenge ciphertext is:

$$\boldsymbol{c}_1 := (\,(\omega, \boxed{w_0}, \zeta, 0, \varphi_0)_{\mathbb{B}_0},\ (\omega\vec{x}_t, \boxed{\vec{w}_t}, 0^{n_t}, \varphi_t)_{\mathbb{B}_t} : t=1,\ldots,\ell\,), \qquad c_2 := g_T^{\zeta} m,$$

where $w_0 \xleftarrow{\cup} \mathbb{F}_q$, $\vec{w}_t \xleftarrow{\cup} \mathbb{F}_q^{n_t}$.

Game 3-h ($\boldsymbol{h} = \mathbf{1},\ldots,\boldsymbol{\nu}$) : Game 3-0 is Game 2. Game 3-h is the same as Game 3-$(h-1)$ except that the hth reveal key query's reply, $\boldsymbol{k}^*_\ell$, is:

$$\boldsymbol{k}^*_\ell := (\,(\,-s_0,\ \boxed{r_0},\ 1,\ \eta_0,\ 0\,)_{\mathbb{B}^*_0},\ (\,s_t\vec{e}_{t,1}+\theta_t\vec{v}_t,\ \boxed{\vec{r}_t},\ \vec{\eta}_t,\ 0\,)_{\mathbb{B}^*_t} : t=1,\ldots,\ell),$$

where $r_0 \xleftarrow{\cup} \mathbb{F}_q, \vec{r}_t \xleftarrow{\cup} \mathbb{F}_q^{n_t}$ for $t=1,\ldots,\ell$, and the other variables are generated as in Game 3-$(h-1)$.

Game 4 : Game 4 is the same as Game 3-ν except that the challenge ciphertext is:

$$\boldsymbol{c}_1 := (\,(\omega, w_0, \boxed{\zeta'}, 0, \varphi_0)_{\mathbb{B}_0},\ (\omega\vec{x}_t, \vec{w}_t, 0^{n_t}, \varphi_t)_{\mathbb{B}_t} : t=1,\ldots,\ell\,), \qquad c_2 := g_T^{\zeta} m,$$

where $\zeta, \zeta' \xleftarrow{\cup} \mathbb{F}_q$.

Let $\mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda)$, $\mathsf{Adv}^{(1)}_{\mathcal{A}}(\lambda)$,$\mathsf{Adv}^{(2)}_{\mathcal{A}}(\lambda)$, $\mathsf{Adv}^{(3\text{-}h)}_{\mathcal{A}}(\lambda)$ and $\mathsf{Adv}^{(4)}_{\mathcal{A}}(\lambda)$ be the advantage of $\mathcal{A}$ in Game 0, Game 1, Game 2, Game 3-h and Game 4. It is obtained that $\mathsf{Adv}^{(4)}_{\mathcal{A}}(\lambda) = 0$.

We can evaluate the gaps between pairs of the above advantages using Problems 1 and 2 as in the proof of Theorem 1. □

G.4. *Attribute-Hiding HIPE Scheme*

G.4.1. *Construction*

$\mathsf{Setup}(1^\lambda,\ \vec{n} := (d; n_1, \ldots, n_d)) :\quad (\mathsf{param}_{\vec{n}}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0,\ldots,d}) \xleftarrow{\mathsf{R}} \mathcal{G}_{\mathsf{ob}}(1^\lambda, \vec{n}),$

$\widehat{\mathbb{B}}_0 := (\boldsymbol{b}_{0,1}, \boldsymbol{b}_{0,3}, \boldsymbol{b}_{0,5}),\ \ \widehat{\mathbb{B}}_t := (\boldsymbol{b}_{t,1}, .., \boldsymbol{b}_{t,n_t}, \boldsymbol{b}_{t,3n_t+1})\ \ \text{for } t = 1, .., d,$

$\widehat{\mathbb{B}}_0^* := (\boldsymbol{b}_{0,1}^*, \boldsymbol{b}_{0,3}^*),\ \ \widehat{\mathbb{B}}_t^* := (\boldsymbol{b}_{t,1}^*, .., \boldsymbol{b}_{t,n_t}^*)\ \ \text{for } t = 1, .., d,$

return $\mathsf{pk} := (1^\lambda, \mathsf{param}_{\vec{n}}, \{\widehat{\mathbb{B}}_t\}_{t=0,\ldots,d}, \boldsymbol{b}_{0,4}^*, \{\boldsymbol{b}_{t,2n_t+1}^*, .., \boldsymbol{b}_{t,3n_t}^*\}_{t=1,\ldots,d}), \mathsf{sk} := \{\widehat{\mathbb{B}}_t^*\}_{t=0,\ldots,d}.$

$\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, (\vec{v}_1, \ldots, \vec{v}_\ell) \in \mathbb{F}_q^{n_1} \times \cdots \times \mathbb{F}_q^{n_\ell}) :$

for $j = 1, \ldots, 2\ell;\ \tau = \ell+1, \ldots, d;\ \iota = 1, \ldots, n_\tau;$

$\psi, s_{\mathsf{dec},t}, s_{\mathsf{ran},1,j,t}, \theta_{\mathsf{dec},t}, \theta_{\mathsf{ran},1,j,t} \xleftarrow{\mathsf{U}} \mathbb{F}_q\ \ \text{for } t = 1, \ldots, \ell,$

$s_{\mathsf{del},(\tau,\iota),t}, s_{\mathsf{ran},2,\tau,t}, \theta_{\mathsf{del},(\tau,\iota),t}, \theta_{\mathsf{ran},2,\tau,t} \xleftarrow{\mathsf{U}} \mathbb{F}_q\ \ \text{for } t = 1, \ldots, \ell+1,$

$s_{\mathsf{dec},0} := \sum_{t=1}^{\ell} s_{\mathsf{dec},t},\ \ s_{\mathsf{del},(\tau,\iota),0} := \sum_{t=1}^{\ell+1} s_{\mathsf{del},(\tau,\iota),t},$

$s_{\mathsf{ran},1,j,0} := \sum_{t=1}^{\ell} s_{\mathsf{ran},1,j,t},\ \ s_{\mathsf{ran},2,\tau,0} := \sum_{t=1}^{\ell+1} s_{\mathsf{ran},2,\tau,t},$

$\vec{\eta}_{\mathsf{dec},t},\ \vec{\eta}_{\mathsf{ran},1,j,t} \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_t}\ \ \text{for } t = 0, \ldots, \ell,$

$\vec{\eta}_{\mathsf{del},(\tau,\iota),t},\ \vec{\eta}_{\mathsf{ran},2,\tau,t} \xleftarrow{\mathsf{U}} \mathbb{F}_q^{n_t},\ \ \text{for } t = 0, \ldots, \ell+1,$

