Practical Functional Encryption for Quadratic Functions with Applications to Predicate Encryption

Carmen Elisabetta Zaira Baltico[1], Dario Catalano[1], Dario Fiore[2], and Romain Gay[3]*

[1] Dipartimento di Matematica e Informatica, Università di Catania, Italy.
carmenez@hotmail.it, catalano@dmi.unict.it
[2] IMDEA Software Institute, Madrid, Spain.
dario.fiore@imdea.org
[3] ENS, CNRS, INRIA, and PSL, Paris, France
rgay@di.ens.fr

Abstract. We present two practically efficient functional encryption schemes for a large class of quadratic functionalities. Specifically, our constructions enable the computation of so-called *bilinear maps* on encrypted vectors. This represents a practically relevant class of functions that includes, for instance, multivariate quadratic polynomials (over the integers). Our realizations work over asymmetric bilinear groups and are surprisingly efficient and easy to implement. For instance, in our most efficient scheme the public key and each ciphertext consist of $2n + 1$ and $4n + 2$ group elements respectively, where n is the dimension of the encrypted vectors, while secret keys are only two group elements. Our two schemes build on similar ideas, but develop them in a different way in order to achieve distinct goals. Our first scheme is proved (selectively) secure under standard assumptions, while our second construction is concretely more efficient and is proved (adaptively) secure in the generic group model. As a byproduct of our functional encryption schemes, we show new predicate encryption schemes for degree-two polynomial evaluation, where ciphertexts consist of only $O(n)$ group elements. This significantly improves the $O(n^2)$ bound one would get from inner product encryption-based constructions.

1 Introduction

Traditional public key encryption allows the owner of a secret key sk to decrypt ciphertexts created with respect to a (matching) public key mpk. At the same time, without sk, ciphertexts should not reveal any non trivial information about encrypted messages. This all-or-nothing nature of encryption is becoming insufficient in applications where a more fine-grained access to data is required. Functional Encryption (FE) allows to overcome this user-centric access to data of encryption in a very elegant way. Intuitively, given $\mathsf{Encrypt}(m)$ and a key sk_f corresponding to some function f, the owner of sk_f learns $f(m)$ and nothing else. Apart from being an interesting theoretical object, Functional Encryption has many natural applications. Think about cloud storage scenarios where users can rely on powerful external servers to store their data. To preserve their privacy, users might want to store their files encrypted. At the same time, the users may wish to let the service providers perform basic data mining operations on this data for commercial purposes, without necessarily disclosing the whole data. Functional Encryption allows to reconcile these seemingly contradicting needs, as service providers can get secret keys that allow them to perform the desired computations while preserving, as much as possible, the privacy of users.

In terms of security, the standard notion for functional encryption is *indistinguishability*. Informally, this notion states that an adversary who is allowed to see the secret keys for functionalities $f_1, \dots f_n$ should not be able to tell apart which of the challenge messages m_0 or m_1 has been encrypted, under the restriction that $f_i(m_0) = f_i(m_1)$, for all i. This notion was studied in [13,35]

* Supported by ERC Project aSCEND (639554).

and shown inadequate for certain, complex, functionalities[4]. They also explored an alternative, simulation-based, definition, which however cannot be satisfied, in general, without resorting to the random oracle heuristic.

Background on Functional Encryption. The idea of functional encryption originates from Identity Based Encryption (IBE) [37,11] and the closely related concept of Searchable Encryption [10,1]. In IBE, the encrypted message can be interpreted as a pair (I, m), where I is a public string and m is the actual message (often called the "payload"). More in general, the index I can be interpreted as a set of attributes that can be either public or private. Public index schemes are often referred to as attribute based encryption [36,27], a primitive that is by now very well understood [25]. For private index schemes, the situation is more intricate. A first distinction is between *weakly* and *fully attribute hiding* schemes [5]. The former notion refers to schemes where the set of secret keys the adversary is allowed to see in the security games is significantly restricted. The adversary is allowed to ask only keys corresponding to functions that cannot be used to decrypt the challenge message. Examples of these schemes are Anonymous Identity based encryption [11,22], Hidden Vector Encryption [15] and (private index) predicate encryption [28,26].

Things are less well established for the setting of private index, fully attribute hiding schemes, a notion that turns out to be equivalent to full fledged functional encryption [13]. Indeed, all known constructions supporting arbitrary circuits, either work for the case of bounded collusions [24,23] or rely on powerful, but poorly understood, assumptions (e.g., [20]). Moreover, they are all terribly inefficient from a practical point of view.

To improve efficiency, a very natural approach is to try to realize schemes using a different, bottom up, perspective. Rather than focusing on generality, one might focus on devising efficient realizations for specific functionalities of practical interest. In 2015, Abdalla *et al.* [2] addressed this question for the case of linear functionalities. In particular, they show a construction which is both very simple and relies on standard, well studied assumptions (such as LWE and DDH). The construction was proved secure in the so-called *selective* setting where the adversary is expected to choose the messages on which she wants to be challenged in advance, even before the public key is set up. Not too surprisingly, this result sparkled significant interest in this bottom-up approach, with several results proposing new schemes [6], models [9,4] and improved security [6,3].

Still, none of these results managed to efficiently support more than linear functionalities. In particular, the technical barrier is to find FE schemes in which ciphertexts have size *linear* in the number of encrypted elements, in contrast to quadratic, as it can be achieved by using a scheme for linear functions.[5] This motivates the following question:

Can we construct a practically efficient functional encryption scheme supporting more than linear functionalities?

1.1 Our Contribution

In this paper we answer the question above in the affirmative. We build two efficient functional encryption schemes for quadratic functions with linear-size ciphertexts. In terms of security, our

[4] Here by complex we intend, for instance, functions that are supposed to have some computational hiding properties. In particular, Boneh *et al.* [13] argue that, in applications where security relies on such properties, indistinguishability might become problematic.

[5] Indeed, we note that a functional encryption for linear polynomials can be used to support, say, quadratic polynomials, by simply encrypting all the degree-two monomials in advance. This however leads to an inefficient solution where the size of the ciphertexts is quadratic in the number of variables.

first scheme is proven selective-secure under standard assumptions (Matrix Decisional Diffie Hellman [18] and 3-party DDH [12]), whereas our second scheme is proven adaptively secure in the generic group model, and is more efficient. In terms of functionality, to be more specific, our schemes allows to compute *bilinear maps over the integers*: messages are expressed as pairs of vectors $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{Z}^n \times \mathbb{Z}^m$, secret keys are associated with $(n \times m)$ matrices $\mathbf{F}$, and decryption allows to compute $\boldsymbol{x}^\top \mathbf{F} \boldsymbol{y} = \sum_{i,j} f_{ij} x_i y_j$. Bilinear maps represent a very general class of quadratic functions that includes, for instance, multivariate quadratic polynomials. These functions have several practical applications. For instance, a quadratic polynomial can express many statistical functions (e.g., (weighted) mean, variance, covariance, root-mean-square), the euclidean distance between two vectors, and the application of a linear or quadratic classifier (e.g., linear or quadratic regression).

In addition to the above applications of quadratic functions, we also show that our FE for bilinear maps can be used to construct new Predicate Encryption schemes (PE for short) that satisfy the *fully attribute hiding* property, and yield efficient solutions for interesting classes of predicates, such as constant-depth boolean formulas and comparisons. In a nutshell, in our PE scheme ciphertexts are associated with a set of attributes $(x_1, \ldots, x_n)$ and a plaintext M, secret keys are associated with a degree-two polynomial P, and the decryption of a ciphertext $\mathsf{Ct}_{(x_1,\ldots,x_n)\in\mathbb{Z}^n}$ with a secret key $\mathsf{sk}_{P\in\mathbb{Z}[X_1,\ldots,X_n],\ \mathsf{deg}(P)\leq 2}$ recovers M if, and only if, $P(x_1, \ldots, x_n) = 1$. The attribute-hiding property refers to the fact that $\mathsf{Ct}_{(x_1,\ldots,x_n)\in\mathbb{Z}^n}$ leaks no information on its attribute $(x_1, \ldots, x_n)$, beyond what is inherently leaked by the boolean value $P(x_1, \ldots, x_n) \overset{?}{=} 1$. Using our new functional encryption schemes as underlying building blocks, we obtain PE constructions for quadratic polynomials where ciphertexts consist of only $O(n)$ group elements. This is in sharp contrast with the $O(n^2)$ solutions one would get via inner product encryption schemes (e.g., [28]).

An informal description of our FE schemes. Our solutions work over asymmetric bilinear groups $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ and are quite efficient. They are both essentially optimal in communication size: public key and ciphertexts are both *linear* in the size of the encrypted vectors; secret keys are only two group elements. Both our schemes share similar underlying ideas. These ideas are however developed in different ways to achieve different security and efficiency goals. Our first scheme, can be proved (selectively) secure under standard intractability assumptions but achieves somewhat worse performances in practice. The second construction, on the other hand, is (concretely) more efficient but it can be proved (adaptively) secure only in the generic group model. In what follows we will highlight some of the core ideas underlying both schemes. How these ideas are implemented and developed in the two cases will be discussed when introducing each specific scheme.

Let us recall that the functionality provided by our FE scheme is that one encrypts pairs of vectors $\boldsymbol{x}, \boldsymbol{y}$, functions are matrices $\mathbf{F}$, and decryption allows to obtain $\boldsymbol{x}^\top \mathbf{F} \boldsymbol{y}$.

The initial idea of the construction is to encrypt the two vectors $\boldsymbol{x} \in \mathbb{Z}^n$ and $\boldsymbol{y} \in \mathbb{Z}^m$ in a sort of "matrix" ElGamal in the two groups $\mathbb{G}_1$ and $\mathbb{G}_2$ respectively. Namely, we set

$$\mathsf{Ct}_{(\boldsymbol{x},\boldsymbol{y})} = \{[\rho \mathbf{A} \boldsymbol{r}_i + \boldsymbol{b} x_i]_1\}_{i=1,\ldots,n}, [\sigma \mathbf{B} \boldsymbol{s}_j + \boldsymbol{a} y_j]_2\}_{j=1,\ldots,m}$$

where: ρ, σ are randomly chosen, $\{\mathbf{A}\boldsymbol{r}_i, \mathbf{B}\boldsymbol{s}_j\}_{i,j}$ are in the public key, and are constructed from two random matrices $\mathbf{A}$ and $\mathbf{B}$ and a collection of random vectors $\{\boldsymbol{r}_i, \boldsymbol{s}_j\}_{i,j}$, and $\boldsymbol{a}, \boldsymbol{b}$ are more carefully chosen vectors (see below) [6]. Towards finding a decryption method, we first observe that, given

[6] Here we adopt the, by now standard, implicit representation $[x]_s = g^x \in \mathbb{G}_s$. This notion can be easily extended to vectors and matrices (see [18]).

$\mathsf{Ct}_{(\boldsymbol{x},\boldsymbol{y})}$ and a function $\mathbf{F}$, one can use the bilinear map to compute

$$U = [(\rho\sigma)\sum_{ij} f_{ij}\boldsymbol{r}_i^\top \mathbf{A}^\top \mathbf{B}\boldsymbol{s}_j + \rho\sum_{ij} f_{ij}\boldsymbol{r}_i^\top \mathbf{A}^\top \boldsymbol{a}y_j + \sigma\sum_{ij} f_{ij}\boldsymbol{s}_j^\top \mathbf{B}^\top \boldsymbol{b}x_i + (\boldsymbol{b}^\top \boldsymbol{a})\cdot \boldsymbol{x}^\top \mathbf{F}\boldsymbol{y}]_T.$$

Moreover, if we let $[\sum_{ij} f_{ij}\boldsymbol{r}_i^\top \mathbf{A}^\top \mathbf{B}\boldsymbol{s}_j]_1$ be the secret key for function $\mathbf{F}$ and include $[\rho\sigma]_2$ in the ciphertext, one can remove the first term in U.

Our two schemes then extend this basic blueprint with additional (but different!) structure so as to enable the extraction from U of the value $[\boldsymbol{x}^\top \mathbf{F}\boldsymbol{y}]_T$. From this, in turn, the function's result can be obtained via a brute force discrete log computation[7]. At a very intuitive level (and deliberately ignoring many important details) a key difference between the two schemes lies in the way $\mathbf{A}$, $\mathbf{B}$, $\boldsymbol{a}$ and $\boldsymbol{b}$ are constructed.

In our first scheme, $\mathbf{A}$ and $\mathbf{B}$ are carefully sampled so that to be able to prove (selective) security under standard intractability assumptions (e.g. Matrix Decisional Diffie-Hellman). Moreover $\boldsymbol{a}$ and $\boldsymbol{b}$ are chosen such that $\mathbf{A}^\top \boldsymbol{a} = \mathbf{B}^\top \boldsymbol{b} = \mathbf{0}$ and $\boldsymbol{b}^\top \boldsymbol{a} = 1$. This ensures that the intermediate values $\rho\sum_{ij} f_{ij}\boldsymbol{r}_i^\top \mathbf{A}^\top \boldsymbol{a}y_j$, $\sigma\sum_{ij} f_{ij}\boldsymbol{s}_j^\top \mathbf{B}^\top \boldsymbol{b}x_i$ cancel out at decryption time.

In our second scheme, on the other hand, the public key values $\mathbf{A}\boldsymbol{r}_i$ and $\mathbf{B}\boldsymbol{s}_j$ are simple scalars, and the "canceling" is performed via an appropriate choice of vectors $\boldsymbol{a}, \boldsymbol{b}$ and simple algebraic manipulations. This makes the resulting construction (concretely) more efficient. At the same time, we lose the possibility to rely on (general) matrix assumptions and we are able to prove (adaptive) security in the generic group model. To this end, as a contribution that can be of independent interest, we state and prove a master theorem that shows hardness in the generic bilinear group model for a broad family of interactive decisional problems (notably, a family that includes our FE scheme), extending some of the tools and results of the generic group framework recently developed by Barthe et al. [8].

Concurrent and Independent work. In concurrent and independent work, Lin [31], and Ananth and Sahai [7] present constructions of *private-key* functional encryption schemes for degree-D polynomials based on D-linear maps. As a special case for $D = 2$, these schemes support quadratic polynomials from bilinear maps, as ours. Also, in terms of security, the construction of Lin is proven selectively secure based on the SXDH assumption, while the scheme of Ananth and Sahai is selectively secure based on ad-hoc assumptions that are justified in the multilinear group model. In comparison to these works, our schemes have the advantage of working in the (arguably more challenging) *public key* setting.

2 Preliminaries

Notation. We denote with $\lambda \in \mathbb{N}$ a security parameter. A *probabilistic polynomial time* (PPT) algorithm $\mathcal{A}$ is a randomized algorithm for which there exists a polynomial $p(\cdot)$ such that for every input x the running time of $\mathcal{A}(x)$ is bounded by $p(|x|)$. We say that a function $\epsilon : \mathbb{N} \to \mathbb{R}^+$ is *negligible* if for every positive polynomial $p(\lambda)$ there exists $\lambda_0 \in \mathbb{N}$ such that for all $\lambda > \lambda_0$: $\epsilon(\lambda) < 1/p(\lambda)$. If S is a set, $x \leftarrow_{\mathrm{R}} S$ denotes the process of selecting x uniformly at random in S. If $\mathcal{A}$ is a probabilistic algorithm, $y \leftarrow_{\mathrm{R}} \mathcal{A}(\cdot)$ denotes the process of running $\mathcal{A}$ on some appropriate input and assigning its output to y. For a positive integer n, we denote by $[n]$ the set $\{1, \ldots, n\}$.

[7] This means that in our scheme messages and functions coefficients are assumed to be sufficiently small integers.

We denote vectors $\boldsymbol{x} = (x_i)$ and matrices $\mathbf{A} = (a_{i,j})$ in bold. For a set S (resp. vector $\boldsymbol{x}$) $|S|$ (resp. $|\boldsymbol{x}|$) denotes its cardinality (resp. number of entries). For any prime p and any matrix $\mathbf{A} \in \mathbb{Z}_p^{n\times m}$ with $n \geq m$, we denote by $\mathsf{orth}(\mathbf{A}) := \{\boldsymbol{a}^\perp \in \mathbb{Z}_p^n : \mathbf{A}^\top \boldsymbol{a}^\perp = \mathbf{0}\}$. For all square matrices $\mathbf{A} \in \mathbb{Z}_p^{n\times n}$, we denote by $\det(\mathbf{A})$ the determinant of $\mathbf{A}$. For any $n \in \mathbb{N}^*$, we denote by GL_n the general linear group of degree n, that is, the set of all $n \times n$ invertible matrices over $\mathbb{Z}_p$.

Bilinear Groups. Let $\mathcal{G}(1^\lambda)$ be an algorithm (that we call a *bilinear group generator*) which takes as input the security parameter and outputs the description of a bilinear group setting $\mathsf{bgp} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2)$, where $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$ are groups of the same prime order $p > 2^\lambda$, $g_1 \in \mathbb{G}_1$ and $g_2 \in \mathbb{G}_2$ are two generators, and $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$ is an efficiently computable, non-degenerate, bilinear map. We define $g_T = e(g_1, g_2)$ as the canonical generator of $\mathbb{G}_T$. In the case $\mathbb{G}_1 = \mathbb{G}_2$, the groups are said *symmetric*, else they are said *asymmetric*. In this paper we work with *asymmetric* bilinear groups in which there is no efficiently computable isomorphisms between $\mathbb{G}_1$ and $\mathbb{G}_2$ (these are also known as Type-III groups [19]).