$\boldsymbol{k}_{\ell,\mathsf{dec}}^* :=\quad (\,(\,-s_{\mathsf{dec},0},\ 0,\ 1,\ \eta_{\mathsf{dec},0},\ 0\,)_{\mathbb{B}_0^*},$

$(\,s_{\mathsf{dec},t}\vec{e}_{t,1} + \theta_{\mathsf{dec},t}\vec{v}_t,\quad 0^{n_t},\quad \vec{\eta}_{\mathsf{dec},t},\ 0\,)_{\mathbb{B}_t^*} : t = 1, \ldots, \ell),$

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

$(\,s_{\mathsf{del},(\tau,\iota),t}\vec{e}_{t,1} + \theta_{\mathsf{del},(\tau,\iota),t}\vec{v}_t,\ 0^{n_t},\quad \vec{\eta}_{\mathsf{del},(\tau,\iota),t},\ 0\,)_{\mathbb{B}_t^*} : t = 1, \ldots, \ell,$

$(\,s_{\mathsf{del},(\tau,\iota),\ell+1}\vec{e}_{\tau,1} + \psi\vec{e}_{\tau,\iota},\quad 0^{n_\tau},\quad \vec{\eta}_{\mathsf{del},(\tau,\iota),\ell+1},\ 0\,)_{\mathbb{B}_\tau^*}),$

$\boldsymbol{k}_{\ell,\mathsf{ran},1,j}^* := (\,(\,-s_{\mathsf{ran},1,j,0},\ 0,\ 0,\ \eta_{\mathsf{ran},1,j,0},\ 0\,)_{\mathbb{B}_0^*},$

$(\,s_{\mathsf{ran},1,j,t}\vec{e}_{t,1} + \theta_{\mathsf{ran},1,j,t}\vec{v}_t,\quad 0^{n_t},\quad \vec{\eta}_{\mathsf{ran},1,j,t},\ 0\,)_{\mathbb{B}_t^*} : t = 1, \ldots, \ell),$

$\boldsymbol{k}_{\ell,\mathsf{ran},2,\tau}^* := (\,(\,-s_{\mathsf{ran},2,\tau,0},\ 0,\ 0,\ \eta_{\mathsf{ran},2,\tau,0},\ 0\,)_{\mathbb{B}_0^*},$

$(\,s_{\mathsf{ran},2,\tau,t}\vec{e}_{t,1} + \theta_{\mathsf{ran},2,\tau,t}\vec{v}_t,\quad 0^{n_t},\quad \vec{\eta}_{\mathsf{ran},2,\tau,t},\ 0\,)_{\mathbb{B}_t^*} : t = 1, \ldots, \ell,$

$(\,s_{\mathsf{ran},2,\tau,\ell+1}\vec{e}_{\tau,1},\quad 0^{n_\tau},\quad \vec{\eta}_{\mathsf{ran},2,\tau,\ell+1},\ 0\,)_{\mathbb{B}_\tau^*}),$

$\mathsf{sk}_\ell := (\boldsymbol{k}_{\ell,\mathsf{dec}}^*, \{\boldsymbol{k}_{\ell,\mathsf{del},(\tau,\iota)}^*\}_{\tau=\ell+1,\ldots,d;\ \iota=1,\ldots,n_\tau}, \{\boldsymbol{k}_{\ell,\mathsf{ran},1,j}^*,\ \boldsymbol{k}_{\ell,\mathsf{ran},2,\tau}^*\}_{j=1,\ldots,2\ell;\ \tau=\ell+1,\ldots,d}),$

return $\mathsf{sk}_\ell.$

$\mathsf{Enc}(\mathsf{pk}, m \in \mathbb{G}_T, (\vec{x}_1, \ldots, \vec{x}_\ell) \in \mathbb{F}_q^{n_1} \times \cdots \times \mathbb{F}_q^{n_\ell}) :$

$\omega, \varphi_0, \ldots, \varphi_\ell \xleftarrow{\mathsf{U}} \mathbb{F}_q,\ \ \boldsymbol{c}_1 := (\,(\omega, 0, \zeta, 0, \varphi_0)_{\mathbb{B}_0}, (\omega\vec{x}_t, 0^{n_t}, 0^{n_t}, \varphi_t)_{\mathbb{B}_t} : t = 1, \ldots, \ell),$

$c_2 := g_T^\zeta m,\quad \mathsf{ct} := (\boldsymbol{c}_1, c_2),\quad \text{return ct.}$

$\mathsf{Dec}(\mathsf{pk}, \boldsymbol{k}_{\ell,\mathsf{dec}}^*, \mathsf{ct}) :\ m' := c_2/e(\boldsymbol{c}_1, \boldsymbol{k}_{\ell,\mathsf{dec}}^*),\quad \text{return } m'.$

$\mathsf{Delegate}_\ell(\mathsf{pk}, \mathsf{sk}_\ell, \vec{v}_{\ell+1} := (v_{\ell+1,1}, \ldots, v_{\ell+1,n_{\ell+1}})) :$

for $j' = 1, \ldots, 2(\ell+1);\ \tau = \ell+2, \ldots, d;\ \iota = 1, \ldots, n_\tau;$

$\phi_{\mathsf{del},(\tau,\iota)},\ \phi_{\mathsf{ran},2,\tau},\ \psi' \xleftarrow{\mathsf{U}} \mathbb{F}_q,$

$\boldsymbol{p}_{\mathsf{dec}}^*, \boldsymbol{p}_{\mathsf{del},(\tau,\iota)}^*, \boldsymbol{p}_{\mathsf{ran},1,j'}^*, \boldsymbol{p}_{\mathsf{ran},2,\tau}^* \xleftarrow{\mathsf{R}} \mathsf{CoreDel}_\ell(\mathsf{pk}, \mathsf{sk}_\ell, \vec{v}_{\ell+1}),$

where $\mathsf{CoreDel}_\ell(\mathsf{pk}, \mathsf{sk}_\ell, \vec{v}_{\ell+1}) :\ \sigma, \alpha_j \xleftarrow{\mathsf{U}} \mathbb{F}_q\ \text{for } j = 1, \ldots, 2\ell+1,$

return $\boldsymbol{p}^* := \sigma(\sum_{i=1}^{n_{\ell+1}} v_{\ell+1,i}\boldsymbol{k}_{\ell,\mathsf{del},(\ell+1,i)}^*) + \sum_{j=1}^{2\ell} \alpha_j \boldsymbol{k}_{\ell,\mathsf{ran},1,j}^* + \alpha_{2\ell+1}\boldsymbol{k}_{\ell,\mathsf{ran},2,\ell+1}^*,$

$\boldsymbol{r}_{\mathsf{dec}}^*, \boldsymbol{r}_{\mathsf{ran},1,j'}^* \xleftarrow{\mathsf{U}} \mathsf{span}\langle \boldsymbol{b}_{0,4}^*, \{\boldsymbol{b}_{t,2n_t+i}^*\}_{t=1,\ldots,\ell+1;\ i=1,\ldots,n_t}\rangle,$

$$\boldsymbol{r}^*_{\mathsf{del},(\tau,\iota)}, \boldsymbol{r}^*_{\mathsf{ran},2,\tau} \xleftarrow{\cup} \mathsf{span}\langle \boldsymbol{b}^*_{0,4}, \{\boldsymbol{b}^*_{t,2n_t+i}\}_{t=1,\ldots,\ell+1,\tau;\ i=1,\ldots,n_t}\rangle,$$
$$\boldsymbol{k}^*_{\ell+1,\mathsf{dec}} := \boldsymbol{k}^*_{\ell,\mathsf{dec}} + \boldsymbol{p}^*_{\mathsf{dec}} + \boldsymbol{r}^*_{\mathsf{dec}},$$
$$\boldsymbol{k}^*_{\ell+1,\mathsf{del},(\tau,\iota)} := \boldsymbol{p}^*_{\mathsf{del},(\tau,\iota)} + \phi_{\mathsf{del},(\tau,\iota)}\boldsymbol{k}^*_{\ell,\mathsf{ran},2,\tau} + \psi'\boldsymbol{k}^*_{\ell,\mathsf{del},(\tau,\iota)} + \boldsymbol{r}^*_{\mathsf{del},(\tau,\iota)},$$
$$\boldsymbol{k}^*_{\ell+1,\mathsf{ran},1,j'} := \boldsymbol{p}^*_{\mathsf{ran},1,j'} + \boldsymbol{r}^*_{\mathsf{ran},1,j'},$$
$$\boldsymbol{k}^*_{\ell+1,\mathsf{ran},2,\tau} := \boldsymbol{p}^*_{\mathsf{ran},2,\tau} + \phi_{\mathsf{ran},2,\tau}\boldsymbol{k}^*_{\ell,\mathsf{ran},2,\tau} + \boldsymbol{r}^*_{\mathsf{ran},2,\tau},$$
$$\mathsf{sk}_{\ell+1} := (\boldsymbol{k}^*_{\ell+1,\mathsf{dec}}, \{\boldsymbol{k}^*_{\ell+1,\mathsf{del},(\tau,\iota)}\}_{\tau=\ell+2,\ldots,d;\ \iota=1,\ldots,n_\tau}, \{\boldsymbol{k}^*_{\ell,\mathsf{ran},1,j'},\ \boldsymbol{k}^*_{\ell,\mathsf{ran},2,\tau}\}_{j'=1,\ldots,2(\ell+1);\ \tau=\ell+2,\ldots,d}),$$
return $\mathsf{sk}_{\ell+1}$.

G.4.2. *Equivalence of Delegated and Freshly Generated Keys*

Lemma 29. *If* sk_ℓ *is generated by* $\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, (\vec{v}_1, \ldots, \vec{v}_\ell))$*, the distribution of* $\mathsf{sk}_{\ell+1}$ *generated by* $\mathsf{Delegate}(\mathsf{pk}, \mathsf{sk}_\ell, \vec{v}_{\ell+1})$ *is equivalent to that of* $\mathsf{sk}_{\ell+1}$ *generated by* $\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, (\vec{v}_1, \ldots, \vec{v}_\ell, \vec{v}_{\ell+1}))$ *except with probability at most* $(2d - 2\ell + 3)/q$.

Proof. The distribution of (a part of) level-ℓ key $\boldsymbol{k}^*_{\ell,J}$ for $J = \mathsf{dec}, (\mathsf{ran}, 1, 1), \ldots, (\mathsf{ran}, 1, 2\ell)$ is represented by that of the 2ℓ coefficients, $(s_{J,1}, \ldots, s_{J,\ell}, \theta_{J,1}, \ldots, \theta_{J,\ell})$, of $(\vec{e}_{1,1}, \ldots, \vec{e}_{\ell,1}, \vec{v}_1, \ldots, \vec{v}_\ell)$ (and random-part coefficients, $\vec{\eta}_{J,t}$). The distribution of level-ℓ key $\boldsymbol{k}^*_{\ell,J}$ for $J = (\mathsf{del}, (\ell+1, 1)), \ldots, (\mathsf{del}, (d, n_d))$ (resp. $J = (\mathsf{ran}, 2, \ell+1), \ldots, (\mathsf{ran}, 2, d)$) is represented by that of the $2\ell+2$ (resp. $2\ell+1$) coefficients, $(s_{J,1}, \ldots, s_{J,\ell}, s_{J,\tau}, \theta_{J,1}, \ldots, \theta_{J,\ell}, \psi)$, of $(\vec{e}_{1,1}, \ldots, \vec{e}_{\ell,1}, \vec{e}_{\tau,1}, \vec{v}_1, \ldots, \vec{v}_\ell, \vec{e}_{\tau,\iota})$ (resp. $(s_{J,1}, \ldots, s_{J,\ell}, s_{J,\tau}, \theta_{J,1}, \ldots, \theta_{J,\ell})$, of $(\vec{e}_{1,1}, \ldots, \vec{e}_{\ell,1}, \vec{e}_{\tau,1}, \vec{v}_1, \ldots, \vec{v}_\ell)$) (and random-part coefficients, $\vec{\eta}_{J,t}$).

Similarly, the distribution of level-$(\ell+1)$ key $\boldsymbol{k}^*_{\ell+1,J}$ is represented by that of the $2(\ell+1), 2(\ell+1)+2$ or $2(\ell+1)+1$ coefficients, $\vec{y}_J := (s_{J,1}, \ldots, s_{J,\ell}, \theta_{J,1}, \ldots, \theta_{J,\ell})$, $(s_{J,1}, \ldots, s_{J,\ell}, s_{J,\tau}, \theta_{J,1}, \ldots, \theta_{J,\ell}, \psi)$, or $(s_{J,1}, \ldots, s_{J,\ell}, s_{J,\tau}, \theta_{J,1}, \ldots, \theta_{J,\ell})$.

Claim 3 shows the coefficients of delegated key is uniformly distributed in the first case.

□

Claim 3. *If* sk_ℓ *is generated by* $\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, (\vec{v}_1, \ldots, \vec{v}_\ell))$*, the distribution of* $\boldsymbol{k}^*_{\ell+1,\mathsf{ran},1,j'}$ *generated in* $\mathsf{Delegate}(\mathsf{pk}, \mathsf{sk}_\ell, \vec{v}_{\ell+1})$ *is equivalent to that of* $\boldsymbol{k}^*_{\ell+1,\mathsf{ran},1,j'}$ *generated by* $\mathsf{KeyGen}(\mathsf{pk}, \mathsf{sk}, (\vec{v}_1, \ldots, \vec{v}_\ell, \vec{v}_{\ell+1}))$ *except with probability at most* $3/q$.