We use implicit representation of group elements as introduced in [18]. For $s \in \{1, 2, T\}$ and $x \in \mathbb{Z}_p$, we let $[x]_s = g_s^x \in \mathbb{G}_s$. This notation is extended to matrices (and vectors) as follows. For any $\mathbf{A} = (a_{i,j}) \in \mathbb{Z}_p^{m\times n}$ we define

$$[\mathbf{A}]_s = \begin{pmatrix} g_s^{a_{1,1}} & \dots & g_s^{a_{1,n}} \\ \\ g_s^{a_{m,1}} & \dots & g_s^{a_{m,n}} \end{pmatrix} \in \mathbb{G}_s^{m\times n}$$

Note that from an element $[x]_s \in \mathbb{G}_s$ and a scalar a it is possible to efficiently compute $[ax] \in \mathbb{G}_s$. Also, given group elements $[a]_1 \in \mathbb{G}_1$ and $[b]_2 \in \mathbb{G}_2$, one can efficiently compute $[ab]_T = e([a]_1, [b]_2)$. Furthermore, given a matrix of scalars $\mathbf{F} = (f_{i,j}) \in \mathbb{Z}_p^{n\times n}$ and two n-dimensional vectors of group elements $[\boldsymbol{a}]_1, [\boldsymbol{b}]_2$, one can efficiently compute

$$[\boldsymbol{a}^\top \mathbf{F}\, \boldsymbol{b}]_T = \left[\sum_{i,j\in[n]} f_{i,j} \cdot a_i \cdot b_j\right]_T = \sum_{i,j\in[n]} f_{i,j} \cdot e([a_i]_1, [b_j]_2)$$

As above, for an easier and more compact presentation, in our work we slightly abuse notation and treat all groups $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ as additive groups.

2.1 Complexity assumptions

We recall the definitions of the Matrix Decision Diffie-Hellman (mddh) Assumption [18].

Definition 1 (Matrix Distribution). *Let $k \in \mathbb{N}$. We call $\mathcal{D}_k$ a matrix distribution if it outputs in polynomial time matrices in $\mathbb{Z}_p^{(k+1)\times k}$ of full rank k, and satisfying the following property,*

Property 1.

$$\Pr[\mathsf{orth}(\mathbf{A}) \subseteq \mathsf{span}(\mathbf{B})] = \frac{1}{\Omega(p)},$$

where $\mathbf{A}, \mathbf{B} \leftarrow_{\mathrm{R}} \mathcal{D}_k$.

Without loss of generality, we assume the first k rows of $\mathbf{A} \leftarrow_{\mathrm{R}} \mathcal{D}_k$ form an invertible matrix. Note that the basis property is not explicit in [18], but, as noted in [16, Lemma 1 (basis lemma)], all examples of matrix distribution presented in [18, Section 3.4], namely $\mathcal{U}_k$, $\mathcal{L}_k$, $\mathcal{SC}_k$, $\mathcal{C}_k$ and $\mathcal{IL}_k$, satisfy this property.

The $\mathcal{D}_k$-Matrix Diffie-Hellman problem in $\mathbb{G}_s$ for $s \in \{1, 2, T\}$ is to distinguish the two distributions $([\mathbf{A}]_s, [\mathbf{A}\boldsymbol{w}]_s)$ and $([\mathbf{A}]_s, [\boldsymbol{u}]_s)$ where $\mathbf{A} \leftarrow_{\mathrm{R}} \mathcal{D}_k$, $\boldsymbol{w} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$ and $\boldsymbol{u} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^{k+1}$.

Definition 2 ($\mathcal{D}_k$-Matrix Diffie-Hellman Assumption $\mathcal{D}_k$-mddh). *Let $\mathcal{D}_k$ be a matrix distribution. The $\mathcal{D}_k$-Matrix Diffie-Hellman ($\mathcal{D}_k$-mddh) Assumption holds relative to $\mathcal{G}$ in $\mathbb{G}_s$, for $s \in \{1, 2, T\}$, if for all PPT adversaries $\mathcal{A}$,*

$$\mathbf{Adv}_{\mathcal{G},\mathcal{A}}^{\mathcal{D}_k\text{-mddh}}(\lambda) := |\Pr[\mathcal{A}(\mathsf{bgp}, [\mathbf{A}]_s, [\mathbf{A}\boldsymbol{w}]_s) = 1] - \Pr[\mathcal{A}(\mathsf{bgp}, [\mathbf{A}]_s, [\boldsymbol{u}]_s) = 1]| = \mathsf{negl}(\lambda),$$

where probabilities are over $\mathsf{bgp} \leftarrow_{\mathrm{R}} \mathcal{G}(1^\lambda)$, $\mathbf{A} \leftarrow_{\mathrm{R}} \mathcal{D}_k, \boldsymbol{w} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k, \boldsymbol{u} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^{k+1}$.

For each $k \geq 1$, [18] specifies distributions ($\mathcal{U}_k$, $\mathcal{L}_k$, $\mathcal{SC}_k$, $\mathcal{C}_k$ and $\mathcal{IL}_k$) over $\mathbb{Z}_p^{(k+1)\times k}$ such that the corresponding $\mathcal{D}_k$-mddh assumptions are generically secure in bilinear groups and form a hierarchy of increasingly weaker assumptions. $\mathcal{L}_k$-mddh is the well known k-Linear Assumption k-Lin with 1-Lin = DDH.

We also recall the definition of 3-party Decision Diffie-Hellman (3-pddh) Assumption introduced in [12]. We give a variant in the asymmetric-pairing setting.

Definition 3 (3-party Decision Diffie-Hellman Assumption 3-pddh). *We say that the 3-party Decision Diffie-Hellman Assumption (3-pddh) Assumption holds relative to $\mathcal{G}$ if for all PPT adversaries $\mathcal{A}$,*

$$\begin{aligned}\mathbf{Adv}_{\mathcal{G},\mathcal{A}}^{3\text{-pddh}}(\lambda) := |\Pr[\mathcal{A}(\mathsf{bgp}, [a]_1, [b]_2, [c]_1, [c]_2, [abc]_1) = 1] \\ - \Pr[\mathcal{A}(\mathsf{bgp}, [a]_1, [b]_2, [c]_1, [c]_2, [d]_1) = 1]| = \mathsf{negl}(\lambda)\end{aligned}$$

where the probability is taken over $\mathsf{bgp} \leftarrow_{\mathrm{R}} \mathcal{G}(1^\lambda)$, $a, b, c, d \leftarrow_{\mathrm{R}} \mathbb{Z}_p$.

2.2 Functional Encryption

We recall the definitions of Functional Encryption as given by Boneh, Sahai and Waters [13].

Definition 4 (Functionality). *A functionality F defined over $(\mathcal{K}, \mathcal{M})$ is a function $F : \mathcal{K} \times \mathcal{M} \to \mathcal{Y} \cup \{\perp\}$ where $\mathcal{K}$ is a key space, $\mathcal{M}$ is a message space and $\mathcal{Y}$ is an output space which does not contain the special symbol $\perp$.*

Definition 5 (Functional Encryption). *A* functional encryption *scheme* FE *for a functionality F is defined by a tuple of algorithms* FE = (Setup, KeyGen, Encrypt, Decrypt) *that work as follows.*

Setup(1^λ, F) *takes as input a security parameter 1^λ, the functionality $F : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$, and outputs a master secret key* msk *and a master public key* mpk.

KeyGen(msk, K) *takes as input the master secret key and a key $K \in \mathcal{K}$ of the functionality (i.e., a function), and outputs a secret key* sk$_K$.

Encrypt(mpk, msk, M) *takes as input the master public key* mpk *and a message $M \in \mathcal{M}$, and outputs a ciphertext* Ct. *It can take as an additional input the master secret key, in which case, we talk about private-key functional encryption. By opposition, when* msk *is not an input of the encryption, algorithm, we say that* FE *is public-key.*

$\mathsf{Decrypt}(\mathsf{sk}_K, \mathsf{Ct})$ *takes as input a secret key* sk_K *and a ciphertext* Ct*, and returns an output* $Y \in \mathcal{Y} \cup \{\perp\}$.

For correctness*, it is required that for all* $(\mathsf{mpk}, \mathsf{msk}) \leftarrow_{\mathrm{R}} \mathsf{Setup}(1^\lambda)$*, all keys* $K \in \mathcal{K}$ *and all messages* $M \in \mathcal{M}$*, if* $\mathsf{sk}_K \leftarrow_{\mathrm{R}} \mathsf{KeyGen}(\mathsf{msk}, K)$ *and* $\mathsf{Ct} \leftarrow_{\mathrm{R}} \mathsf{Encrypt}(\mathsf{mpk}, \mathsf{msk}, M)$*, then it holds with overwhelming probability that* $\mathsf{Decrypt}(\mathsf{sk}_K, \mathsf{Ct}) = F(K, M)$ *whenever* $F(K, M) \neq \perp$.

Indistinguishability-Based Security. For a functional encryption scheme FE for a functionality F over $(\mathcal{K}, \mathcal{M})$, security against chosen-plaintext attacks (IND-FE-CPA, for short) is defined via the following experiment, denoted $\mathbf{Exp}^{\mathsf{ind\text{-}fe\text{-}cpa\text{-}}\beta}_{\mathsf{FE},\mathcal{A}}(\lambda)$, which is parametrized by an adversary $\mathcal{A}$, a bit $\beta \in \{0, 1\}$, and a security parameter λ.

Setup: run $(\mathsf{mpk}, \mathsf{msk}) \leftarrow_{\mathrm{R}} \mathsf{Setup}(1^\lambda)$ and give mpk to $\mathcal{A}$.
Query: $\mathcal{A}$ adaptively makes secret key queries. At each query, $\mathcal{A}$ specifies a key K and obtains $\mathsf{sk}_K \leftarrow_{\mathrm{R}} \mathsf{KeyGen}(\mathsf{msk}, K)$ from the challenger.
Challenge: $\mathcal{A}$ chooses a pair of messages $M_0, M_1 \in \mathcal{M}$ such that $F(K, M_0) = F(K, M_1)$ holds for all keys K queried in the previous phase. The challenger computes $\mathsf{Ct}^* \leftarrow_{\mathrm{R}} \mathsf{Encrypt}(\mathsf{mpk}, M_\beta)$ and returns Ct^* to $\mathcal{A}$.
Query: $\mathcal{A}$ makes more secret key queries. At each query $\mathcal{A}$ can adaptively choose a key $K \in \mathcal{K}$, but under the requirement that $F(K, M_0) = F(K, M_1)$.
Guess: $\mathcal{A}$ eventually outputs a bit $\beta' \in \{0, 1\}$, and the experiment outputs the same bit.

For any stateful adversary $\mathcal{A}$, any functional encryption scheme FE for a functionality F over $(\mathcal{K}, \mathcal{M})$, any bit $\beta \in \{0, 1\}$, and any security parameter λ, we give a compact description of experiment $\mathbf{Exp}^{\mathsf{ind\text{-}pe\text{-}cpa\text{-}}\beta}_{\mathsf{PE},\mathcal{A}}(\lambda)$, and its selective version $\mathbf{Exp}^{\mathsf{sel\text{-}ind\text{-}pe\text{-}cpa\text{-}}\beta}_{\mathsf{PE},\mathcal{A}}(\lambda)$, in Figure 1.

```
$\boxed{\mathbf{Exp}^{\mathsf{ind\text{-}fe\text{-}cpa\text{-}}\beta}_{\mathsf{FE},\mathcal{A}}(\lambda)}$, $\boxed{\mathbf{Exp}^{\mathsf{sel\text{-}ind\text{-}fe\text{-}cpa\text{-}}\beta}_{\mathsf{FE},\mathcal{A}}(\lambda)}$ :
$\boxed{(M_0, M_1) \leftarrow \mathcal{A}(1^\lambda)}$
$(\mathsf{mpk}, \mathsf{msk}) \leftarrow_{\mathrm{R}} \mathsf{Setup}(1^\lambda)$
$\boxed{\mathsf{Ct} := \mathsf{EncO}(M_0, M_1)}$
$\beta' \leftarrow \mathcal{A}(\mathsf{mpk}, \boxed{\mathsf{Ct}})^{\mathsf{KeyGenO}(\cdot), \boxed{\mathsf{EncO}(\cdot,\cdot)}}$
Return $\beta'$.

$\mathsf{EncO}(M_0, M_1)$:
Return $\mathsf{Ct}^\star := \mathsf{Encrypt}(\mathsf{mpk}, \mathsf{msk}, M_\beta)$

$\mathsf{KeyGenO}(K \in \mathcal{K})$:
Return $\mathsf{sk}_K := \mathsf{KeyGen}(\mathsf{msk}, K)$
```

Fig. 1. Experiments $\mathbf{Exp}^{\mathsf{ind\text{-}fe\text{-}cpa\text{-}}\beta}_{\mathsf{FE},\mathcal{A}}(\lambda)$ and $\mathbf{Exp}^{\mathsf{sel\text{-}ind\text{-}fe\text{-}cpa\text{-}}\beta}_{\mathsf{FE},\mathcal{A}}(\lambda)$ for $b \in \{0, 1\}$, used to define adaptive, and selective security of FE, respectively. In each procedure, the components inside a solid (dotted) frame are only present in the games marked by a solid (dotted) frame, and the components inside a gray frame only appears for private-key FE schemes. In both games, the oracle $\mathsf{EncO}(\cdot, \cdot)$ is queries at most once (by $\mathcal{A}$ or the game itself), on M_0, M_1, such that for all queries K to $\mathsf{KeyGenO}(\cdot)$, we have: $F(K, M_0) = F(K, M_1)$. Note that in the case of private-key FE, this corresponds to single-ciphertext security (which does not imply many-ciphertext security).

We define the advantage of $\mathcal{A}$ for adaptive security as:

$$\begin{aligned}\mathbf{Adv}^{\mathsf{ind\text{-}fe\text{-}cpa}}_{\mathsf{FE},\mathcal{A}}(\lambda) &:= \left|\Pr[\mathbf{Exp}^{\mathsf{ind\text{-}fe\text{-}cpa\text{-}0}}_{\mathsf{FE},\mathcal{A}}(\lambda) = 1] - \Pr[\mathbf{Exp}^{\mathsf{ind\text{-}fe\text{-}cpa\text{-}1}}_{\mathsf{FE},\mathcal{A}}(\lambda) = 1]\right| \\ &= \left|1 - 2\Pr\left[\beta' = \beta : \begin{array}{l}\beta \leftarrow_{\mathrm{R}} \{0,1\} \\ \mathbf{Exp}^{\mathsf{ind\text{-}fe\text{-}cpa\text{-}}\beta}_{\mathsf{FE},\mathcal{A}}(\lambda) = \beta'\end{array}\right]\right|\end{aligned}$$

We define the advantage $\mathbf{Adv}^{\mathsf{sel\text{-}ind\text{-}fe\text{-}cpa}}_{\mathsf{FE},\mathcal{A}}(\lambda)$ for selective security similarly, with respect to experiments $\mathbf{Exp}^{\mathsf{sel\text{-}ind\text{-}fe\text{-}cpa\text{-}}\beta}_{\mathsf{FE},\mathcal{A}}(\lambda)$ for $\beta \in \{0,1\}$.

Definition 6 (Indistinguishability-Based Security). *A functional encryption scheme* FE *is* adaptively *secure (resp.* selectively *secure) against chosen-plaintext attacks if for every PPT algorithm $\mathcal{A}$,* $\mathbf{Adv}^{\mathsf{ind\text{-}fe\text{-}cpa}}_{\mathsf{FE},\mathcal{A}}(\lambda)$ *(resp.* $\mathbf{Adv}^{\mathsf{sel\text{-}ind\text{-}fe\text{-}cpa}}_{\mathsf{FE},\mathcal{A}}(\lambda)$*) is negligible.*

2.3 Bilinear Maps Functionality

In this work we consider functional encryption schemes for the following *bilinear map functionality.* Let $\mathsf{bgp} = (p, \mathbb{G}_1, \mathbb{G}_2, g_1, g_2, \mathbb{G}_T, e) \leftarrow_{\mathrm{R}} \mathcal{G}(1^\lambda)$ be a bilinear group setting, and let $n, m \in \mathbb{N}^+$ be positive integers. We let the message space $\mathcal{M} := \mathbb{Z}_p^n \times \mathbb{Z}_p^m$ – every message M is a pair of vectors $(\boldsymbol{x}, \boldsymbol{y})$ – the key space $\mathcal{K} := \mathbb{Z}_p^{n \times m}$ consists of matrices – every key $K \in \mathcal{K}$ is a matrix $\mathbf{F} = (f_{i,j})$ – and the output space is $\mathcal{Y} := \mathbb{G}_T$. The functionality $F(K, M)$ is the one that computes the value $[\boldsymbol{x}^\top \mathbf{F}\, \boldsymbol{y}]_T \in \mathbb{G}_T$. As we discuss below, this functionality allows for interesting appliations.