Proof of Claim 3. The distribution of $\boldsymbol{k}^*_{\ell,J}$ ($J = (\mathsf{del}, (\tau, \iota)), (\mathsf{ran}, 1, j), (\mathsf{ran}, 2, \tau)$) in sk_ℓ is represented by $2\ell+1$-dimensional vectors as (except for ψ):

$$\begin{aligned}\vec{y}_{\ell,J} &:= (s_{\ell,J,1}, \ldots, s_{\ell,J,\ell}, s_{\ell,J,\tau}, \theta_{\ell,J,1}, \ldots, \theta_{\ell,J,\ell}) \text{ if } J = (\mathsf{del}, (\tau, \iota)),\\ &:= (s_{\ell,J,1}, \ldots, s_{\ell,J,\ell}, 0, \theta_{\ell,J,1}, \ldots, \theta_{\ell,J,\ell}) \text{ if } J = (\mathsf{ran}, 1, j),\\ &:= (s_{\ell,J,1}, \ldots, s_{\ell,J,\ell}, s_{\ell,J,\tau}, \theta_{\ell,J,1}, \ldots, \theta_{\ell,J,\ell}) \text{ if } J = (\mathsf{ran}, 2, \tau).\end{aligned}$$

The coefficients $\vec{y}_{\ell+1,\mathsf{ran},1,j'}$ of $\boldsymbol{k}^*_{\ell+1,\mathsf{ran},1,j'}$ except for that of $\vec{v}_{\ell+1}$ are given as:

$$\begin{aligned}\vec{y}_{\ell+1,\mathsf{ran},1,j'} &:= (s_{\ell+1,\mathsf{ran},1,j',1}, \ldots, s_{\ell+1,\mathsf{ran},1,j',\ell+1}, \theta_{\ell+1,\mathsf{ran},1,j',1}, \ldots, \theta_{\ell+1,\mathsf{ran},1,j',\ell})\\ &= \sigma_{\mathsf{ran},1,j'} \textstyle\sum_{i=1}^{n_{\ell+1}} v_{\ell+1,i}\vec{y}_{\ell,\mathsf{del},(\ell+1,i)} + \sum_{j=1}^{2\ell} \alpha_{\mathsf{ran},1,j',j}\vec{y}_{\ell,\mathsf{ran},1,j} + \alpha_{\mathsf{ran},1,j',2\ell+1}\vec{y}_{\ell,\mathsf{ran},2,\ell+1}\\ &= \sigma_{\mathsf{ran},1,j'} \textstyle\sum_{i=1}^{n_{\ell+1}} v_{\ell+1,i}\vec{y}_{\ell,\mathsf{del},(\ell+1,i)} + \vec{\alpha}_{\mathsf{ran},1,j'} \cdot Y_{\ell,\mathsf{ran}} \in \mathbb{F}_q^{2\ell+1},\end{aligned}$$

$$\text{where } \vec{\alpha}_{\mathsf{ran},1,j'} := (\alpha_{\mathsf{ran},1,j',1}, \ldots, \alpha_{\mathsf{ran},1,j',2\ell}, \alpha_{\mathsf{ran},1,j',2\ell+1}) \xleftarrow{\cup} \mathbb{F}_q^{2\ell+1},$$

$$Y_{\ell,\mathsf{ran}} := \begin{pmatrix} \vec{y}_{\ell,\mathsf{ran},1,1} \\ \vdots \\ \vec{y}_{\ell,\mathsf{ran},1,2\ell} \\ \vec{y}_{\ell,\mathsf{ran},2,\ell+1} \end{pmatrix} \in \mathbb{F}_q^{(2\ell+1)\times(2\ell+1)}.$$

Moreover, the coefficient of $\vec{v}_{\ell+1}$ in $\boldsymbol{k}^{\mathsf{ran},1,j'}$ of level-$(\ell+1)$ is given by $\theta_{\ell+1,\mathsf{ran},1,j',\ell+1} := \sigma_{\mathsf{ran},1,j'} \cdot \psi$, where ψ is given in the level-ℓ key and $\sigma_{\mathsf{ran},1,j'}$ is generated in $\mathsf{CoreDel}_\ell$ in the delegation.

We consider the joint distribution of $\vec{y}_{\ell+1,\mathsf{ran},1,j'}$ and $\theta_{\ell+1,\mathsf{ran},1,j',\ell+1}$, i.e., $\{s_{\ell+1,\mathsf{ran},1,j',t}, \theta_{\ell+1,\mathsf{ran},1,j',t}\}_{t=1,\ldots,\ell+1}$.

If the matrix $Y_{\ell,\mathsf{ran}}$ is regular and $\psi \neq 0$, since $\vec{\alpha}_{\mathsf{ran},1,j'} \xleftarrow{\cup} \mathbb{F}_q^{2\ell+1}$, $\sigma_{\mathsf{ran},1,j'} \xleftarrow{\cup} \mathbb{F}_q$, and variables $\vec{\alpha}_{\mathsf{ran},1,j'} \cdot Y_{\ell,\mathsf{ran}}$ and $\sigma_{\mathsf{ran},1,j'} \cdot \psi$ are independent, $(\vec{y}_{\ell+1,\mathsf{ran},1,j'}, \theta_{\ell+1,\mathsf{ran},1,j',\ell+1}) \in \mathbb{F}_q^{2(\ell+1)}$ for $j' = 1, \ldots, 2(\ell+1)$ are uniformly and independently distributed in $\mathbb{F}_q^{2(\ell+1)}$.

Here, $Y_{\ell,\mathsf{ran}}$ ($(2\ell+1) \times (2\ell+1)$ matrix) of sk_ℓ is regular and $\psi \neq 0$ except with probability at most $2/q + 1/q = 3/q$, from Claim 4. □

Since $\boldsymbol{k}^*_{\ell+1,\mathsf{ran},1,j'} + \boldsymbol{b}^*_{0,3}$ $(j' = 1, \ldots, 2(\ell+1))$ has the same distribution as $\boldsymbol{k}^*_{\ell+1,\mathsf{dec}}$, Lemma 29 holds for $\boldsymbol{k}^*_{\ell+1,\mathsf{dec}}$ from Claim 3.

For $\boldsymbol{k}^*_{\ell+1,\mathsf{ran},2,\tau}$ $(\tau = \ell+2, \ldots, d)$, the level-$(\ell+1)$ coefficient $s_{\ell+1,\mathsf{ran},2,\tau,\ell+1}$ of $\vec{e}_{\tau,1}$ is given by $\phi_{\mathsf{ran},2,\tau} \cdot s_{\ell,\mathsf{ran},2,\tau,\ell+1}$ where $\phi_{\mathsf{ran},2,\tau}$ is generated in $\mathsf{Delegate}_\ell$ and $s_{\ell,\mathsf{ran},2,\tau,\ell+1}$ the level-ℓ coefficient of $\vec{e}_{\tau,1}$. Therefore, Lemma 29 holds for $\boldsymbol{k}^*_{\ell+1,\mathsf{ran},2,\tau}$ from Claim 3 except for negligible probability, i.e., at most $(d-\ell)/q$.