Bilinear maps over the integers. We note that for appropriate choices of $\mathcal{M} \subset \mathbb{Z}_p^n \times \mathbb{Z}_p^m$ and $\mathcal{K} \subset \mathbb{Z}_p^{n \times m}$, the output space of $F(\mathcal{K}, \mathcal{M})$ can be made of size polynomial in the security parameter. In this case, there exist efficient methods to extract $\boldsymbol{x}^\top \mathbf{F}\, \boldsymbol{y} \in \mathbb{Z}_p$ from $[\boldsymbol{x}^\top \mathbf{F}\, \boldsymbol{y}]_T \in \mathbb{G}_T$.

For example, one can fix integers $B_x, B_y, B_f \in \mathbb{N}$, and define $\mathcal{M} := \{0, \ldots, B_x\}^n \times \{0, \ldots, B_y\}^m$, $\mathcal{K} := \{0, \ldots, B_f\}^{n \times m}$. Then the quantity $B = mnB_xB_yB_f < p$ must be small enough to allow for efficient discrete logarithm computation.

Multivariate quadratic polynomials. We also note that bilinear maps over the integers capture an interesting class of quadratic functions, such *multivariate quadratic polynomials*:

$$p(\boldsymbol{m}) = p_0 + \sum_i p_i \cdot m_i + \sum_{i,j} p_{i,j} \cdot m_i \cdot m_j.$$

This can be captured by setting $\boldsymbol{x} = \boldsymbol{y} = (1, \boldsymbol{m}) \in \mathbb{Z}_p^{n+1}$ and by encoding p's coefficients in an upper triangular matrix $\mathbf{F} = (f_{i,j}) \in \mathbb{Z}_p^{(n+1)\times(n+1)}$ where: $f_{1,1} = p_0$, $f_{1,i} = p_{i-1}$ for all $i \in [2, n+1]$, $f_{i,j} = 0$ for all $i > j$, and $f_{i,j} = p_{i-1,j-1}$ for all $i \in [2, n+1]$ and $j \geq i$.

2.4 Predicate Encryption

We recall the definition of predicate encryption, as originally defined in [28,29].

Definition 7 (Predicate). *A predicate* P *defined over* $(\mathcal{X}, \mathcal{Y})$ *is a boolean function:* $\mathsf{P} : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$.

Definition 8 (Predicate Encryption). *A predicate encryption (PE) scheme for a predicate* $\mathsf{P} : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$ *consists of four algorithms* (Setup, $\mathsf{Encrypt}$, KeyGen, $\mathsf{Decrypt}$)*:*

$\mathsf{Setup}(1^\lambda, \mathsf{P}, \mathcal{M}) \to (\mathsf{mpk}, \mathsf{msk})$. *The setup algorithm gets as input the security parameter λ, the predicate* $\mathsf{P} : \mathcal{X} \times \mathcal{Y} \to \{0,1\}$*, the message space $\mathcal{M}$ and outputs the public parameter* mpk*, and the master key* msk.

$\mathsf{Encrypt}(\mathsf{mpk}, x, M) \to \mathsf{Ct}_x$. *The encryption algorithm gets as input* mpk*, an attribute $x \in \mathcal{X}$ and a message $M \in \mathcal{M}$. It outputs a ciphertext* Ct_x.

$\mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, y) \to \mathsf{sk}_y$. *The key generation algorithm gets as input* msk *and a value* $y \in \mathcal{Y}$, *and outputs a secret key* sk_y. *Note that* y *is public in* sk_y.

$\mathsf{Decrypt}(\mathsf{mpk}, \mathsf{sk}_y, \mathsf{Ct}_x) \to M$. *The decryption algorithm gets as input* sk_y *and* Ct_x *such that* $\mathsf{P}(x, y) = 1$. *It outputs a message* M.

For correctness, it is requires that for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$ *such that* $\mathsf{P}(x, y) = 1$ *and all* $M \in \mathcal{M}$,

$$\Pr[\mathsf{Decrypt}(\mathsf{mpk}, \mathsf{sk}_y, \mathsf{Encrypt}(\mathsf{mpk}, x, M)) = M] = 1,$$

where the probability is taken over $(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda, \mathcal{X}, \mathcal{Y}, \mathcal{M})$, $\mathsf{sk}_y \leftarrow \mathsf{KeyGen}(\mathsf{mpk}, \mathsf{msk}, y)$, *and the coins of* $\mathsf{Encrypt}$.

Fully Attribute-Hiding Security. We recall the notion of *fully attribute-hiding* security for predicate encryption as defined in [28]. The fully attribute hiding property refers to the fact that an adversary cannot distinguish a ciphertext for attribute $x^{(0)}$ from a ciphertext for $x^{(1)}$, as long as it only queries keys sk_y where $\mathsf{P}(x^{(0)}, y) = \mathsf{P}(x^{(1)}, y)$. This is stronger than the so-called *weakly attribute hiding* property, which requires the adversary to only query keys sk_y where $\mathsf{P}(x^{(0)}, y) = \mathsf{P}(x^{(1)}, y) = 0$.

Fully attribute hiding security is essentially the specialization of the indistinguishability based security notion for functional encryption, for the functionality $F_{\mathsf{P}}(y, (x, M))$ that outputs M if $\mathsf{P}(x, y) = 1$ and $\perp$ otherwise.

For any stateful adversary $\mathcal{A}$, any predicate encryption scheme PE, any bit $\beta \in \{0, 1\}$, and any security parameter λ, we define the experiments $\mathbf{Exp}^{\mathsf{ind\text{-}pe\text{-}cpa\text{-}}\beta}_{\mathsf{PE},\mathcal{A}}(\lambda)$ and $\mathbf{Exp}^{\mathsf{sel\text{-}ind\text{-}pe\text{-}cpa\text{-}}\beta}_{\mathsf{PE},\mathcal{A}}(\lambda)$ in Figure 2. We define the advantage of $\mathcal{A}$ for adaptive security as:

$$\begin{aligned}\mathbf{Adv}^{\mathsf{ind\text{-}pe\text{-}cpa}}_{\mathsf{FE},\mathcal{A}}(\lambda) &:= \left|\Pr[\mathbf{Exp}^{\mathsf{ind\text{-}pe\text{-}cpa\text{-}0}}_{\mathsf{FE},\mathcal{A}}(\lambda) = 1] - \Pr[\mathbf{Exp}^{\mathsf{ind\text{-}pe\text{-}cpa\text{-}1}}_{\mathsf{FE},\mathcal{A}}(\lambda) = 1]\right| \\ &= \left|1 - 2\Pr\left[\beta' = \beta : \begin{array}{l}\beta \leftarrow_{\mathrm{R}} \{0,1\} \\ \mathbf{Exp}^{\mathsf{ind\text{-}pe\text{-}cpa\text{-}}\beta}_{\mathsf{FE},\mathcal{A}}(\lambda) = \beta'\end{array}\right]\right|\end{aligned}$$

We define the advantage $\mathbf{Adv}^{\mathsf{sel\text{-}ind\text{-}pe\text{-}cpa}}_{\mathsf{FE},\mathcal{A}}(\lambda)$ for selective security similarly, with respect to experiments $\mathbf{Exp}^{\mathsf{sel\text{-}ind\text{-}pe\text{-}cpa\text{-}}\beta}_{\mathsf{FE},\mathcal{A}}(\lambda)$ for $\beta \in \{0, 1\}$.

Definition 9 (Fully Attribute-Hiding Security). *A predicate encryption scheme* PE *is fully attribute hiding,* adaptively *secure (resp.* selectively *secure) against chosen-plaintext attacks if for every PPT algorithm* $\mathcal{A}$, $\mathbf{Adv}^{\mathsf{ind\text{-}pe\text{-}cpa}}_{\mathsf{PE},\mathcal{A}}(\lambda)$ *(resp.* $\mathbf{Adv}^{\mathsf{sel\text{-}ind\text{-}pe\text{-}cpa}}_{\mathsf{PE},\mathcal{A}}(\lambda)$*) is negligible.*

3 Our Functional Encryption for Bilinear Maps from MDDH

In this Section we present a functional encryption scheme that supports the bilinear maps functionality described in Section 2.3, and is proven selectively secure under standard assumptions.

To begin with, in Section 3.1 we describe a simple FE scheme that works in the private-key setting, is only single-ciphertext secure, and supports the bilinear maps functionality $F : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$, where $\mathcal{M} \subset \mathbb{Z}_p^n \times \mathbb{Z}_p^m$ and $\mathcal{K} \subset \mathbb{Z}_p^{n \times m}$ are such that for for all $(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{M}$, $\mathbf{F} \in \mathcal{K}$,

$$F(\mathbf{F}, (\boldsymbol{x}, \boldsymbol{y})) = \boldsymbol{x}^\top \mathbf{F} \boldsymbol{y} \in \{0, 1\}.$$

This private-key scheme is used as a building block in the security proof of our main public-key FE scheme that we present in Section 3.2. We stress that our public-key FE scheme supports the bilinear map functionality without the restriction on boolean outputs as above.

```
$\boxed{\mathbf{Exp}^{\mathsf{ind\text{-}pe\text{-}cpa\text{-}}\beta}_{\mathsf{PE},\mathcal{A}}(\lambda)}$, $\boxed{\mathbf{Exp}^{\mathsf{sel\text{-}ind\text{-}pe\text{-}cpa\text{-}}\beta}_{\mathsf{PE},\mathcal{A}}(\lambda)}$ :
$\boxed{(x^{(0)}, M_0, x^{(1)}, M_1) \leftarrow \mathcal{A}(1^\lambda)}$
$(\mathsf{mpk}, \mathsf{msk}) \leftarrow_{\mathrm{R}} \mathsf{Setup}(1^\lambda)$
$\boxed{\mathsf{Ct} := \mathsf{EncO}(x^{(0)}, M_0, x^{(1)}, M_1)}$
$\beta' \leftarrow \mathcal{A}(\mathsf{mpk}, \boxed{\mathsf{Ct}})^{\mathsf{KeyGenO}(\cdot), \boxed{\mathsf{EncO}(\cdot,\cdot,\cdot,\cdot)}}$
Return $\beta'$.

$\underline{\mathsf{EncO}(x^{(0)}, M_0, x^{(1)}, M_1):}$
Return $\mathsf{Ct}^\star := \mathsf{Encrypt}(\mathsf{mpk}, x^{(\beta)}, M_\beta)$

$\underline{\mathsf{KeyGenO}(y \in \mathcal{Y}):}$
Return $\mathsf{sk}_K := \mathsf{KeyGen}(\mathsf{msk}, y)$
```

Fig. 2. Experiments $\mathbf{Exp}^{\mathsf{ind\text{-}pe\text{-}cpa\text{-}}\beta}_{\mathsf{PE},\mathcal{A}}(\lambda)$ and $\mathbf{Exp}^{\mathsf{sel\text{-}ind\text{-}pe\text{-}cpa\text{-}}\beta}_{\mathsf{PE},\mathcal{A}}(\lambda)$ for $b \in \{0,1\}$, used to define adaptive, and selective security of PE, respectively. In each procedure, the components inside a solid (dotted) frame are only present in the games marked by a solid (dotted) frame. In both games, the oracle $\mathsf{EncO}(\cdot,\cdot,\cdot,\cdot)$ is queried at most once (by $\mathcal{A}$ or the game itself), on $x^{(0)}, M_0, x^{(1)}, M_1$, such that for all queries y to $\mathsf{KeyGenO}(\cdot)$, we have: $\mathsf{P}(x^{(0)}, y) = P(x^{(1)}, y)$. Moreover, if $\mathsf{P}(x^{(0)}, y)$ for some query y to $\mathsf{KeyGenO}(\cdot)$, then $M_0 = M_1$.

3.1 Private-key, single-ciphertext secure FE for bilinear maps with boolean ouput

In this section, we present a family of private-key, single-ciphertext secure functional encryption schemes for bilinear maps with boolean outputs, parametrized by an integer $k \geq 1$ and a matrix distribution $\mathcal{D}_k$ (see Definition 1). That is, for each $k \in \mathbb{N}$, and each matrix distribution $\mathcal{D}_k$, the scheme $\mathsf{FE}_{\mathsf{one}}(k, \mathcal{D}_k)$, presented in Figure 3, is single-ciphertext, selectively secure under the $\mathcal{D}_k$-mddh assumption, on asymmetric pairings.

Technical overview. Before describing the scheme in full detail in Figure 3, we give an informal exposition of our techniques. The basic idea in our private-key, single ciphertext secure FE is to create the ciphertext and the secret keys of the form:

$$\mathsf{Ct}_{(\boldsymbol{x},\boldsymbol{y})} := \{[\mathbf{A}\boldsymbol{r}_i + \boldsymbol{b}^\perp x_i]_1\}_{i\in[n]}, \{[\mathbf{B}\boldsymbol{s}_j + \boldsymbol{a}^\perp y_j]_2\}_{j\in[m]},\ \mathsf{sk}_{\mathbf{F}} := [\sum_{i,j} f_{i,j} \boldsymbol{r}_i^\top \mathbf{A}^\top \mathbf{B} \boldsymbol{s}_j]_T,$$

where $\mathbf{A}, \mathbf{B} \leftarrow_{\mathrm{R}} \mathcal{D}_k$, and $(\mathbf{A}|\boldsymbol{b}^\perp)$, $(\mathbf{B}|\boldsymbol{a}^\perp)$ are bases of $\mathbb{Z}_p^{k+1}$ such that $\boldsymbol{a}^\perp \in \mathsf{orth}(\mathbf{A})$ and $\boldsymbol{b}^\perp \in \mathsf{orth}(\mathbf{B})$, à la [16]. The vectors $[\mathbf{A}\boldsymbol{r}_i]_1$ and $[\mathbf{B}\boldsymbol{s}_j]_2$ for $i \in [n], j \in [m]$, $\boldsymbol{a}^\perp$ and $\boldsymbol{b}^\perp$ are part of a master secret key, used to (deterministically) generate $\mathsf{Ct}_{\boldsymbol{x},\boldsymbol{y}}$ and $\mathsf{sk}_{\mathbf{F}}$. Correctness follows from the orthogonal property: decryption computes $\sum_{i,j} f_{i,j} e([\mathbf{A}\boldsymbol{r}_i + \boldsymbol{b}^\perp x_i]_1^\top, [\mathbf{B}\boldsymbol{s}_j + \boldsymbol{a}^\perp y_j]_2) = \mathsf{sk}_{\mathbf{F}} + (\boldsymbol{a}^\perp)^\top \boldsymbol{b}^\perp \cdot [F(\mathbf{F}, (\boldsymbol{x}, \boldsymbol{y}))]_T$ which is equal to $\mathsf{sk}_{\mathbf{F}}$ if, and only if, $F(\mathbf{F}, (\boldsymbol{x}, \boldsymbol{y})) = 0$. Security relies on the $\mathcal{D}_k$-mddh Assumption [18], which stipulates that given $[\mathbf{A}]_1, [\mathbf{B}]_2$ drawn from a matrix distribution $\mathcal{D}_k$ over $\mathbb{Z}_p^{(k+1)\times k}$,

$$[\mathbf{A}\boldsymbol{r}]_1 \approx_c [\boldsymbol{u}]_1 \approx_c [\mathbf{A}\boldsymbol{r} + \boldsymbol{b}^\perp]_1 \text{ and } [\mathbf{B}\boldsymbol{s}]_2 \approx_c [\boldsymbol{v}]_2 \approx_c [\mathbf{B}\boldsymbol{s} + \boldsymbol{a}^\perp]_2,$$

where $\boldsymbol{r}, \boldsymbol{s} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$, and $\boldsymbol{u}, \boldsymbol{v} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^{k+1}$. This allows to change $\mathsf{Ct}_{(\boldsymbol{x}^{(0)},\boldsymbol{y}^{(0)})}$ into $\mathsf{Ct}_{(\boldsymbol{x}^{(1)},\boldsymbol{y}^{(1)})}$, but creates an extra term $\left[\boldsymbol{x}^{(1)\top}\mathbf{F}\boldsymbol{y}^{(1)} - \boldsymbol{x}^{(0)\top}\mathbf{F}\boldsymbol{y}^{(0)}\right]_T$ in the secret keys $\mathsf{sk}_{\mathbf{F}}$. We conclude the proof using the fact that for all $\mathbf{F}$ queried to KeyGen, $F(\mathbf{F}, (\boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)})) = F(\mathbf{F}, (\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)}))$, as required by the security definition for FE (see Section 2.2 for the definition of FE), which cancels out the extra term in all secret keys.

In the following theorem we prove the correctness of the scheme $\mathsf{FE}_{\mathsf{one}}$.