Since $\boldsymbol{k}^*_{\ell+1,\mathsf{ran},2,\tau} + \psi \boldsymbol{b}^*_{\tau,\iota}$ $(\tau = \ell+2, \ldots, d; \iota = 1, \ldots, n_\tau)$ has the same distribution as $\boldsymbol{k}^*_{\ell+1,\mathsf{del},(\tau,\iota)}$, Lemma 29 holds for $\boldsymbol{k}^*_{\ell+1,\mathsf{del},(\tau,\iota)}$ from Claim 3 except for negligible probability, i.e., at most $(d-\ell+1)/q$.

Therefore, Lemma 29 holds except for negligible probability, i.e., at most $(2d-2\ell+3)/q$.

□

Claim 4. (Claim 4 in [32]) *Let* $q > 2$ *and* $\Delta := \{M \mid \det M \neq 0\} \subset \mathbb{F}_q^{l\times l}$. *Then,* $\frac{|\Delta|}{q^{l^2}} < \frac{2}{q}$.

G.4.3. *Security*

The definition of *adaptively weakly attribute-hiding* security and the advantage $\mathsf{Adv}^{\mathsf{HIPE,\,wAH}}_{\mathcal{A}}(\lambda)$ of adversary $\mathcal{A}$ are shown in Definition 47 of the full version of [32]. In the definition, the levels ℓ and ℓ' of the two challenge vectors given by an adversary, $(\vec{x}_i^{(0)})_{i=1,\ldots,\ell}$ and $(\vec{x}_i^{(1)})_{i=1,\ldots,\ell'}$, can be different, i.e., $\ell \neq \ell'$ is allowed. The proposed HIPE scheme only satisfies the security definition under the restriction that $\ell = \ell'$. Here, this restricted security ensures the anonymity of attributes of a ciphertext but with revealing the number of levels of attributes, while the security definition in [32] ensures the anonymity of attributes as well as the number of levels. (The HIPE scheme in [32]

satisfies the unrestricted security.) Our scheme can be modified to satisfy the unrestricted security in [32] as: when generating a ciphertext in Enc, input vectors $(\vec{x}_i)_{i=1,\ldots,\ell}$ are padded with random vectors $(\vec{x}_i)_{i=\ell+1,\ldots,d}$ for a maximum level d, in the same manner as the HIPE in [32].

Theorem 7. *The proposed HIPE scheme is adaptively weakly attribute-hiding against chosen-plaintext attacks under the DLIN assumption.*

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,

$$\mathsf{Adv}^{\mathit{HIPE,\ wAH}}_{\mathcal{A}}(\lambda) < \mathit{Adv}^{\mathsf{DLIN}}_{\mathcal{E}_1}(\lambda) + \sum_{h=1}^{\nu}\sum_{I=1}^{L} \mathsf{Adv}^{\mathit{DLIN}}_{\mathcal{E}_{2,(h,I)}}(\lambda) + \epsilon,$$

where $\mathcal{E}_{2,(h,I)}(\cdot) := \mathcal{E}_2((h,I),\cdot)$ $(h = 1,\ldots,\nu;\ I = 1,\ldots,L)$, ν is the maximum number of adversary $\mathcal{A}$'s key queries, $L := d + 2 + \sum_{\tau=2}^{d} n_\tau$, and $\epsilon = ((d+8)L\nu + 3d + 8)/q$.

Proof Outline of Theorem 7: To prove Theorem 7, we consider the following $(L\nu+3)$ 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 which were changed in a game from the previous game.

Game 0 : Original game. That is, the reply to a key query consists of:

$$\begin{aligned}
\boldsymbol{k}^*_{\ell,\mathsf{dec}} := \quad & (\,(-s_{\mathsf{dec},0},\ \boxed{0},\ 1,\ \eta_{\mathsf{dec},0},\ 0\,)_{\mathbb{B}^*_0},\\
& (\,s_{\mathsf{dec},t}\vec{e}_{t,1} + \theta_{\mathsf{dec},t}\vec{v}_t,\quad \boxed{0^{n_t}},\quad \vec{\eta}_{\mathsf{dec},t},\ 0\,)_{\mathbb{B}^*_t} : t = 1,\ldots,\ell),\\
\boldsymbol{k}^*_{\ell,\mathsf{del},(\tau,\iota)} := \ & (\,(-s_{\mathsf{del},(\tau,\iota),0},\ \boxed{0},\ 0,\ \eta_{\mathsf{del},(\tau,\iota),0},\ 0\,)_{\mathbb{B}^*_0},\\
& (\,s_{\mathsf{del},(\tau,\iota),t}\vec{e}_{t,1} + \theta_{\mathsf{del},(\tau,\iota),t}\vec{v}_t,\ \boxed{0^{n_t}},\quad \vec{\eta}_{\mathsf{del},(\tau,\iota),t},\ 0\,)_{\mathbb{B}^*_t} : t = 1,\ldots,\ell,\\
& (\,s_{\mathsf{del},(\tau,\iota),\ell+1}\vec{e}_{\tau,1} + \psi\vec{e}_{\tau,\iota},\quad \boxed{0^{n_\tau}},\quad \vec{\eta}_{\mathsf{del},(\tau,\iota),\ell+1},\ 0\,)_{\mathbb{B}^*_\tau}),\\
\boldsymbol{k}^*_{\ell,\mathsf{ran},1,j} := \ & (\,(-s_{\mathsf{ran},1,j,0},\ \boxed{0},\ 0,\ \eta_{\mathsf{ran},1,j,0},\ 0\,)_{\mathbb{B}^*_0},\\
& (\,s_{\mathsf{ran},1,j,t}\vec{e}_{t,1} + \theta_{\mathsf{ran},1,j,t}\vec{v}_t,\quad \boxed{0^{n_t}},\quad \vec{\eta}_{\mathsf{ran},1,j,t},\ 0\,)_{\mathbb{B}^*_t} : t = 1,\ldots,\ell),\\
\boldsymbol{k}^*_{\ell,\mathsf{ran},2,\tau} := \ & (\,(-s_{\mathsf{ran},2,\tau,0},\ \boxed{0},\ 0,\ \eta_{\mathsf{ran},2,\tau,0},\ 0\,)_{\mathbb{B}^*_0},\\
& (\,s_{\mathsf{ran},2,\tau,t}\vec{e}_{t,1} + \theta_{\mathsf{ran},2,\tau,t}\vec{v}_t,\quad \boxed{0^{n_t}},\quad \vec{\eta}_{\mathsf{ran},2,\tau,t},\ 0\,)_{\mathbb{B}^*_t} : t = 1,\ldots,\ell,\\
& (\,s_{\mathsf{ran},2,\tau,\ell+1}\vec{e}_{\tau,1},\quad \boxed{0^{n_\tau}},\quad \vec{\eta}_{\mathsf{ran},2,\tau,\ell+1},\ 0\,)_{\mathbb{B}^*_\tau}).
\end{aligned}$$