Theorem 1 (Correctness). *For any $k \in \mathbb{N}^*$ and any matrix distribution $\mathcal{D}_k$, the functional encryption scheme $\mathsf{FE}_{\mathsf{one}}(k, \mathcal{D}_k)$ defined in Figure 3 has perfect correctness.*

```
Setup($1^\lambda, F$):
    $\mathsf{bgp} \leftarrow_{\mathrm{R}} \mathcal{G}(1^\lambda)$, $\mathbf{A}, \mathbf{B} \leftarrow_{\mathrm{R}} \mathcal{D}_k$;
    $\boldsymbol{a}^\perp \in \mathsf{orth}(\mathbf{A}), \boldsymbol{b}^\perp \in \mathsf{orth}(\mathbf{B})$ s.t. $(\boldsymbol{b}^\perp)^\top \boldsymbol{a}^\perp = 1$
    For $i \in [n], j \in [m]$, $\boldsymbol{r}_i, \boldsymbol{s}_j \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$, $r, s \leftarrow_{\mathrm{R}} \mathbb{Z}_p$.
    Return $\mathsf{mpk} := \mathsf{bgp}$ and
    $\mathsf{msk} := \Big(\mathbf{A}, \boldsymbol{a}^\perp, \mathbf{B}, \boldsymbol{b}^\perp, \{\boldsymbol{r}_i, \boldsymbol{s}_j, r, s\}_{i \in [n], j \in [m]}\Big)$

KeyGen($\mathsf{msk}, \mathbf{F} \in \mathbb{Z}_p^{n \times m}$):
    $K := [\sum_{i \in [n], j \in [m]} f_{i,j} \boldsymbol{r}_i^\top \mathbf{A}^\top \mathbf{B} \boldsymbol{s}_j]_1 - [u]_1$, $\widehat{K} := [u]_2$, where $u \leftarrow_{\mathrm{R}} \mathbb{Z}_p$
    Return $\mathsf{sk}_\mathbf{F} := (K, \widehat{K}) \in \mathbb{G}_1 \times \mathbb{G}_2$

Encrypt($\mathsf{mpk}, \mathsf{msk}, (\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{Z}_p^n \times \mathbb{Z}_p^m$):
    For $i \in [n]$: $\boldsymbol{c}_i := \mathbf{A}\boldsymbol{r}_i + \boldsymbol{b}^\perp r x_i$,
    For $j \in [m]$: $\widehat{\boldsymbol{c}}_j := \mathbf{B}\boldsymbol{s}_j + \boldsymbol{a}^\perp s y_j$,
    $\mathsf{Ct}_{(\boldsymbol{x},\boldsymbol{y})} := \{[\boldsymbol{c}_i]_1, [\widehat{\boldsymbol{c}}_j]_2\}_{i \in [n], j \in [m]}$
    Return $\mathsf{Ct}_{(\boldsymbol{x},\boldsymbol{y})} \in \mathbb{G}_1^{n(k+1)} \times \mathbb{G}_2^{m(k+1)}$

Decrypt($\mathsf{mpk}, \mathsf{Ct}_{(\boldsymbol{x},\boldsymbol{y})}, \mathsf{sk}_\mathbf{F}$):
    Return the boolean: $\sum_{i \in [n], j \in [m]} f_{i,j} \cdot e([\boldsymbol{c}_i^\top]_1, [\widehat{\boldsymbol{c}}_j]_2) \overset{?}{=} e(K, [1]_2) + e([1]_1, \widehat{K})$.
```

Fig. 3. $\mathsf{FE}_{\mathsf{one}}(k, \mathcal{D}_k)$, a family of private-key, functional encryption schemes parametrized by $k \in \mathbb{N}^*$ and a matrix distribution $\mathcal{D}_k$, single-ciphertext, selectively secure under the $\mathcal{D}_k$-mddh assumption on asymmetric pairings.

Proof of Theorem 1. To prove correctness, we first use the fact that for any matrix distribution $\mathcal{D}_k$, by Property 1 of Definition 1, with probability $1 - \frac{1}{\Omega(p)}$ over the choices of $\mathbf{A}, \mathbf{B} \leftarrow_{\mathrm{R}} \mathcal{D}_k$, we have: $\mathsf{orth}(\mathbf{A}) \nsubseteq \mathsf{span}(\mathbf{B})$. Thus, there exist vectors $\boldsymbol{a}^\perp \in \mathsf{orth}(\mathbf{A})$, $\boldsymbol{b}^\perp \in \mathsf{orth}(\mathbf{B})$ such that $(\boldsymbol{b}^\perp)^\top \boldsymbol{a}^\perp = 1$. Note that we can sample these vectors efficiently given $\mathbf{A}, \mathbf{B} \in \mathbb{Z}_p^{(k+1) \times k}$.

Then, we use the fact that for all $i \in [n], j \in [m]$,

$$e([\boldsymbol{c}_i^\top]_1, [\widehat{\boldsymbol{c}}_j]_2) = [\boldsymbol{r}_i^\top \mathbf{A}^\top \mathbf{B} \boldsymbol{s}_j + \underbrace{(\boldsymbol{a}^\perp)^\top \boldsymbol{b}^\perp}_{=1} r s x_i y_j]_T,$$

since $\mathbf{A}^\top \boldsymbol{a}^\perp = \mathbf{B}^\top \boldsymbol{b}^\perp = \mathbf{0}$. Therefore, the decryption gets

$$[\sum_{i,j} f_{i,j} \boldsymbol{r}_i^\top \mathbf{A}^\top \mathbf{B} \boldsymbol{s}_j + rs \cdot \sum_{i,j} f_{i,j} x_i y_j]_T = e(K, [1]_2) - e([1]_1, \widehat{K}) + rs[\sum_{i,j} f_{i,j} x_i y_j]_T,$$

which allows to check if $\sum_{i,j} f_{i,j} x_i y_j$ is 0. □

Next, we show that $\mathsf{FE}_{\mathsf{one}}$ is selective-secure, for adversaries that make a single challenge encryption query, under the MDDH assumption.

Theorem 2 (Security). *For any $k \in \mathbb{N}^*$ and any matrix distribution $\mathcal{D}_k$, if the $\mathcal{D}_k$-mddh assumptions hold in $\mathbb{G}_1$ and $\mathbb{G}_2$, then the functional encryption scheme $\mathsf{FE}_{\mathsf{one}}(k, \mathcal{D}_k)$ defined in Figure 3 is selectively secure, in a single-ciphertext setting (see Definition 6). Namely, for any PPT adversary $\mathcal{A}$, there exists a PPT adversary $\mathcal{B}$ such that:*

$$\mathbf{Adv}^{\mathsf{sel\text{-}ind\text{-}fe\text{-}cpa}}_{\mathsf{FE}_{\mathsf{one}}, \mathcal{A}}(\lambda) \leq 6 \cdot \mathbf{Adv}^{\mathcal{D}_k\text{-}\mathsf{mddh}}_{\mathcal{G}, \mathcal{B}}(\lambda) + \frac{2}{p}.$$

Proof of Theorem 2. We prove the security of $\mathsf{FE}_{\mathsf{one}}(k, \mathcal{D}_k)$ via a series of games that is compactly presented in Figure 4. Before going to the details of the proof and proving the indistinguishability of each consecutive pair of games, we provide below a high level view of the game transitions:

Game G_0 is the selective security experiment for scheme $\mathsf{FE}_{\mathsf{one}}$.

```
$\mathrm{G}_0, \boxed{\mathrm{G}_1}, \bbox[1px dashed]{\mathrm{G}_2, \mathrm{G}_3,}, \colorbox{lightgray}{$\mathrm{G}_4$}$:
$(\boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}), (\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})) \leftarrow \mathcal{A}(1^\lambda)$
$\mathsf{mpk} := \mathsf{bgp} \leftarrow_{\mathrm{R}} \mathcal{G}(1^\lambda)$; $\mathbf{A}, \mathbf{B} \leftarrow_{\mathrm{R}} \mathcal{D}_k$; $\beta \leftarrow_{\mathrm{R}} \{0,1\}$;
$\boldsymbol{a}^\perp \in \mathsf{orth}(\mathbf{A}), \boldsymbol{b}^\perp \in \mathsf{orth}(\mathbf{B})$ s.t. $(\boldsymbol{b}^\perp)^\top \boldsymbol{a}^\perp = 1$
For $i \in [n], j \in [m]$: $\boldsymbol{r}_i \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$, $\boldsymbol{s}_j \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$, $\boldsymbol{h} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^{k+1}$, $r, s \leftarrow_{\mathrm{R}} \mathbb{Z}_p$, $w \leftarrow_{\mathrm{R}} \mathbb{Z}_p$
$\boldsymbol{c}_i := \mathbf{A}\boldsymbol{r}_i + rx_i^{(\beta)}\boldsymbol{b}^\perp$; $\boxed{\boldsymbol{c}_i := \mathbf{A}\boldsymbol{r}_i + x_i^{(\beta)}\boldsymbol{h}}$; $\bbox[1px dashed]{\boldsymbol{c}_i := \mathbf{A}\boldsymbol{r}_i}$
$\widehat{\boldsymbol{c}}_j := \mathbf{B}\boldsymbol{s}_j + sy_j^{(\beta)}\boldsymbol{a}^\perp$; $\colorbox{lightgray}{$\widehat{\boldsymbol{c}}_j := \mathbf{B}\boldsymbol{s}_j$}$
$\mathsf{Ct}^\star := \{[\boldsymbol{c}_i]_1, [\widehat{\boldsymbol{c}}_j]_1\}_{i\in[n], j\in[m]}$
$\beta' \leftarrow \mathcal{A}^{\mathsf{KeyGenO}(\cdot)}(\mathsf{mpk}, \mathsf{Ct}_{(\boldsymbol{x}^{(\beta)}, \boldsymbol{y}^{(\beta)})})$
Return 1 if $\beta' = \beta$, 0 otherwise.

$\underline{\mathsf{KeyGenO}(\mathbf{F} \in \mathbb{Z}_p^{n\times m})}$:                $\mathrm{G}_0, \mathrm{G}_1, \mathrm{G}_2, \colorbox{lightgray}{$\mathrm{G}_3, \mathrm{G}_4$}$
$K := [u]_1 \leftarrow_{\mathrm{R}} \mathbb{G}_1$; $\widehat{K} := [\sum_{i,j} f_{i,j}\boldsymbol{c}_i^\top \widehat{\boldsymbol{c}}_j]_2 - [u]_2 - (rs + \colorbox{lightgray}{$w$}) \cdot [\sum_{i,j} f_{i,j} x_i^{(\beta)} y_j^{(\beta)}]_2$
Return $\mathsf{sk}_{\mathbf{F}} := (K, \widehat{K})$
```

Fig. 4. Games G_i, for $i = 0, \ldots, 4$ for the proof of selective security of $\mathsf{FE}_{\mathsf{one}}(k, \mathcal{D}_k)$ in Figure 3. In each procedure, the components inside a solid (dotted, gray) frame are only present in the games marked by a solid (dotted, gray) frame.

Game G_1 is the same as game G_0 except that in the $\boldsymbol{c}_i$ ciphertext components we replace the vector $r \cdot \boldsymbol{b}^\perp$ with a fresh vector $\boldsymbol{h} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^{k+1}$. In Lemma 1 we show that G_0 is computationally indistinguishable from G_1 under the MDDH assumption.

Game G_2 is the same as game G_1 except that the $\boldsymbol{c}_i$ ciphertext components encrypt $\mathbf{0}$. In Lemma 2 we show that G_1 is computationally indistinguishable from G_2 under the MDDH assumption.

Game G_3 is the same as game G_2 except that in the secret keys we switch the value rs used for computing $\widehat{K} := [\sum_{i,j} f_{i,j}\boldsymbol{c}_i^\top \widehat{\boldsymbol{c}}_j - rs\sum_{i,j} f_{i,j}x_i^{(\beta)}y_j^{(\beta)} - u]_2$ into $rs + w$, for a fresh $w \leftarrow_{\mathrm{R}} \mathbb{Z}_p$. In Lemma 3 we use a statistical argument to show that G_2 is negligibly close to G_3.

Game G_4 : here we change the $\widehat{\boldsymbol{c}}_j$ ciphertext components so that they encrypt $\mathbf{0}$ instead of $\boldsymbol{y}^{(\beta)}$. In Lemma 4 we use the MDDH assumption to show that G_4 is computationally indistinguishable from G_3. Finally, in Lemma 5 we argue that the adversary's view in this game is independent of the bit β, and thus the adversary's advantage in this game is zero.

More formally, in what follows we use Adv_i to denote the advantage of $\mathcal{A}$ in game G_i, that is $\mathsf{Adv}_i := |1 - 2\Pr[\mathrm{G}_i \text{ returns } 1]|$. Note that G_0 is defined as:

$$\mathrm{G}_0 : \begin{array}{l} \beta \leftarrow_{\mathrm{R}} \{0,1\} \\ \beta' \leftarrow \mathbf{Exp}^{\mathsf{sel\text{-}ind\text{-}fe\text{-}cpa}\text{-}\beta}_{\mathsf{FE}_{\mathsf{one}},\mathcal{A}}(\lambda) \\ \quad \text{Return 1 if } \beta' = \beta, \text{ 0 otherwise.} \end{array}$$

Where $\mathbf{Exp}^{\mathsf{sel\text{-}ind\text{-}fe\text{-}cpa}\text{-}\beta}_{\mathsf{FE}_{\mathsf{one}},\mathcal{A}}(\lambda)$ is the experiment used in Definition 6 of indistinguishability-based security for functional encryption. In particular, we have

$$\mathsf{Adv}_0 = \mathbf{Adv}^{\mathsf{sel\text{-}ind\text{-}fe\text{-}cpa}}_{\mathsf{FE}_{\mathsf{one}},\mathcal{A}}(\lambda).$$

Lemma 1 (G_0 to G_1). *There exists a PPT adversary $\mathcal{B}_0$ such that:*

$$|\mathsf{Adv}_0 - \mathsf{Adv}_1| \leq 2 \cdot \mathbf{Adv}^{\mathcal{D}_k\text{-}\mathsf{mddh}}_{\mathcal{G},\mathcal{B}_0}(\lambda).$$

Proof of Lemma 1. Here, we use the mddh assumption on $[\mathbf{A}]_1$ to change the distribution of the challenge ciphertext, after arguing that one can simulate the game without knowing $\boldsymbol{a}^\perp$ or $[\mathbf{A}]_2$.

Namely, we build a PPT adversary $\mathcal{B}_0$ as described in Figure 5, and we prove that $|\mathsf{Adv}_0 - \mathsf{Adv}_1| \leq \mathbf{Adv}_{\mathcal{G},\mathcal{B}_0}^{\mathcal{D}_k\text{-}\mathsf{mddh}}(\lambda)$.

```
$\mathcal{B}_0(\mathsf{bgp}, [\mathbf{A}]_1, [\boldsymbol{h}]_1)$:
$\big((\boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}), (\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})\big) \leftarrow \mathcal{A}(1^\lambda)$
$\mathsf{mpk} := \mathsf{bgp}$, $\mathbf{B} \leftarrow_{\mathrm{R}} \mathcal{D}_k$; $\beta \leftarrow_{\mathrm{R}} \{0,1\}$; $\boldsymbol{b}^\perp \leftarrow_{\mathrm{R}} \mathsf{orth}(\mathbf{B})$
For $i \in [n], j \in [m]$: $\boldsymbol{r}_i \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$, $\boldsymbol{s}_j \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$, $\boldsymbol{z} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^{k+1}$, $r \leftarrow_{\mathrm{R}} \mathbb{Z}_p$
$\boldsymbol{c}_i := \mathbf{A}\boldsymbol{r}_i + x_i^{(\beta)}\boldsymbol{h} + r x_i^{(\beta)}\boldsymbol{b}^\perp$
$\widehat{\boldsymbol{c}}_j := \mathbf{B}\boldsymbol{s}_j + y_j^{(\beta)}\boldsymbol{z}$;
$\mathsf{Ct}^\star := \{[\boldsymbol{c}_i]_1, [\widehat{\boldsymbol{c}}_j]_2\}_{i\in[n], j\in[m]}$
$\beta' \leftarrow \mathcal{A}^{\mathsf{KeyGenO}(\cdot)}(\mathsf{mpk}, \mathsf{Ct}^\star)$
Return 1 if $\beta = \beta'$, 0 otherwise.

$\mathsf{KeyGenO}(\mathbf{F} \in \mathbb{Z}_p^{n\times m})$:
$u \leftarrow_{\mathrm{R}} \mathbb{Z}_p$, $\widehat{K} := [u]_2$;
$K := [\sum_{i,j} f_{i,j}\boldsymbol{c}_i^\top \widehat{\boldsymbol{c}}_j - \sum_{i,j} f_{i,j} x_i^{(\beta)} r (\boldsymbol{b}^\perp)^\top \widehat{\boldsymbol{c}}_j]_1 - [u]_1$
Return $\mathsf{sk}_{\mathbf{F}} := (K, \widehat{K})$
```

Fig. 5. Adversary $\mathcal{B}_0$ against the $\mathcal{D}_k$-MDDH assumption, for the proof of Lemma 1.

First, assume that $\mathcal{B}_0$ is given a real mddh challenge, that is, $[\boldsymbol{h}]_1 := [\mathbf{A}\boldsymbol{r}]_1$ for $\boldsymbol{r} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$. We show that in that case, $\mathcal{B}_0$ simulates G_0.

-Simulation of $\mathsf{Ct}^\star$: First, we use the fact that for all $x_i^{(\beta)} \in \mathbb{Z}_p$, $\boldsymbol{r} \in \mathbb{Z}_p^k$, the following distributions are equal: $\{\boldsymbol{r}_i\}_{i\in[n]}$ and $\{\boldsymbol{r}_i + x_i^{(\beta)}\boldsymbol{r}\}_{i\in[n]}$, where $\boldsymbol{r}_i \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$. Therefore, we can argue that $\{\boldsymbol{c}_i := \mathbf{A}(\boldsymbol{r}_i + x_i^{(\beta)}\boldsymbol{r}) + r x_i^{(\beta)}\boldsymbol{b}^\perp]_1\}_{i\in[n]}$, is identically distributed to $\{\boldsymbol{c}_i := \mathbf{A}\boldsymbol{r}_i + r x_i^{(\beta)}\boldsymbol{b}^\perp\}_{i\in[n]}$ as in G_0. Note that here, we are relying on the fact the games are *single-ciphertext*, and challenge (x_i^0, x_i^1) is *independent* from the vectors $\{\boldsymbol{r}_i\}_{i\in[n]}$, since the games here are *selective*.

Then, we use the fact that $\boldsymbol{z} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^{k+1}$ is identically distributed to $\mathbf{B}\boldsymbol{t} + s\boldsymbol{a}^\perp$, where $\boldsymbol{t} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$ and $s \leftarrow_{\mathrm{R}} \mathbb{Z}_p$, since $(\mathbf{B}|\boldsymbol{a}^\perp)$ is a basis of $\mathbb{Z}_p^{k+1}$ (this is implied by the fact that $(\boldsymbol{b}^\perp)^\top\boldsymbol{a}^\perp \neq 0$, for $\boldsymbol{b}^\perp \in \mathsf{orth}(\mathbf{B})$). This allows to write $\{\widehat{\boldsymbol{c}}_j := \mathbf{B}(\boldsymbol{s}_j + y_j^{(\beta)}\boldsymbol{t}) + s y_j^{(\beta)}\boldsymbol{a}^\perp\}_{j\in[m]}$, which is identically distributed to $\{\widehat{\boldsymbol{c}}_j := \mathbf{B}\boldsymbol{s}_j + s y_j^{(\beta)}\boldsymbol{a}^\perp\}_{j\in[m]}$ as in G_0.

-Simulation of $\mathsf{KeyGenO}(\mathbf{F} \in \mathbb{Z}_p^{n\times m})$: As we argued previously, $\{\boldsymbol{c}_i, \widehat{\boldsymbol{c}}_j\}_{i\in[n],j\in[m]}$ simulated by $\mathcal{B}_0$ are identically distributed to those in G_0. Therefore, we have:

$$\begin{aligned}
K &:= [\sum_{i,j} f_{i,j}\boldsymbol{c}_i^\top\widehat{\boldsymbol{c}}_j - \sum_{i,j} f_{i,j}x_i^{(\beta)} r(\boldsymbol{b}^\perp)^\top\widehat{\boldsymbol{c}}_j]_1 - [u]_1 \\
&= [\sum_{i,j} f_{i,j}\boldsymbol{c}_i^\top\widehat{\boldsymbol{c}}_j - \sum_{i,j} f_{i,j}x_i^{(\beta)} r(\boldsymbol{b}^\perp)^\top(\mathbf{B}\boldsymbol{s}_j + s y_j^{(\beta)}\boldsymbol{a}^\perp)]_1 - [u]_1 \\
&= [\sum_{i,j} f_{i,j}\boldsymbol{c}_i^\top\widehat{\boldsymbol{c}}_j - rs\sum_{i,j} f_{i,j}x_i^{(\beta)} y_j^{(\beta)}]_1 - [u]_1,
\end{aligned}$$

and $\widehat{K} := [u]_2$, which is distributed exactly as in G_0.

Now, assume that $\mathcal{B}_0$ is given a uniform challenge, that is, $[\boldsymbol{h}]_1 \leftarrow_{\mathrm{R}} \mathbb{G}_1^{k+1}$. We show that in that case, $\mathcal{B}_0$ simulates G_1.

-Simulation of $\mathsf{Ct}^\star$: First, we use the fact that for all $\boldsymbol{b}^\perp \in \mathbb{Z}_p^{k+1}$, the following distributions are equal: $\boldsymbol{h} + \boldsymbol{b}^\perp$ and $\boldsymbol{z}$, where $\boldsymbol{h}, \boldsymbol{z} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^{k+1}$. Therefore, we can argue that $\{\boldsymbol{c}_i := \mathbf{A}\boldsymbol{r}_i + x_i^{(\beta)}\boldsymbol{z}\}_{i\in[n]}$, as in G_1.

Then, since the vectors $\{\widehat{\boldsymbol{c}}_j\}_{j\in[m]}$ are identically distributed in G_0 and G_1, we can argue exactly as before (for the case where $\mathcal{B}_0$ is given a real mddh challenge), that the $\{\widehat{\boldsymbol{c}}_j\}_{j\in[m]}$ are distributed as in G_1.

-Simulation of $\mathsf{KeyGenO}(\mathbf{F} \in \mathbb{Z}_p^{n\times m})$: As we argued previously for the case where $\mathcal{B}_0$ is given a real mddh challenge, the key computed by $\mathcal{B}_0$ is of the form: $K := [\sum_{i,j} f_{i,j}\boldsymbol{c}_i^\top \widehat{\boldsymbol{c}}_j - rs\sum_{i,j} f_{i,j}x_i^{(\beta)}y_j^{(\beta)}]_1 - [u]_1$, as in G_1. □

Lemma 2 **(G_1 to G_2).** *There exists a PPT adversary $\mathcal{B}_1$ such that:*

$$|\mathsf{Adv}_1 - \mathsf{Adv}_2| \leq 2 \cdot \mathbf{Adv}_{\mathcal{G},\mathcal{B}_1}^{\mathcal{D}_k\text{-}\mathsf{mddh}}(\lambda).$$

Proof of Lemma 2. Here, we use the mddh assumption on $[\mathbf{A}]_1$ to change the distribution of the challenge ciphertext, after arguing that one can simulate the game without knowing $\boldsymbol{a}^\perp$ or $[\mathbf{A}]_2$, as for the previous transition.

Namely, we build a PPT adversary $\mathcal{B}_1$ as described in Figure 6, and we prove that $|\mathsf{Adv}_1 - \mathsf{Adv}_2| \leq \mathbf{Adv}_{\mathcal{G},\mathcal{B}_1}^{\mathcal{D}_k\text{-}\mathsf{mddh}}(\lambda)$.

```
$\mathcal{B}_1(\mathsf{bgp}, [\mathbf{A}]_1, [\boldsymbol{h}]_1)$:
$((\boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}), (\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})) \leftarrow \mathcal{A}(1^\lambda)$
$\mathsf{mpk} := \mathsf{bgp}$, $\mathbf{B} \leftarrow_{\mathrm{R}} \mathcal{D}_k$; $\beta \leftarrow_{\mathrm{R}} \{0,1\}$; $\boldsymbol{b}^\perp \leftarrow_{\mathrm{R}} \mathsf{orth}(\mathbf{B})$
For $i \in [n], j \in [m]$: $\boldsymbol{r}_i \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$, $\boldsymbol{s}_j \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$, $\boldsymbol{z} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^{k+1}$, $r \leftarrow_{\mathrm{R}} \mathbb{Z}_p$
$\boldsymbol{c}_i := \mathbf{A}\boldsymbol{r}_i + x_i^{(\beta)}\boldsymbol{h}$
$\widehat{\boldsymbol{c}}_j := \mathbf{B}\boldsymbol{s}_j + y_j^{(\beta)}\boldsymbol{z}$;
$\mathsf{Ct}^\star := \{[\boldsymbol{c}_i]_1, [\widehat{\boldsymbol{c}}_j]_2\}_{i\in[n],j\in[m]}$
$\beta' \leftarrow \mathcal{A}^{\mathsf{KeyGenO}(\cdot)}(\mathsf{mpk}, \mathsf{Ct}^\star)$
Return 1 if $\beta = \beta'$, 0 otherwise.

$\mathsf{KeyGenO}(\mathbf{F} \in \mathbb{Z}_p^{n\times m})$:
$u \leftarrow_{\mathrm{R}} \mathbb{Z}_p$, $\widehat{K} := [u]_2$;
$K := [\sum_{i,j} f_{i,j}\boldsymbol{c}_i^\top\widehat{\boldsymbol{c}}_j - \sum_{i,j} f_{i,j}x_i^{(\beta)}r(\boldsymbol{b}^\perp)^\top\widehat{\boldsymbol{c}}_j]_1 - [u]_1$
Return $\mathsf{sk}_{\mathbf{F}} := (K, \widehat{K})$
```

Fig. 6. Adversary $\mathcal{B}_1$ against the $\mathcal{D}_k$-MDDH assumption, for the proof of Lemma 2.

First, assume that $\mathcal{B}_1$ is given a real mddh challenge, that is, $[\boldsymbol{h}]_1 := [\mathbf{A}\boldsymbol{r}]_1$ for $\boldsymbol{r} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$. We show that in that case, $\mathcal{B}_1$ simulates G_2.

-Simulation of $\mathsf{Ct}^\star$: First, we use the fact that for all $x_i^{(\beta)} \in \mathbb{Z}_p$, $\boldsymbol{r} \in \mathbb{Z}_p^k$, the following distributions are equal: $\{\boldsymbol{r}_i\}_{i\in[n]}$ and $\{\boldsymbol{r}_i + x_i^{(\beta)}\boldsymbol{r}\}_{i\in[n]}$, where $\boldsymbol{r}_i \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$. Therefore, we can argue that $\{\boldsymbol{c}_i :=$

$\mathbf{A}(\boldsymbol{r}_i + x_i^{(\beta)}\boldsymbol{r})]_1\}_{i\in[n]}$, is identically distributed to $\{\boldsymbol{c}_i := \mathbf{A}\boldsymbol{r}_i\}_{i\in[n]}$ as in G_2. Note that here, we are relying on the fact the the games are *single-ciphertext*, and the challenge (x_i^0, x_i^1) is *independent* from the vectors $\{\boldsymbol{r}_i\}_{i\in[n]}$, since the games here are *selective*.

Then, since the vectors $\{\widehat{\boldsymbol{c}}_j\}_{j\in[m]}$ are identically distributed in G_1 and G_2, we can argue exactly as before (for the transition from G_0 to G_1, see Lemma 1), that the $\{\widehat{\boldsymbol{c}}_j\}_{j\in[m]}$ are distributed as in G_2.

-Simulation of $\mathsf{KeyGenO}(\mathbf{F} \in \mathbb{Z}_p^{n\times m})$: The key K is exactly as in G_2, namely, of the form:

$$\begin{aligned} K &:= [\sum_{i,j} f_{i,j}\boldsymbol{c}_i^\top\widehat{\boldsymbol{c}}_j - \sum_{i,j} f_{i,j}x_i^{(\beta)}r(\boldsymbol{b}^\perp)^\top\widehat{\boldsymbol{c}}_j]_1 - [u]_1 \\ &= [\sum_{i,j} f_{i,j}\boldsymbol{c}_i^\top\widehat{\boldsymbol{c}}_j - \sum_{i,j} f_{i,j}x_i^{(\beta)}r(\boldsymbol{b}^\perp)^\top(\mathbf{B}\boldsymbol{s}_j + sy_j^{(\beta)}\boldsymbol{a}^\perp)]_1 - [u]_1 \\ &= [\sum_{i,j} f_{i,j}\boldsymbol{c}_i^\top\widehat{\boldsymbol{c}}_j - rs\sum_{i,j} f_{i,j}x_i^{(\beta)}y_j^{(\beta)}]_1 - [u]_1, \end{aligned}$$

where the $\{\boldsymbol{c}_i\}_{i\in[n]}$ are distributed as in G_2.

Now, assume that $\mathcal{B}_1$ is given a uniform challenge, that is, $[\boldsymbol{h}]_1 \leftarrow_{\mathrm{R}} \mathbb{G}_1^{k+1}$. In that case, it is clear that $\mathcal{B}_1$ simulates G_1. □

Lemma 3 **(G_2 to G_3).** $|\mathsf{Adv}_2 - \mathsf{Adv}_3| \leq \frac{2}{p}$.

Proof of Lemma 3. Here, we change the distribution of the keys computed by $\mathsf{KeyGenO}$, using a statistical argument.

Namely, we use the fact that the following distributions are $\frac{1}{p}$-close: (s, rs) and $(s, rs + w)$, where $r, s, w \leftarrow_{\mathrm{R}} \mathbb{Z}_p$. This allows to switch the value rs used when computing $\widehat{K} := [\sum_{i,j} f_{i,j}\boldsymbol{c}_i^\top\widehat{\boldsymbol{c}}_j - rs\sum_{i,j} f_{i,j}x_i^{(\beta)}y_j^{(\beta)}]_2 - [u]_2$ to $rs + w$, where $w \leftarrow_{\mathrm{R}} \mathbb{Z}_p$. □

Lemma 4 **(G_3 to G_4).** *There exists a PPT adversary $\mathcal{B}_3$ such that:*

$$|\mathsf{Adv}_3 - \mathsf{Adv}_4| \leq 2 \cdot \mathbf{Adv}_{\mathcal{G},\mathcal{B}_3}^{\mathcal{D}_k\text{-}\mathsf{mddh}}(\lambda).$$

Proof of Lemma 4. Here, we use the mddh assumption on $[\mathbf{B}]_2$ to change the distribution of the challenge ciphertext, after arguing that one can simulate the game without knowing $\boldsymbol{b}^\perp$ or $[\mathbf{B}]_1$.

Namely, we build a PPT adversary $\mathcal{B}_3$ as described in Figure 7, and we prove that $|\mathsf{Adv}_3 - \mathsf{Adv}_4| \leq \mathbf{Adv}_{\mathcal{G},\mathcal{B}_3}^{\mathcal{D}_k\text{-}\mathsf{mddh}}(\lambda)$.

First, assume that $\mathcal{B}_3$ is given a real mddh challenge, that is, $[\boldsymbol{h}]_2 := [\mathbf{B}\boldsymbol{s}]_2$ for $\boldsymbol{s} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$. We show that in that case, $\mathcal{B}_3$ simulates G_4.

-Simulation of $\mathsf{Ct}^\star$: We have $\{\boldsymbol{c}_i := \mathbf{A}\boldsymbol{r}_i, \widehat{\boldsymbol{c}}_j := \mathbf{B}\boldsymbol{s}_j\}_{i\in[n],j\in[m]}$, as in G_4.

-Simulation of $\mathsf{KeyGenO}(\mathbf{F} \in \mathbb{Z}_p^{n\times m})$: Using the fact that the following distributions are equal: $(s, rs + w)$, and (s, v), where $r, s, v, w \leftarrow_{\mathrm{R}} \mathbb{Z}_p$, we can argue that the key $\widehat{K} := [\sum_{i,j} f_{i,j}\boldsymbol{c}_i^\top\widehat{\boldsymbol{c}}_j - v\sum_{i,j} f_{i,j}x_i^{(\beta)}y_j^{(\beta)}]_1 - [u]_1$ is identically distributed than in G_4.

Now, assume that $\mathcal{B}_3$ is given a uniform challenge, that is, $[\boldsymbol{h}]_2 \leftarrow_{\mathrm{R}} \mathbb{G}_2^{k+1}$. We show that in that case, $\mathcal{B}_3$ simulates G_3.

```
$\mathcal{B}_3(\mathsf{bgp}, [\mathbf{B}]_2, [\boldsymbol{h}]_2)$:
$((\boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}), (\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})) \leftarrow \mathcal{A}(1^\lambda)$
$\mathsf{mpk} := \mathsf{bgp}$, $\mathbf{A} \leftarrow_{\mathrm{R}} \mathcal{D}_k$; $\beta \leftarrow_{\mathrm{R}} \{0,1\}$
For $i \in [n], j \in [m]$: $\boldsymbol{r}_i \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$, $\boldsymbol{s}_j \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$, $v \leftarrow_{\mathrm{R}} \mathbb{Z}_p$
$\boldsymbol{c}_i := \mathbf{A}\boldsymbol{r}_i$
$\widehat{\boldsymbol{c}}_j := \mathbf{B}\boldsymbol{s}_j + y_j^{(b)}\boldsymbol{h}$;
$\mathsf{Ct}^\star := \{[\boldsymbol{c}_i]_1, [\widehat{\boldsymbol{c}}_j]_2\}_{i\in[n],j\in[m]}$
$\beta' \leftarrow \mathcal{A}^{\mathsf{KeyGenO}(\cdot)}(\mathsf{mpk}, \mathsf{Ct}^\star)$
Return 1 if $\beta = \beta'$, 0 otherwise.

$\mathsf{KeyGenO}(\mathbf{F} \in \mathbb{Z}_p^{n\times m})$:
$u \leftarrow_{\mathrm{R}} \mathbb{Z}_p$, $K := [u]_1$;
$\widehat{K} := [\sum_{i,j} f_{i,j}\boldsymbol{c}_i^\top\widehat{\boldsymbol{c}}_j - v\sum_{i,j} f_{i,j}x_i^{(b)}y_j^{(b)}]_2 - [u]_2$
Return $\mathsf{sk}_{\mathbf{F}} := (K, \widehat{K})$
```

Fig. 7. Adversary $\mathcal{B}_3$ against the $\mathcal{D}_k$-MDDH assumption, for the proof of Lemma 4.