The challenge ciphertext consists of:

$$\begin{aligned}
\boldsymbol{c}_1 &:= (\,(\omega,\ \boxed{0},\ \boxed{\zeta},\ 0,\ \varphi_0)_{\mathbb{B}_0},\ (\boxed{\omega\vec{x}_t},\ \boxed{0^{n_t}},\ 0^{n_t},\ \varphi_t)_{\mathbb{B}_t} : t = 1,\ldots,\ell\,),\\
c_2 &:= g_T^{\zeta} m.
\end{aligned}$$

Remark 6. In the following, queried keys, $\boldsymbol{k}^*_{\ell,J}$ for $J \in \{\mathsf{dec}, (\mathsf{del}, (\tau, \iota)), (\mathsf{ran}, 1, j), (\mathsf{ran}, 2, \tau) \mid j = 1, \ldots, 2\ell, \tau = \ell+1, \ldots, d, \iota = 1, \ldots, n_\tau\}$, are described in a unified way as:

$$\begin{aligned}
\boldsymbol{k}^*_{\ell,J} := \ & (\,(-s_{J,0},\ \boxed{0},\ 1,\ \eta_{J,0},\ 0\,)_{\mathbb{B}^*_0},\\
& (\,s_{J,t}\vec{e}_{t,1} + \theta_{J,t}\vec{v}_t,\quad \boxed{0^{n_t}},\ \vec{\eta}_{J,t},\ 0\,)_{\mathbb{B}^*_t} : t = 1, \ldots, \ell\\
& (\,s_{J,\ell+1}\vec{e}_{\tau,1} + \widetilde{\psi}\vec{e}_{\tau,\iota},\ \boxed{0^{n_\tau}},\ \vec{\eta}_{J,\ell+1},\ 0\,)_{\mathbb{B}^*_\tau}),\\
& \text{where } s_{J,\ell+1} := 0,\ \vec{\eta}_{J,\ell+1} := 0^{n_\tau} \text{ if } J = \mathsf{dec}, (\mathsf{ran}, 1, j),\\
& \widetilde{\psi} := \psi \text{ if } J = (\mathsf{del}, (\tau, \iota)),\ \ \widetilde{\psi} := 0 \text{ otherwise},
\end{aligned}$$

and all the other variables, i.e., $s_{J,\cdot}, \theta_{J,\cdot}$ for $J \neq \mathsf{dec}, (\mathsf{ran}, 1, j)$, are defined in the description of Game 0. (This notation is well-defined when $J = \mathsf{dec}, (\mathsf{ran}, 1, j)$ and $\tau = \ell+1, \ldots, d$.)

Game 1 : Game 1 is the same as Game 0 except the following procedures.

1. When a create key query is issued by $\mathcal{A}$, the challenger of the game only records the specified predicates, and when a create delegated key query is issued, the challenger only records the specified keys and predicates. In this step, just the query is recorded, but no corresponding key is created.
2. When a reveal key query is issued for a hierarchical (level-ℓ) predicate $(\vec{v}_1, \ldots, \vec{v}_\ell)$ which has been already recorded, the challenger creates the queried key by using KeyGen.

Game 2 : Same as Game 1 except that the challenge ciphertext is:

$$\begin{aligned}
\boldsymbol{c}_1 &:= (\,(\omega, \boxed{w_0}, \zeta, 0, \varphi_0)_{\mathbb{B}_0},\ (\omega\vec{x}_t, \boxed{\vec{w}_t}, 0^{n_t}, \varphi_t)_{\mathbb{B}_t} : t = 1, \ldots, \ell\,),\\
c_2 &:= g_T^{\zeta} m,
\end{aligned}$$

where $w_0 \xleftarrow{\cup} \mathbb{F}_q$, $\vec{w}_t \xleftarrow{\cup} \mathbb{F}_q^{n_t}$.

Game 3-$(\boldsymbol{h}, \boldsymbol{J})$ ($\boldsymbol{h} = \boldsymbol{1}, \ldots, \boldsymbol{\nu}$; $\boldsymbol{J} \in \boldsymbol{\Pi} := \{\mathsf{dec}, (\mathsf{del}, (\boldsymbol{\tau}, \boldsymbol{\iota})), (\mathsf{ran}, 1, j), (\mathsf{ran}, \boldsymbol{2}, \boldsymbol{\tau}) \mid \boldsymbol{j} = \boldsymbol{1}, \ldots, \boldsymbol{2\ell}, \boldsymbol{\tau} = \boldsymbol{\ell+1}, \ldots, \boldsymbol{d}, \boldsymbol{\iota} = \boldsymbol{1}, \ldots, \boldsymbol{n_\tau}\}$) : Index J is incremented in the lexicographic order given in the description of Π. Game 3-$(1, 0)$ is Game 2. Game 3-$(h, (\mathsf{ran}, 2, d))$ is Game 3-$(h+1, 0)$.

Game 3-(h, J) is the same as Game 3-$(h, J-1)$ except that the Jth key, $\boldsymbol{k}^*_{\ell,J}$: in the hth reveal key query's reply is:

$$\begin{aligned}
\boldsymbol{k}^*_{\ell,J} := \ & (\,(-s_{J,0},\ \boxed{r_{J,0}},\ 1,\ \eta_{J,0},\ 0\,)_{\mathbb{B}^*_0},\\
& (\,s_{J,t}\vec{e}_{t,1} + \theta_{J,t}\vec{v}_t,\quad \boxed{\vec{r}_{J,t}},\ \vec{\eta}_{J,t},\ 0\,)_{\mathbb{B}^*_t} : t = 1, \ldots, \ell\\
& (\,s_{J,\ell+1}\vec{e}_{\tau,1} + \widetilde{\psi}\vec{e}_{\tau,\iota},\ \boxed{\vec{r}_{J,\tau}},\ \vec{\eta}_{J,\ell+1},\ 0\,)_{\mathbb{B}^*_\tau}),
\end{aligned}$$

where $r_{J,0} \xleftarrow{\cup} \mathbb{F}_q, \vec{r}_{J,t} \xleftarrow{\cup} \mathbb{F}_q^{n_t}$ for $t = 1, \ldots, \ell, \tau$, and the other variables are generated as in Game 3-$(h, J-1)$.