-Simulation of $\mathsf{Ct}^\star$: We use the fact that $\boldsymbol{z} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^{k+1}$ is identically distributed to $\mathbf{B}\boldsymbol{t} + s\boldsymbol{a}^\perp$, where $\boldsymbol{t} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$, $s \leftarrow_{\mathrm{R}} \mathbb{Z}_p$, and $\boldsymbol{a}^\perp \in \mathsf{orth}(\mathbf{A})$ such that $\boldsymbol{a}^\perp \notin \mathsf{span}(\mathbf{B})$ (such a vector exists by Property 1 in Definition 1), since in this case, $(\mathbf{B}|\boldsymbol{a}^\perp)$ is a basis of $\mathbb{Z}_p^{k+1}$. This allows to write $\{\widehat{\boldsymbol{c}}_j := \mathbf{B}(\boldsymbol{s}_j + y_j^{(\beta)}\boldsymbol{t}) + sy_j^{(\beta)}\boldsymbol{a}^\perp\}_{j\in[m]}$, which is identically distributed to $\{\widehat{\boldsymbol{c}}_j := \mathbf{B}\boldsymbol{s}_j + sy_j^{(\beta)}\boldsymbol{a}^\perp\}_{j\in[m]}$ as in G_3.

-Simulation of $\mathsf{KeyGenO}(\mathbf{F} \in \mathbb{Z}_p^{n\times m})$: As we argued previously for the case where $\mathcal{B}_3$ is given a real mddh challenge, the key computed by $\mathcal{B}_3$ is of the form: $K := [u]_1 \leftarrow_{\mathrm{R}} \mathbb{G}_1$, and $\widehat{K} := [\sum_{i,j} f_{i,j}\boldsymbol{c}_i^\top\widehat{\boldsymbol{c}}_j - v\sum_{i,j} f_{i,j}x_i^{(\beta)}y_j^{(\beta)}]_2 - [u]_2$, as in G_3.

□

Lemma 5 (G_4)**.** $\mathsf{Adv}_4 = 0$.

Proof of Lemma 5. In this game, the random bit $\beta \leftarrow_{\mathrm{R}} \{0,1\}$ sampled by SetupO only shows up in $\sum_{i,j} f_{i,j}x_i^{(\beta)}y_j^{(\beta)}$ in the secret keys. However, recall that the challenge messages $(\boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}), (\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})$ and the functions $\mathbf{F} \in \mathbb{Z}_p^{n\times m}$ queried to $\mathsf{KeyGenO}$ are such that

$$\sum_{i,j} f_{i,j}x_i^{(0)}y_j^{(0)} = \sum_{i,j} f_{i,j}x_i^{(1)}y_j^{(1)},$$

by definition of the security game. Therefore, the adversary's view in G_4 does not depend on β. □

Combining Lemma 1-5 gives Theorem 2. □

3.2 Public-key FE for Bilinear Maps

In this section, we propose a family of public-key functional encryption schemes for the bilinear map functionality, that is $F : \mathcal{K} \times \mathcal{M} \to \mathcal{Y}$, where $\mathcal{K} := \mathbb{Z}_p^{n\times m}$, $\mathcal{M} := \mathbb{Z}_p^n \times \mathbb{Z}_p^m$, and $\mathcal{Y} := \mathbb{G}_T$. The family of schemes is parametrized by an integer $k \geq 1$ and a matrix distribution $\mathcal{D}_k$ (see Definition 1) so that, for each $k \in \mathbb{N}$, and each matrix distribution $\mathcal{D}_k$, the scheme $\mathsf{FE}(k, \mathcal{D}_k)$, presented in Figure 8, is selectively secure under the $\mathcal{D}_k$-mddh and the 3-pddh assumptions, on asymmetric pairings.

Technical overview. We first give a high level view of our techniques. Our public-key FE builds on the private-key, single ciphertext secure FE presented in Section 3.1, but differs from it in the following essential way.

- In the public-key setting, for the encryption to compute $[\mathbf{A}\boldsymbol{r}_i + r\boldsymbol{b}^{\perp}x_i]$ and $[\mathbf{B}\boldsymbol{s}_j + s\boldsymbol{a}^{\perp}y_j]$ for $i \in [n], j \in [m]$ and any $\boldsymbol{x} \in \mathbb{Z}_p^n, \boldsymbol{y} \in \mathbb{Z}_p^m$, the vectors $[\boldsymbol{a}^{\perp}]_2$ and $[\boldsymbol{b}^{\perp}]_1$ would need to be part of the public key, which is incompatible with the mddh assumption on $[\mathbf{A}]_1$ or $[\mathbf{B}]_2$. To solve this problem, we add an extra dimension, namely, we use bases $\left(\begin{array}{c|c} \mathbf{A}|\boldsymbol{b}^{\perp} & 0 \\ \hline \mathbf{0} & 1 \end{array}\right)$ and $\left(\begin{array}{c|c} \mathbf{B}|\boldsymbol{a}^{\perp} & 0 \\ \hline \mathbf{0} & 1 \end{array}\right)$ where the extra dimension will be used for correctness, while $(\mathbf{A}|\boldsymbol{b}^{\perp})$ and $(\mathbf{B}|\boldsymbol{a}^{\perp})$ will be used for security (using the mddh assumption, since $\boldsymbol{a}^{\perp}$ and $\boldsymbol{b}^{\perp}$ are not part of the public key anymore).
- To avoid mix and match attacks, the encryption randomizes the bases

$$\left(\begin{array}{c|c} \mathbf{A}|\boldsymbol{b}^{\perp} & 0 \\ \hline \mathbf{0} & 1 \end{array}\right) \text{ and } \left(\begin{array}{c|c} \mathbf{B}|\boldsymbol{a}^{\perp} & 0 \\ \hline \mathbf{0} & 1 \end{array}\right)$$

  into

$$\mathbf{W}^{-1}\left(\begin{array}{c|c} \mathbf{A}|\boldsymbol{b}^{\perp} & 0 \\ \hline \mathbf{0} & 1 \end{array}\right) \text{ and } \mathbf{W}^{\top}\left(\begin{array}{c|c} \mathbf{B}|\boldsymbol{a}^{\perp} & 0 \\ \hline \mathbf{0} & 1 \end{array}\right)$$

  for $\mathbf{W} \leftarrow_{\mathrm{R}} \mathsf{GL}_{k+2}$ a random invertible matrix. This "glues" the components of a ciphertext that are in $\mathbb{G}_1$ to those that are in $\mathbb{G}_2$.
- We randomize the ciphertexts so as to contain $[\mathbf{A}\boldsymbol{r}_i \cdot \gamma]_1$ and $[\mathbf{B}\boldsymbol{s}_j \cdot \sigma]_2$, where $\gamma, \sigma \leftarrow_{\mathrm{R}} \mathbb{Z}_p$ are the same for all $i \in [n]$, and $j \in [m]$, but fresh for each ciphertext. The ciphertexts also contain $[\gamma \cdot \sigma]_1$, for correctness.

Discussion on the Techniques. We note that the techniques used here share some similarities with Dual Pairing Vector Space constructions (e.g., [33,34,30,17]). In particular, our produced ciphertexts and private keys are distributed as in their corresponding counterparts in [33]. The similarities end here though. These previous constructions all rely on the Dual System Encryption paradigm [40], where the security proof uses a hybrid argument over all secret keys, leaving the distribution of the public key untouched. Our approach, on the other hand, manages to avoid this inherent security loss by changing the distributions of *both* the secret and public keys. Our approach also differs from [12] and follow-up works [14,21] in that they focus on the comparison predicate (see Section 6), a function that can be expressed via a quadratic function that is significantly simpler than those considered here. Indeed, for the case of comparisons predicates it is enough to consider vectors of the form: $[\mathbf{A}\boldsymbol{r}_i + x_i\boldsymbol{b}^{\perp}]_1, [\mathbf{B}\boldsymbol{s}_j + y_j\boldsymbol{a}^{\perp}]_2$, where x_i and y_j are either 0, or some random value (fixed at setup time, and identical for all ciphertexts and secret keys), or are just random garbage.

In the following theorem we show that the scheme satisfies correctness.

Theorem 3 (Correctness). *For any $k \in \mathbb{N}^*$ and any matrix distribution $\mathcal{D}_k$, the functional encryption scheme $\mathsf{FE}(k, \mathcal{D}_k)$ defined in Figure 8 has perfect correctness.*

Proof of Theorem 3. Correctness follows from the facts that for all $i \in [n]$, $j \in [m]$:

$$e([\boldsymbol{c}_i]_1, [\widehat{\boldsymbol{c}}_j]_2) = [\gamma\boldsymbol{r}_i^{\top}\mathbf{A}^{\top}\mathbf{B}\boldsymbol{s}_j + x_iy_j]_T \text{ and } e([\boldsymbol{c}_{n+i}]_1, [\widehat{\boldsymbol{c}}_{m+j}]_2) = [\gamma\boldsymbol{r}_{n+i}^{\top}\mathbf{A}^{\top}\mathbf{B}\boldsymbol{s}_{m+j}]_T.$$

```
$\underline{\mathsf{Setup}(1^\lambda, F)}$:
$\mathsf{bgp} \leftarrow_{\mathrm{R}} \mathcal{G}(1^\lambda)$, $\mathbf{A}, \mathbf{B} \leftarrow_{\mathrm{R}} \mathcal{D}_k$;
For $i \in [2n], j \in [2m]$, $\boldsymbol{r}_i, \boldsymbol{s}_j \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$.
Return $\mathsf{mpk} := \{[\mathbf{A}\boldsymbol{r}_i]_1, [\mathbf{B}\boldsymbol{s}_j]_2\}_{i\in[2n],j\in[2m]}$
and $\mathsf{msk} := \Big(\mathbf{A}, \mathbf{B}, \{\boldsymbol{r}_i, \boldsymbol{s}_j\}_{i\in[2n],j\in[2m]}\Big)$

$\underline{\mathsf{KeyGen}(\mathsf{msk}, \mathbf{F} \in \mathbb{Z}_p^{n\times m})}$:
$K := [\sum_{i\in[n],j\in[m]} f_{i,j}\big(\boldsymbol{r}_i^\top \mathbf{A}^\top \mathbf{B}\boldsymbol{s}_j + \boldsymbol{r}_{i+n}^\top \mathbf{A}^\top \mathbf{B}\boldsymbol{s}_{j+m}\big)]_1 - [u]_1 \in \mathbb{G}_1$
$\widehat{K} := [u]_2 \in \mathbb{G}_2$, where $u \leftarrow_{\mathrm{R}} \mathbb{Z}_p$.
Return $\mathsf{sk}_{\mathbf{F}} := (K, \widehat{K}) \in \mathbb{G}_1 \times \mathbb{G}_2$

$\underline{\mathsf{Encrypt}(\mathsf{mpk}, (\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{Z}_p^n \times \mathbb{Z}_p^m)}$:
$\mathbf{W}, \mathbf{V} \leftarrow_{\mathrm{R}} \mathsf{GL}_{k+2}$, $\gamma \leftarrow_{\mathrm{R}} \mathbb{Z}_p$; $c_0 = \widehat{c}_0 := \gamma$; for all $i \in [n], j \in [m]$:
$\boldsymbol{c}_i := \begin{pmatrix} \gamma \cdot \mathbf{A}\boldsymbol{r}_i \\ x_i \end{pmatrix}^\top \mathbf{W}^{-1}$, $\boldsymbol{c}_{n+i} := \begin{pmatrix} \gamma \cdot \mathbf{A}\boldsymbol{r}_{n+i} \\ 0 \end{pmatrix}^\top \mathbf{V}^{-1}$,
$\widehat{\boldsymbol{c}}_j := \mathbf{W}\begin{pmatrix} \mathbf{B}\boldsymbol{s}_j \\ y_j \end{pmatrix}$, $\widehat{\boldsymbol{c}}_{m+j} := \mathbf{V}\begin{pmatrix} \mathbf{B}\boldsymbol{s}_{m+j} \\ 0 \end{pmatrix}$
$\mathsf{Ct}_{(\boldsymbol{x},\boldsymbol{y})} := \{[c_0]_1, [\widehat{c}_0]_2, [\boldsymbol{c}_i]_1, [\widehat{\boldsymbol{c}}_j]_2\}_{i\in[2n],j\in[2m]} \in \mathbb{G}_1^{2n(k+2)+1} \times \mathbb{G}_2^{2m(k+2)+1}$

$\underline{\mathsf{Decrypt}(\mathsf{mpk}, \mathsf{Ct}_{(\boldsymbol{x},\boldsymbol{y})}, \mathsf{sk}_{\mathbf{F}})}$:
Return $\sum_{i\in[n],j\in[m]} f_{i,j}\big(e([\boldsymbol{c}_i]_1, [\widehat{\boldsymbol{c}}_j]_2) + e([\boldsymbol{c}_{n+i}]_1, [\widehat{\boldsymbol{c}}_{m+j}]_2)\big) - e([c_0]_1, \widehat{K}) - e(K, [\widehat{c}_0]_2)$.
```

Fig. 8. $\mathsf{FE}(k, \mathcal{D}_k)$, a family of functional encryption schemes parametrized by $k \in \mathbb{N}^*$ and a matrix distribution $\mathcal{D}_k$, selectively secure under the $\mathcal{D}_k$-mddh and 3-pddh assumptions.

Therefore, the decryption gets

$$\begin{aligned}
&[\sum_{i\in[n],j\in[m]} f_{i,j}\gamma\big(\boldsymbol{r}_i^\top \mathbf{A}^\top \mathbf{B}\boldsymbol{s}_j + \boldsymbol{r}_{n+i}^\top \mathbf{A}^\top \mathbf{B}\boldsymbol{s}_{m+j}\big)]_T \\
&+ [\sum_{i\in[n],j\in[m]} f_{i,j} x_i y_j]_T - e([c_0]_1, \widehat{K}) - e(K, [\widehat{c}_0]_2) \\
&= [\sum_{i\in[n],j\in[m]} f_{i,j} x_i y_j]_T.
\end{aligned}$$

□

Next, in the following theorem we prove that the scheme satisfies indistinguishability based security in a selective sense.

Theorem 4 (Security). *For any $k \in \mathbb{N}^*$ and any matrix distribution $\mathcal{D}_k$, if the $\mathcal{D}_k$-mddh and the 3-pddh assumptions hold relative to $\mathcal{G}$, then the functional encryption scheme $\mathsf{FE}(k, \mathcal{D}_k)$ defined in Figure 8 is selectively secure. Precisely, for any PPT adversary $\mathcal{A}$, there exists PPT adversaries $\mathcal{B}$ and $\mathcal{B}'$ such that:*

$$\mathbf{Adv}_{\mathsf{FE},\mathcal{A}}^{\mathsf{sel\text{-}ind\text{-}fe\text{-}cpa}}(\lambda) \leq 24 \cdot \mathbf{Adv}_{\mathcal{G},\mathcal{B}}^{\mathcal{D}_k\text{-}\mathsf{mddh}}(\lambda) + 4 \cdot \mathbf{Adv}_{\mathcal{G},\mathcal{B}'}^{3\text{-}\mathsf{pddh}}(\lambda) + 2^{-\Omega(\lambda)}.$$

Proof of Theorem 4. We prove the security of $\mathsf{FE}(k, \mathcal{D}_k)$ via a series of games that are compactly presented in Figure 9. Before going to the details of the proof and proving the indistinguishability of each consecutive pair of games, we give below a more intuitive description of each game transition:

$\mathrm{G}_0, \boxed{\mathrm{G}_1, \dbox{$\mathrm{G}_2, \colorbox{lightgray}{G_3}$}, \colorbox{lightgray}{$\mathrm{G}_4$}}, \colorbox{lightgray}{G_5}:$

$\big((\boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}), (\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})\big) \leftarrow \mathcal{A}(1^\lambda)$

$\mathsf{bgp} \leftarrow_{\mathrm{R}} \mathcal{G}(1^\lambda)$; $\mathbf{A}, \mathbf{B} \leftarrow_{\mathrm{R}} \mathcal{D}_k$; $\beta \leftarrow_{\mathrm{R}} \{0,1\}$; $\boxed{\boldsymbol{a}^\perp \leftarrow_{\mathrm{R}} \mathsf{orth}(\mathbf{A}), \boldsymbol{b}^\perp \leftarrow_{\mathrm{R}} \mathsf{orth}(\mathbf{B}) \text{ s.t. } (\boldsymbol{b}^\perp)^\top \boldsymbol{a}^\perp = 1.}$

For $i \in [2n], j \in [2m]$: $\boldsymbol{r}_i \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$, $\boldsymbol{s}_j \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$, $\boxed{r, s \leftarrow_{\mathrm{R}} \mathbb{Z}_p}$

$$\mathsf{mpk} := \Big\{ \Big[\mathbf{A}\boldsymbol{r}_i + \boxed{r x_i^{(\beta)} \boldsymbol{b}^\perp}\Big]_1, \Big[\mathbf{A}\boldsymbol{r}_{n+i} - \boxed{r x_i^{(0)} \boldsymbol{b}^\perp}\Big]_1, \Big[\mathbf{B}\boldsymbol{s}_j + \boxed{s y_j^{(\beta)} \boldsymbol{a}^\perp}\Big]_2$$
$$\Big[\mathbf{B}\boldsymbol{s}_{m+j} + \boxed{s y_j^{(0)} \boldsymbol{a}^\perp}\Big]_2 \Big\}_{i \in [n], j \in [m]}$$