Game 4 : Game 4 is the same as Game 3-$(\nu, (\mathsf{ran}, 2, d)$ except that the challenge ciphertext is:

$$\boldsymbol{c}_1 := (\ (\omega, w_0, \boxed{\zeta'}, 0, \varphi_0)_{\mathbb{B}_0},\ (\boxed{\vec{x}'_t}, \vec{w}_t, 0^{n_t}, \varphi_t)_{\mathbb{B}_t} : t = 1, \ldots, \ell\ ),$$
$$c_2 := g_T^{\zeta} m,$$

where $\zeta, \zeta' \xleftarrow{\cup} \mathbb{F}_q, \vec{x}'_t \xleftarrow{\cup} \mathbb{F}_q^{n_t}$.

Let $\mathsf{Adv}^{(0)}_{\mathcal{A}}(\lambda)$, $\mathsf{Adv}^{(1)}_{\mathcal{A}}(\lambda)$,$\mathsf{Adv}^{(2)}_{\mathcal{A}}(\lambda)$, $\mathsf{Adv}^{(3\text{-}(h,J))}_{\mathcal{A}}(\lambda)$ and $\mathsf{Adv}^{(4)}_{\mathcal{A}}(\lambda)$ be the advantage of $\mathcal{A}$ in Game 0, Game 1, Game 2, Game 3-(h, J) and Game 4. It is obtained that $\mathsf{Adv}^{(4)}_{\mathcal{A}}(\lambda) = 0$.

We can evaluate the gaps between pairs of the above advantages using Problems 1 and 2 as in the proof of Theorem 1. □

References

[1] A. Agrawal, S. Gorbunov, V. Vaikuntanathan, H. Wee, Functional encryption: New perspectives and lower bounds, in R. Canetti, J.A. Garay, (eds.) *CRYPTO 2013, Part II*. LNCS, vol. 8043. (Springer, Heidelberg, 2013), pp. 500–518

[2] P.V. Ananth, A. Sahai, Functional Encryption for Turing Machines, in *TCC* 2016, pp. 125–153 (2016)

[3] N. Attrapadung, H. Imai, Dual-policy attribute-based encryption: Simultaneous access control with ciphertext and key policies. *IEICE Trans. Fundamentals*, E93-A, no. 1, pp. 116–125 (2010)

[4] A. Beimel, Secure schemes for secret sharing and key distribution. PhD Thesis, Israel Institute of Technology, Technion, Haifa, Israel, (1996)

[5] J. Bethencourt, A. Sahai, B. Waters, Ciphertext-policy attribute-based encryption, in *2007 IEEE Symposium on Security and Privacy* (IEEE Press, Los Alamitos, 2007), pp. 321–334.

[6] D. Boneh, X. Boyen, Efficient selective-ID secure identity based encryption without random oracles, in C. Cachin, J. Camenisch, (eds.) *EUROCRYPT 2004*. LNCS, vol. 3027 (Springer, Heidelberg, 2004), pp. 223–238

[7] D. Boneh, X. Boyen, Secure identity based encryption without random oracles, in M.K. Franklin, (ed.) *CRYPTO 2004*. LNCS, vol. 3152, (Springer, Heidelberg, 2004), pp. 443–459

[8] D. Boneh, X. Boyen, E. Goh, Hierarchical identity based encryption with constant size ciphertext, in Cramer, R. (ed.) *EUROCRYPT 2005*. LNCS, vol. 3494, (Springer, Heidelberg, 2005), pp. 440–456

[9] D. Boneh, X. Boyen, H. Shacham, Short group signatures, in Franklin, M.K. (ed.) *CRYPTO 2004*. LNCS, vol. 3152, (Springer, Heidelberg, 2004), pp. 41–55

[10] D. Boneh, M. Franklin, Identity-based encryption from the Weil pairing, in Kilian, J. (ed.) *CRYPTO 2001*. LNCS, vol. 2139, (Springer, Heidelberg, 2001), pp. 213–229

[11] D. Boneh, E.J. Goh, K. Nissim, Evaluating 2-DNF Formulas on Ciphertexts, in Kilian, J. (ed.) *TCC 2005*. LNCS, vol. 3378, (Springer, Heidelberg, 2005), pp. 325–341

[12] D. Boneh, M. Hamburg, Generalized identity based and broadcast encryption scheme, in Pieprzyk, J. (ed.) *ASIACRYPT 2008*. LNCS, vol. 5350, (Springer, Heidelberg, 2008), pp. 455–470

[13] D. Boneh, J. Katz, Improved efficiency for CCA-secure cryptosystems built using identity based encryption, in Menezes, A. (ed.) *CT-RSA 2005*. LNCS, vol. 3376, (Springer, Heidelberg, 2005), pp. 87–103

[14] D. Boneh, A. Sahai, B. Waters, Functional encryption: Definitions and challenges. *TCC 2011*, 253–273 (2011)

[15] D. Boneh, B. Waters, Conjunctive, subset, and range queries on encrypted data, in Vadhan, S.P. (ed.) *TCC 2007*. LNCS, vol. 4392, (Springer, Heidelberg, 2007), pp. 535–554