$\mathbf{W} \leftarrow_{\mathrm{R}} \mathsf{GL}_{k+2}$, $\gamma \leftarrow_{\mathrm{R}} \mathbb{Z}_p$; $\dbox{$v \leftarrow_{\mathrm{R}} \mathbb{Z}_p$}$; $c_0 = \widehat{c}_0 := \gamma$

$$\boldsymbol{c}_i := \begin{pmatrix} \gamma \mathbf{A}\boldsymbol{r}_i + \boxed{\gamma r x_i^{(\beta)} \boldsymbol{b}^\perp} + \dbox{$v x_i^{(\beta)} \boldsymbol{b}^\perp$} \\ x_i^{(\beta)} - \colorbox{lightgray}{$x_i^{(\beta)}$} \end{pmatrix}^\top \mathbf{W}^{-1} \ ; \ \boldsymbol{c}_{n+i} := \begin{pmatrix} \gamma \mathbf{A}\boldsymbol{r}_{n+i} - \boxed{\gamma r x_i^{(0)} \boldsymbol{b}^\perp} - \dbox{$v x_i^{(0)} \boldsymbol{b}^\perp$} \\ 0 + \colorbox{lightgray}{$x_i^{(0)}$} \end{pmatrix}^\top \mathbf{V}^{-1}$$

$$\widehat{\boldsymbol{c}}_j := \mathbf{W} \begin{pmatrix} \mathbf{B}\boldsymbol{s}_j + \boxed{s y_j^{(\beta)} \boldsymbol{a}^\perp} \\ y_j^{(\beta)} - \colorbox{lightgray}{$y_j^{(\beta)}$} \end{pmatrix}; \ \widehat{\boldsymbol{c}}_{m+j} := \mathbf{V} \begin{pmatrix} \mathbf{B}\boldsymbol{s}_{m+j} + \boxed{s y_j^{(0)} \boldsymbol{a}^\perp} \\ 0 + \colorbox{lightgray}{$y_j^{(0)}$} \end{pmatrix}$$

$\mathsf{Ct}^\star := \{[c_0]_1, [\widehat{c}_0]_2, [\boldsymbol{c}_i]_1, [\widehat{\boldsymbol{c}}_j]_2\}_{i \in [2n], j \in [2m]}$

$\beta' \leftarrow \mathcal{A}^{\mathsf{KeyGenO}(\cdot)}(\mathsf{mpk}, \mathsf{Ct}^\star)$

Return 1 if $\beta' = \beta$, 0 otherwise.

$\underline{\mathsf{KeyGenO}(\mathbf{F} \in \mathbb{Z}_p^{n \times m})\text{:}}$

$K := [\sum_{i \in [n], j \in [m]} f_{i,j} \big(\boldsymbol{r}_i^\top \mathbf{A}^\top \mathbf{B}\boldsymbol{s}_j + \boldsymbol{r}_{n+i}^\top \mathbf{A}^\top \mathbf{B}\boldsymbol{s}_{m+j}\big)]_1 - [u]_1 \in \mathbb{G}_1$

$\widehat{K} := [u]_2 \in \mathbb{G}_2$, where $u \leftarrow_{\mathrm{R}} \mathbb{Z}_p$.

Return $\mathsf{sk}_{\mathbf{F}} := (K, \widehat{K}) \in \mathbb{G}_1 \times \mathbb{G}_2$

Fig. 9. Games G_i, $i = 0, \ldots, 5$ for the proof of selective security of $\mathsf{FE}(k, \mathcal{D}_k)$ in Figure 8. In each procedure, the components inside a solid (dotted, gray) frame are only present in the games marked by a solid (dotted, gray) frame.

Game G_0 is the selective security experiment for scheme FE. For the sake of the proof, we look at the public key elements $\{[\mathbf{A}\boldsymbol{r}_i]_1, [\mathbf{B}\boldsymbol{s}_j]_2\}_{i\in[2n],j\in[2m]}$ as a ciphertext of the $\mathsf{FE}_{\mathsf{one}}$ scheme encrypting vectors $(\mathbf{0},\mathbf{0}) \in \mathbb{Z}_p^{2n} \times \mathbb{Z}_p^{2m}$.

Game G_1**:** with the above observation in mind, in this game we change the distribution of the public key elements so as to be interpreted as an $\mathsf{FE}_{\mathsf{one}}$ ciphertext encrypting the vectors

$$(\widetilde{\boldsymbol{x}}, \widetilde{\boldsymbol{y}}) = \left(\left(\frac{\boldsymbol{x}^{(\beta)}}{-\boldsymbol{x}^{(0)}}\right), \left(\frac{\boldsymbol{y}^{(\beta)}}{\boldsymbol{y}^{(0)}}\right)\right) \in \mathbb{Z}_p^{2n} \times \mathbb{Z}_p^{2m}$$

In Lemma 6 we show how to argue the indistinguishability of G_1 from G_0 based on the selective, single-ciphertext security of $\mathsf{FE}_{\mathsf{one}}$ (that in turn reduces to $\mathcal{D}_k$-mddh).

Game G_2**:** in this game we change the distribution of the $\boldsymbol{c}_i$ components of the challenge ciphertext. We switch from using $\{\gamma\mathbf{A}\boldsymbol{r}_i + \widetilde{x}_i \cdot \gamma r \boldsymbol{b}^\perp\}_{i\in[2n]}$ to $\{\gamma\mathbf{A}\boldsymbol{r}_i + \widetilde{x}_i \cdot (\gamma r + v)\boldsymbol{b}^\perp\}_{i\in[2n]}$, for a random $v \leftarrow_{\mathrm{R}} \mathbb{Z}_p$. In Lemma 7 we argue the indistinguishability of this change under the 3-pddh assumption.

Game G_3 : by using a statistical argument we show that in this game the challenge ciphertexts can be rewritten as

$$\boldsymbol{c}_i := \begin{pmatrix} \gamma\mathbf{A}\boldsymbol{r}_i + (\gamma r + v)x_i^{(\beta)}\boldsymbol{b}^\perp \\ 0 \end{pmatrix}^\top \mathbf{W}^{-1};\ \boldsymbol{c}_{n+i} := \begin{pmatrix} \gamma\mathbf{A}\boldsymbol{r}_{n+i} - (\gamma r + v)x_i^{(0)}\boldsymbol{b}^\perp x_i^{(0)} \end{pmatrix}^\top \mathbf{V}^{-1};$$

$$\widehat{\boldsymbol{c}}_j := \mathbf{W}\begin{pmatrix} \mathbf{B}\boldsymbol{s}_j + s y_j^{(\beta)}\boldsymbol{a}^\perp \\ 0 \end{pmatrix}; \widehat{\boldsymbol{c}}_{m+j} := \mathbf{V}\begin{pmatrix} \mathbf{B}\boldsymbol{s}_{m+j} + s y_j^{(0)}\boldsymbol{a}^\perp \\ y_j^{(0)} \end{pmatrix}.$$

This step essentially shows that the change in game G_2 made the ciphertexts less dependent on the bit β.

Game G_4**:** in this game we change again the distribution of the challenge ciphertext components $\boldsymbol{c}_i$ switching from using $\{\gamma\mathbf{A}\boldsymbol{r}_i + \widetilde{x}_i \cdot (\gamma r + v)\boldsymbol{b}^\perp\}_{i\in[2n]}$ to $\{\gamma\mathbf{A}\boldsymbol{r}_i + \widetilde{x}_i \cdot \gamma r \boldsymbol{b}^\perp\}_{i\in[2n]}$. This change is analogous to that introduced in game G_2, and its indistinguishability follows from the 3-pddh assumption.

The crucial observation is that the public key in this game can be seen as an $\mathsf{FE}_{\mathsf{one}}$ ciphertext encrypting vector $(\widetilde{\boldsymbol{x}}, \widetilde{\boldsymbol{y}})$, while the challenge ciphertext of game G_4 can be seen as an encryption of vectors

$$\left(\left(\frac{\mathbf{0}}{\boldsymbol{x}^{(0)}}\right), \left(\frac{\mathbf{0}}{\boldsymbol{y}^{(0)}}\right)\right) \in \mathbb{Z}_p^{2n} \times \mathbb{Z}_p^{2m}$$

using such public key. At a high level, the idea is that we moved to a game in which the dependence on the challenge messages $(\boldsymbol{x}^{(\beta)}, \boldsymbol{y}^{(\beta)})$ is only in the public key.

Game G_5**:** in this game we change back the distribution of the public key elements so as to be interpreted as an $\mathsf{FE}_{\mathsf{one}}$ ciphertext encrypting vectors $(\mathbf{0}, \mathbf{0})$. The indistinguishability of this game from game G_4 can be argued based on the selective, single-ciphertext security of the $\mathsf{FE}_{\mathsf{one}}$ scheme.

The proof is concluded by arguing that in this game the view of the adversary is independent of the bit β.

In what follows we use Adv_i to denote the advantage of $\mathcal{A}$ in game G_i, that is $\mathsf{Adv}_i := |1 - 2\Pr[\mathrm{G}_i \text{ returns } 1]|$. G_0 is defined as:

$$\begin{aligned} & \beta \leftarrow_{\mathrm{R}} \{0,1\} \\ \mathrm{G}_0 : \ & \beta' \leftarrow \mathbf{Exp}_{\mathsf{FE},\mathcal{A}}^{\mathsf{sel\text{-}ind\text{-}fe\text{-}cpa\text{-}}\beta}(\lambda) \\ & \text{Return 1 if } \beta' = \beta, \text{ 0 otherwise.} \end{aligned}$$

Where $\mathbf{Exp}^{\mathsf{sel\text{-}ind\text{-}fe\text{-}cpa\text{-}}\beta}_{\mathsf{FE},\mathcal{A}}(\lambda)$ is the experiment used in Definition 6 of indistinguishability-based security for functional encryption. In particular, we have

$$\mathsf{Adv}_0 = \mathbf{Adv}^{\mathsf{sel\text{-}ind\text{-}fe\text{-}cpa}}_{\mathsf{FE},\mathcal{A}}(\lambda).$$

Lemma 6 **(**G_0 **to** G_1**).** *There exists a PPT adversary* $\mathcal{B}_0$*:*

$$|\mathsf{Adv}_0 - \mathsf{Adv}_1| \leq 12 \cdot \mathbf{Adv}^{\mathcal{D}_k\text{-}\mathsf{mddh}}_{\mathcal{G},\mathcal{B}_0}(\lambda) + \frac{4}{p}.$$

Proof of Lemma 6. Using the selective, single-ciphertext security of the underlying private-key scheme (which is exactly the scheme in Figure 3), we can change the distribution of the public key elements from $\{[\mathbf{A}\boldsymbol{r}_i]_1, [\mathbf{B}\boldsymbol{s}_j]_2\}_{i\in[2n],j\in[2m]}$ to

$$\left\{[\mathbf{A}\boldsymbol{r}_i + rx_i^{(\beta)}\boldsymbol{b}^\perp]_1, [\mathbf{A}\boldsymbol{r}_{n+i} - rx_i^{(0)}\boldsymbol{b}^\perp]_1, [\mathbf{B}\boldsymbol{s}_j + sy_j^{(\beta)}\boldsymbol{a}^\perp]_2, [\mathbf{B}\boldsymbol{s}_{m+j} + sy_j^{(0)}\boldsymbol{a}^\perp]_2\right\}_{i\in[n],j\in[m]}.$$

In order to apply Theorem 2 we rely on the fact that the FE public key can be seen as an $\mathsf{FE}_{\mathsf{one}}$ encryption of longer vectors

$$\widetilde{\boldsymbol{x}}^{(0)} = \mathbf{0} \in \mathbb{Z}_p^{2n} \text{ and } \widetilde{\boldsymbol{y}}^{(0)} = \mathbf{0} \in \mathbb{Z}_p^{2m} \text{ in } G_0,$$

$$\widetilde{\boldsymbol{x}}^{(1)} = (\boldsymbol{x}^{(\beta)} || -\boldsymbol{x}^{(0)}) \in \mathbb{Z}_p^{2n} \text{ and } \widetilde{\boldsymbol{y}}^{(1)} = (\boldsymbol{y}^{(\beta)} || \boldsymbol{y}^{(0)}) \in \mathbb{Z}_p^{2m} \text{ in } G_1.$$

Also, secret keys in FE can be seen as $\mathsf{FE}_{\mathsf{one}}$ secret keys corresponding to matrices

$$\widetilde{\mathbf{F}} = \left(\begin{array}{c|c} \mathbf{F} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{F} \end{array}\right) \in \mathbb{Z}_p^{2n\times 2m}$$

With this observation in mind, it can be seen that the restriction

$$\boldsymbol{x}^{(1)\top}\,\mathbf{F}\,\boldsymbol{y}^{(1)} = \boldsymbol{x}^{(0)\top}\,\mathbf{F}\,\boldsymbol{y}^{(0)}$$

in the queries made by $\mathcal{A}$ translates into legitimate queries by $\mathcal{B}_0$ since $\boldsymbol{x}^{(\beta)\top}\,\mathbf{F}\,\boldsymbol{y}^{(\beta)} - \boldsymbol{x}^{(0)\top}\,\mathbf{F}\,\boldsymbol{y}^{(0)} = 0$ and $\widetilde{\boldsymbol{x}}^{(0)\top}\,\widetilde{\mathbf{F}}\,\widetilde{\boldsymbol{y}}^{(0)} = \widetilde{\boldsymbol{x}}^{(1)\top}\,\widetilde{\mathbf{F}}\,\widetilde{\boldsymbol{y}}^{(1)} = 0$. Thus, by Theorem 2 (security of the single-ciphertext secure scheme), we obtain the lemma. □

Lemma 7 **(**G_1 **to** G_2**).** *There exists a PPT adversary* $\mathcal{B}_1$ *such that:*

$$|\mathsf{Adv}_1 - \mathsf{Adv}_2| \leq 2 \cdot \mathbf{Adv}^{3-\mathsf{pddh}}_{\mathcal{G},\mathcal{B}_1}(\lambda) + \frac{2}{p}.$$

Here, we change the distribution of the challenge ciphertexts, using the 3-pddh assumption.

Proof of Lemma 7. Upon receiving a 3-pddh challenge $(\mathsf{bgp}, [a]_1, [b]_2, [c]_1, [c]_2, [z]_1)$ (see Definition 3), and the challenge messages $(\boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}), (\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})$, $\mathcal{B}_1$ simulates the output of the Setup phase by picking $\mathbf{A}, \mathbf{B} \leftarrow_{\mathrm{R}} \mathcal{D}_k$; $\beta \leftarrow_{\mathrm{R}} \{0,1\}$; $\boldsymbol{a}^\perp \leftarrow_{\mathrm{R}} \mathsf{orth}(\mathbf{A}), \boldsymbol{b}^\perp \leftarrow_{\mathrm{R}} \mathsf{orth}(\mathbf{B})$ s.t. $(\boldsymbol{b}^\perp)^\top\boldsymbol{a}^\perp = 1$, and setting:

$$[r]_1 := [a]_1, [s]_2 := [b]_2, [\gamma]_1 := [c]_1 \text{ and } [\gamma]_2 := [c]_2.$$

Then, for $i \in [2n], j \in [2m]$, $\mathcal{B}_1$ picks $\boldsymbol{r}_i \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$, $\boldsymbol{s}_j \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$ and computes

$$\mathsf{mpk} := \left\{\left[\mathbf{A}\boldsymbol{r}_i + rx_i^{(\beta)}\boldsymbol{b}^\perp\right]_1, \left[\mathbf{A}\boldsymbol{r}_{n+i} - rx_i^{(0)}\boldsymbol{b}^\perp\right]_1, \left[\mathbf{B}\boldsymbol{s}_j + sy_j^{(\beta)}\boldsymbol{a}^\perp\right]_2, \left[\mathbf{B}\boldsymbol{s}_{m+j} + sy_j^{(0)}\boldsymbol{a}^\perp\right]_2\right\}_{i\in[n],j\in[m]}.$$

It picks $\widetilde{\mathbf{W}}, \widetilde{\mathbf{V}} \leftarrow_{\mathrm{R}} \mathsf{GL}_{k+2}$ and implicitly sets

$$\mathbf{W} := \widetilde{\mathbf{W}} \left(\begin{array}{c|c} \mathbf{B}|s \cdot \boldsymbol{a}^{\perp} & 0 \\ \hline \mathbf{0} & 1 \end{array}\right)^{-1} \text{ and } \mathbf{V} := \widetilde{\mathbf{V}} \left(\begin{array}{c|c} \mathbf{B}|s \cdot \boldsymbol{a}^{\perp} & 0 \\ \hline \mathbf{0} & 1 \end{array}\right)^{-1}.$$

Here we use the fact that $(\mathbf{B}|\boldsymbol{a}^{\perp})$ is full rank since $\boldsymbol{a}^{\perp} \notin \mathsf{span}(\mathbf{B})$ (this is implied by the fact that $(\boldsymbol{b}^{\perp})^{\top}\boldsymbol{a}^{\perp} \neq 0$, with $\boldsymbol{b}^{\perp} \in \mathsf{orth}(\mathbf{B})$), and that with probability $1 - \frac{1}{p}$ over the choices of $s \leftarrow_{\mathrm{R}} \mathbb{Z}_p$, $s \neq 0$.