[16] X. Boyen, B. Waters, Anonymous hierarchical identity-based encryption (without random oracles), in Dwork, C. (ed.) *CRYPTO 2006*. LNCS, vol. 4117, (Springer, Heidelberg, 2006), pp. 290–307
[17] R. Canetti, S. Halevi, J. Katz, Chosen-ciphertext security from identity-based encryption, in Cachin, C., Camenisch, J.L. (eds.) *EUROCRYPT 2004*. LNCS, vol. 3027, (Springer, Heidelberg, 2004), pp. 207–222
[18] A.D. Caro, V. Iovino, A. Jain, A. O'Neill, O. Paneth, G. Persiano, On the achievability of simulation-based security for functional encryption, in Canetti, R., Garay, J.A. (eds.) *CRYPTO 2013*, Part II. LNCS, vol. 8043, (Springer, Heidelberg, 2013), pp. 519–535
[19] J. Chen, H.W. Lim, S. Ling, H. Wang, The relation and transformation between hierarchical inner product encryption and spatial encryption. *Design, Codes, and Cryptography*. Springer, Heidelberg, **71**(2), 347–364 (2014)
[20] J. Chen, H.M. Lim, S. Ling, L. Su, H. Wang, Spatial encryption supporting non-monotone access structure. *Des. Codes Cryptography. Design, Codes, and Cryptography*. Springer, Heidelberg, **73**(3), 731–746 (2014)
[21] C. Cocks, An identity based encryption scheme based on quadratic residues, in Honary, B. (ed.) *IMA Int. Conf.* LNCS, vol. 2260, (Springer, Heidelberg, 2001), pp. 360–363
[22] S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, B. Waters, Candidate Indistinguishability Obfuscation and Functional Encryption for all Circuits. *FOCS*, pp. 40–49 (2013)
[23] S. Garg, C. Gentry, S. Halevi, A. Sahai, B. Waters, Attribute-Based Encryption for Circuits from Multilinear Maps. *Crypto 2013*, LNCS, (Springer, Heidelberg, 2013)
[24] C. Gentry, Practical identity-based encryption without random oracles, in Vaudenay, S. (ed.) *EUROCRYPT 2006*. LNCS, vol. 4004, (Springer, Heidelberg, 2006), pp. 445–464
[25] C. Gentry, A. Silverberg, Hierarchical ID-based cryptography, in Zheng, Y. (ed.) *ASIACRYPT 2002*. LNCS, vol. 2501, (Springer, Heidelberg, 2002), pp. 548–566
[26] S. Goldwasser, D. Gordon, V. Goyal, A. Jain, J. Katz, F.H. Liu, A. Sahai, E. Shi, H.S. Zhou, Multi-input Functional Encryption. *EUROCRYPT 2014*, pp.578-602 (2014)
[27] S. Gorbunov, V. Vaikuntanathan, H. Wee, Attribute-Based Encryption for Circuits, *STOC 2013*, pp. 545–554 (2013)
[28] V. Goyal, O. Pandey, A. Sahai, B. Waters, Attribute-based encryption for fine-grained access control of encrypted data, in *ACM Conference on Computer and Communication Security 2006*, (ACM, New York, 2006), pp. 89–98
[29] J. Groth, A. Sahai, Efficient non-interactive proof systems for bilinear groups, in Smart, N.P. (ed.) *EUROCRYPT 2008*. LNCS, vol. 4965, (Springer, Heidelberg, 2008), pp. 415–432
[30] J. Katz, A. Sahai, B. Waters, Predicate encryption supporting disjunctions, polynomial equations, and inner products, in Smart, N.P. (ed.) *EUROCRYPT 2008*. LNCS, vol. 4965, (Springer, Heidelberg, 2008), pp. 146–162
[31] A.B. Lewko, Tools for Simulating Features of Composite Order Bilinear Groups in the Prime Order Setting, in Pointcheval, D., Johansson, T (eds.) *EUROCRYPT 2012*. LNCS, vol. 7237, (Springer, Heidelberg, 2012), pp. 318–335
[32] A. Lewko, T. Okamoto, A. Sahai, K. Takashima, B. Waters, Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption, in Gilbert, H. (ed.) *EUROCRYPT 2010*. LNCS, vol. 6110, (Springer, Heidelberg, 2010), pp. 62–91
[33] A.B. Lewko, B. Waters, New techniques for dual system encryption and fully secure HIBE with short ciphertexts, in Micciancio, D. (ed.) *TCC 2010*. LNCS, vol. 5978, (Springer, Heidelberg, 2010), pp. 455–479
[34] A.B. Lewko, B. Waters, Unbounded HIBE and Attribute-Based Encryption, in Paterson, K. (ed.) *EUROCRYPT 2011*. LNCS, vol. 6632, (Springer, Heidelberg, 2011), pp. 547–567
[35] T. Okamoto, K. Takashima, Homomorphic encryption and signatures from vector decomposition, in Galbraith, S.D., Paterson, K.G. (eds.) *Pairing 2008*. LNCS, vol. 5209, (Springer, Heidelberg, 2008), pp. 57–74
[36] T. Okamoto, K. Takashima, Hierarchical predicate encryption for inner-products, in Matsui, M. (ed.) *ASIACRYPT 2009*. LNCS, vol. 5912, (Springer, Heidelberg, 2009), pp. 214–231
[37] T. Okamoto, K. Takashima, Fully secure functional encryption with general relations from the decisional linear assumption, in Rabin, T. (ed.) *CRYPTO 2010*. LNCS, vol. 6223, (Springer, Heidelberg, 2010), pp. 191–208

[38] T. Okamoto, K. Takashima, Adaptively Attribute-Hiding (Hierarchical) Inner Product Encryption, in Pointcheval, D., Johansson, T., (eds.) *EUROCRYPT 2012*. LNCS, vol. 7237, (Springer, Heidelberg, 2012), pp. 591–608

[39] T. Okamoto, K. Takashima, Fully Secure Unbounded Inner-Product and Attribute-Based Encryption, in Wang, X., Sako, K., (eds.) *ASIACRYPT 2012*. LNCS, vol. 7658, (Springer, Heidelberg, 2012), pp. 349–366

[40] T. Okamoto, K. Takashima, Achieving short ciphertexts or short secret-keys for adaptively secure general inner-product encryption, in Des. Codes Cryptography, vol. 77 (2–3), (Springer, Heidelberg 2015), pp. 725–771, Preliminary version appeared in CANS 2011.

[41] A. O'Neill, Definitional issues in functional encryption. Cryptology ePrint Archive, Report 2010/556, 2010. http://eprint.iacr.org/ (2010).

[42] R. Ostrovsky, A. Sahai, B. Waters, Attribute-based encryption with non-monotonic access structures, in *ACM Conference on Computer and Communication Security 2007*, (ACM, New York, 2007), pp. 195–203

[43] M. Pirretti, P. Traynor, P. McDaniel, B. Waters, Secure attribute-based systems, in *ACM Conference on Computer and Communication Security 2006*, (ACM, New York, 2006), pp. 99–112

[44] A. Sahai, B. Waters, Fuzzy identity-based encryption. in Cramer, R. (ed.) *EUROCRYPT 2005*. LNCS, vol. 3494, (Springer, Heidelberg, 2005), pp. 457–473

[45] E. Shi, B. Waters, Delegating capability in predicate encryption systems, in L. Aceto, I. Damgård, L.A. Goldberg, M.M. Halldørsson, A. Ingølfsdøttir, I. Walukiewicz, (eds.) *ICALP* (2) 2008. LNCS, vol. 5126, (Springer, Heidelberg, 2008), pp. 560–578

[46] K. Takashima, Expressive attribute-based encryption with constant-size ciphertexts from the decisional linear assumption, in M. Abdalla, R. De Prisco, (eds.) *SCN* 2014. LNCS, vol. 8642, (Springer, Heidelberg, 2014), pp. 298–317

[47] K. Takashima, New proof techniques for DLIN-based adaptively secure attribute-based encryption, in J. Pieprzyk, S. Suriadi, (eds.) *ACISP 2017 (I)*. LNCS, vol. 10342, (Springer, Heidelberg, 2017), pp. 85–105

[48] B. Waters, Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization, in D. Catalano, N. Fazio, R. Gennaro, A. Nicolosi, (eds.) *PKC 2011*. LNCS, vol. 6571, (Springer, Heidelberg, 2011), pp. 53–70

[49] B. Waters, Dual system encryption: realizing fully secure IBE and HIBE under simple assumptions, in Halevi, S. (ed.) *CRYPTO 2009*. LNCS, vol. 5677, (Springer, Heidelberg, 2009), pp. 619–636