Then, for $i \in [n], j \in [m]$, it computes

$$[\boldsymbol{c}_i]_1 := \left[\begin{pmatrix} \gamma \boldsymbol{r}_i \\ z \cdot x_i^{(\beta)} \\ x_i^{(\beta)} \end{pmatrix}^{\top} \left(\begin{array}{c|c|c} \mathbf{A}^{\top}\mathbf{B} & 0 & 0 \\ \hline 0 & \underbrace{(\boldsymbol{b}^{\perp})^{\top}\boldsymbol{a}^{\perp}}_{=1} & 0 \\ \hline 0 & 0 & 1 \end{array}\right) \widetilde{\mathbf{W}}^{-1} \right]_1 \text{ and } [\widehat{\boldsymbol{c}}_j]_2 := \left[\widetilde{\mathbf{W}} \begin{pmatrix} \boldsymbol{s}_j \\ y_j^{(\beta)} \\ y_j^{(\beta)} \end{pmatrix} \right]_2$$

$$[\boldsymbol{c}_{n+i}]_1 := \left[\begin{pmatrix} \gamma \boldsymbol{r}_{n+i} \\ -z \cdot x_i^{(0)} \\ 0 \end{pmatrix}^{\top} \left(\begin{array}{c|c|c} \mathbf{A}^{\top}\mathbf{B} & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 1 \end{array}\right) \widetilde{\mathbf{V}}^{-1} \right]_1 \text{ and } [\widehat{\boldsymbol{c}}_{m+j}]_2 := \left[\widetilde{\mathbf{V}} \begin{pmatrix} \boldsymbol{s}_{m+j} \\ y_j^{(0)} \\ 0 \end{pmatrix} \right]_2 .$$

$\mathcal{B}_1$ computes $[c_0]_1 := [\gamma]_1$, $[\widehat{c}_0]_2 := [\gamma]_2$, and simulates $\mathsf{KeyGenO}$ as in G_2 (see Figure 9). Finally, if $\mathcal{A}$ outputs β', $\mathcal{B}_1$ outputs 1 if $\beta' = \beta$, and 0 otherwise.

It can be seen that when $[z]_1$ is a real 3-pddh challenge, i.e., $[z]_1 = [abc]_1$, then $\mathcal{B}_1$ simulates G_1; whereas it simulates G_2 when $[z]_1 \leftarrow_{\mathrm{R}} \mathbb{G}_1$. In particular, while this is easy to see for the elements of the public key and for ciphertexts $[\widehat{\boldsymbol{c}}_j]_2$, $[\widehat{\boldsymbol{c}}_{m+j}]_2$, for the ciphertext elements $[\boldsymbol{c}_i]_1$, $[\boldsymbol{c}_{n+i}]_1$ we observe that they can be written as

$$\boldsymbol{c}_i := \begin{pmatrix} \gamma \mathbf{B}^{\top}\mathbf{A}\boldsymbol{r}_i \\ z \cdot x_i^{(\beta)} \\ x_i^{(\beta)} \end{pmatrix}^{\top} \left(\begin{array}{c|c} \mathbf{B}|s \cdot \boldsymbol{a}^{\perp} & 0 \\ \hline \mathbf{0} & 1 \end{array}\right)^{-1} \mathbf{W}^{-1} = \begin{pmatrix} \gamma \mathbf{A}\boldsymbol{r}_i + zs^{-1} \cdot x_i^{(\beta)} \\ x_i^{(\beta)} \end{pmatrix}^{\top} \mathbf{W}^{-1}$$

$$\boldsymbol{c}_{n+i} := \begin{pmatrix} \gamma \mathbf{B}^{\top}\mathbf{A}\boldsymbol{r}_{n+i} \\ -z \cdot x_i^{(0)} \\ 0 \end{pmatrix}^{\top} \left(\begin{array}{c|c} \mathbf{B}|s \cdot \boldsymbol{a}^{\perp} & 0 \\ \hline \mathbf{0} & 1 \end{array}\right)^{-1} \mathbf{V}^{-1} = \begin{pmatrix} \gamma \mathbf{A}\boldsymbol{r}_{n+i} + zs^{-1} \cdot x_i^{(0)} \\ 0 \end{pmatrix}^{\top} \mathbf{V}^{-1}.$$

So, if $z = abc$, then $zs^{-1} = r\gamma$ and the ciphertexts are distributed as in G_1; otherwise if z is random zs^{-1} is identically distributed to $(r\gamma + v)$ as in G_2. This proves $|\mathsf{Adv}_1 - \mathsf{Adv}_2| \leq 2 \cdot \mathbf{Adv}_{\mathcal{G},\mathcal{B}_1}^{3-\mathsf{pddh}}(\lambda) + \frac{2}{p}$. □

Lemma 8 **(G_2 to G_3).** $|\mathsf{Adv}_2 - \mathsf{Adv}_3| \leq \frac{12}{p}$.

Here, we change the distribution of the challenge ciphertexts, using a statistical argument.

Proof of Lemma 8. First, we use the fact that for all $r, \gamma \in \mathbb{Z}_p$, the following are identically distributed: (r, γ, v) and $(r, \gamma, v + \gamma r)$, where $v \leftarrow_{\mathrm{R}} \mathbb{Z}_p$. Therefore, we can write the challenge ciphertexts as follows. For all i$\in [n], j \in [m]$: $\boldsymbol{c}_i := \begin{pmatrix} \gamma \mathbf{A}\boldsymbol{r}_i + v x_i^{(\beta)} \boldsymbol{b}^{\perp} \\ x_i^{(\beta)} \end{pmatrix}^{\top} \mathbf{W}^{-1}$; $\boldsymbol{c}_{n+i} := \begin{pmatrix} \gamma \mathbf{A}\boldsymbol{r}_{n+i} - v x_i^{(0)} \boldsymbol{b}^{\perp} \\ 0 \end{pmatrix}^{\top} \mathbf{V}^{-1}$.

Then, we use the facts that:

- (s,v) where $s, v \leftarrow_{\mathrm{R}} \mathbb{Z}_p$ and (s,v) where $s \leftarrow_{\mathrm{R}} \mathbb{Z}_p^*$ and $v \leftarrow_{\mathrm{R}} \mathbb{Z}_p$ such that $sv+1 \neq 0$ are statistically $\frac{2}{p}$-close distributions.
- $\mathbf{W} \leftarrow_{\mathrm{R}} \mathsf{GL}_{k+2}$ and $\widetilde{\mathbf{W}} \cdot \left(\begin{array}{c|c} \mathbf{B}|\boldsymbol{a}^\perp & 0 \\ \hline \mathbf{0} & 1 \end{array}\right) \cdot \left(\begin{array}{c|c|c} \mathsf{Id}_{k\times k} & 0 & 0 \\ \hline 0 & 1-\frac{1}{sv+1} & \frac{s}{sv+1} \\ \hline 0 & \frac{1}{sv+1} & \frac{-s}{sv+1} \end{array}\right) \cdot \left(\begin{array}{c|c} \mathbf{B}|\boldsymbol{a}^\perp & 0 \\ \hline \mathbf{0} & 1 \end{array}\right)^{-1}$, where $\widetilde{\mathbf{W}} \leftarrow_{\mathrm{R}} \mathsf{GL}_{k+2}$, $s \leftarrow_{\mathrm{R}} \mathbb{Z}_p^*$,and $v \leftarrow_{\mathrm{R}} \mathbb{Z}_p$ such that $sv+1 \neq 0$, are identically distributed, since $(\mathbf{B}|\boldsymbol{a}^\perp)$ is full rank (this is implied by the fact that $\boldsymbol{a}^\perp \notin \mathsf{span}(\mathbf{B})$, since $(\boldsymbol{b}^\perp)^\top \boldsymbol{a}^\perp \neq 0$), and $\det \begin{pmatrix} 1-\frac{1}{sv+1} & \frac{s}{sv+1} \\ \frac{1}{sv+1} & \frac{-s}{sv+1} \end{pmatrix} = \frac{1}{sv+1} \neq 0$.

Therefore, we can change the distribution of $\{\boldsymbol{c}_i, \widehat{\boldsymbol{c}}_j\}_{i\in[n], j\in[m]}$ as follows:

$$\begin{aligned}
\widehat{\boldsymbol{c}}_j &= \widetilde{\mathbf{W}} \cdot \left(\begin{array}{c|c} \mathbf{B}|\boldsymbol{a}^\perp & 0 \\ \hline \mathbf{0} & 1 \end{array}\right) \cdot \left(\begin{array}{c|c|c} \mathsf{Id}_{k\times k} & 0 & 0 \\ \hline 0 & 1-\frac{1}{sv+1} & \frac{s}{sv+1} \\ \hline 0 & \frac{1}{sv+1} & \frac{-s}{sv+1} \end{array}\right) \begin{pmatrix} \boldsymbol{s}_j \\ s y_j^{(\beta)} \\ y_j^{(\beta)} \end{pmatrix} \\
&= \widetilde{\mathbf{W}} \cdot \left(\begin{array}{c|c} \mathbf{B}|\boldsymbol{a}^\perp & 0 \\ \hline \mathbf{0} & 1 \end{array}\right) \cdot \begin{pmatrix} \boldsymbol{s}_j \\ s y_j^{(\beta)} \\ 0 \end{pmatrix} \\
&= \widetilde{\mathbf{W}} \cdot \begin{pmatrix} \mathbf{B}\boldsymbol{s}_j + s y_j^{(\beta)} \boldsymbol{a}^\perp \\ 0 \end{pmatrix}
\end{aligned}$$

and

$$\begin{aligned}
\boldsymbol{c}_i &= \begin{pmatrix} \gamma \boldsymbol{r}_i \\ v x_i^{(\beta)} \\ x_i^{(\beta)} \end{pmatrix}^\top \left(\begin{array}{c|c|c} \mathbf{A}^\top\mathbf{B} & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 1 \end{array}\right) \cdot \left(\begin{array}{c|c|c} \mathsf{Id}_{k\times k} & 0 & 0 \\ \hline 0 & 1-\frac{1}{sv+1} & \frac{s}{sv+1} \\ \hline 0 & \frac{1}{sv+1} & \frac{-s}{sv+1} \end{array}\right)^{-1} \cdot \left(\begin{array}{c|c} \mathbf{B}|\boldsymbol{a}^\perp & 0 \\ \hline \mathbf{0} & 1 \end{array}\right)^{-1} \cdot \widetilde{\mathbf{W}}^{-1} \\
&= \begin{pmatrix} \gamma \boldsymbol{r}_i \\ v \cdot x_i^{(\beta)} \\ x_i^{(\beta)} \end{pmatrix}^\top \left(\begin{array}{c|c|c} \mathbf{A}^\top\mathbf{B} & 0 & 0 \\ \hline 0 & 1 & 1 \\ \hline 0 & \frac{1}{s} & -v \end{array}\right) \cdot \left(\begin{array}{c|c} \mathbf{B}|\boldsymbol{a}^\perp & 0 \\ \hline \mathbf{0} & 1 \end{array}\right)^{-1} \cdot \widetilde{\mathbf{W}}^{-1} \\
&= \begin{pmatrix} \gamma \boldsymbol{r}_i \\ (v+\frac{1}{s}) \cdot x_i^{(\beta)} \\ 0 \end{pmatrix}^\top \cdot \left(\begin{array}{c|c|c} \mathbf{A}^\top\mathbf{B} & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 1 \end{array}\right) \cdot \left(\begin{array}{c|c} \mathbf{B}|\boldsymbol{a}^\perp & 0 \\ \hline \mathbf{0} & 1 \end{array}\right)^{-1} \cdot \widetilde{\mathbf{W}}^{-1} \\
&= \begin{pmatrix} \gamma \mathbf{A} \boldsymbol{r}_i + (v+\frac{1}{s}) \cdot x_i^{(\beta)} \boldsymbol{b}^\perp \\ 0 \end{pmatrix}^\top \cdot \widetilde{\mathbf{W}}^{-1}
\end{aligned}$$

Then, we use the facts that:

- (s,v) where $s \leftarrow_{\mathrm{R}} \mathbb{Z}_p^*$ and $v \leftarrow_{\mathrm{R}} \mathbb{Z}_p$ such that $sv+1 \neq 0$ is statistically $\frac{1}{p}$-close to(s,v) where $s \leftarrow_{\mathrm{R}} \mathbb{Z}_p^*$ and $v \leftarrow_{\mathrm{R}} \mathbb{Z}_p^*$ such that $sv+1 \neq 0$.
- $\mathbf{V} \leftarrow_{\mathrm{R}} \mathsf{GL}_{k+2}$ and $\widetilde{\mathbf{V}} \cdot \left(\begin{array}{c|c} \mathbf{B}|\boldsymbol{a}^\perp & 0 \\ \hline \mathbf{0} & 1 \end{array}\right) \cdot \left(\begin{array}{c|c|c} \mathsf{Id}_{k\times k} & 0 & 0 \\ \hline 0 & 1 & \frac{1}{v} \\ \hline 0 & \frac{1}{s} & 1+\frac{1}{sv} \end{array}\right) \cdot \left(\begin{array}{c|c} \mathbf{B}|\boldsymbol{a}^\perp & 0 \\ \hline \mathbf{0} & 1 \end{array}\right)^{-1}$, where $\widetilde{\mathbf{V}} \leftarrow_{\mathrm{R}} \mathsf{GL}_{k+2}$, $s \leftarrow_{\mathrm{R}} \mathbb{Z}_p^*$,and $v \leftarrow_{\mathrm{R}} \mathbb{Z}_p^*$ such that $sv+1 \neq 0$, are identically distributed, since $(\mathbf{B}|\boldsymbol{a}^\perp)$ is full rank, and $\det \begin{pmatrix} 1 & \frac{1}{v} \\ \frac{1}{s} & 1+\frac{1}{sv} \end{pmatrix} = 1$.

Therefore, we can change the distribution of $\{\boldsymbol{c}_{n+i}, \widehat{\boldsymbol{c}}_{m+j}\}_{i\in[n],j\in[m]}$ as follows:

$$\begin{aligned}
\widehat{\boldsymbol{c}}_{m+j} &= \widetilde{\mathbf{V}} \cdot \left(\begin{array}{c|c}\mathbf{B}|\boldsymbol{a}^{\perp} & 0 \\ \hline \mathbf{0} & 1\end{array}\right) \cdot \left(\begin{array}{c|c|c}\mathsf{Id}_{k\times k} & 0 & 0 \\ \hline 0 & 1 & \frac{1}{v} \\ \hline 0 & \frac{1}{s} & 1+\frac{1}{sv}\end{array}\right) \begin{pmatrix} \boldsymbol{s}_j \\ sy_j^{(0)} \\ 0 \end{pmatrix} \\
&= \widetilde{\mathbf{V}} \cdot \left(\begin{array}{c|c}\mathbf{B}|\boldsymbol{a}^{\perp} & 0 \\ \hline \mathbf{0} & 1\end{array}\right) \cdot \begin{pmatrix} \boldsymbol{s}_j \\ sy_j^{(0)} \\ y_j^{(0)} \end{pmatrix} \\
&= \widetilde{\mathbf{V}} \cdot \begin{pmatrix} \mathbf{B}\boldsymbol{s}_j + sy_j^{(0)}\boldsymbol{a}^{\perp} \\ y_j^{(0)} \end{pmatrix}
\end{aligned}$$

and

$$\begin{aligned}
\boldsymbol{c}_{n+i} &= \begin{pmatrix} \gamma\boldsymbol{r}_{n+i} \\ -vx_i^{(0)} \\ 0 \end{pmatrix}^{\top} \left(\begin{array}{c|c|c}\mathbf{A}^{\top}\mathbf{B} & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 1\end{array}\right) \cdot \left(\begin{array}{c|c|c}\mathsf{Id}_{k\times k} & 0 & 0 \\ \hline 0 & 1 & \frac{1}{v} \\ \hline 0 & \frac{1}{s} & 1+\frac{1}{sv}\end{array}\right)^{-1} \cdot \left(\begin{array}{c|c}\mathbf{B}|\boldsymbol{a}^{\perp} & 0 \\ \hline \mathbf{0} & 1\end{array}\right)^{-1} \cdot \widetilde{\mathbf{V}}^{-1} \\
&= \begin{pmatrix} \gamma\boldsymbol{r}_{n+i} \\ -(v+\frac{1}{s})x_i^{(0)} \\ x_i^{(0)} \end{pmatrix}^{\top} \left(\begin{array}{c|c|c}\mathbf{A}^{\top}\mathbf{B} & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 1\end{array}\right) \cdot \left(\begin{array}{c|c}\mathbf{B}|\boldsymbol{a}^{\perp} & 0 \\ \hline \mathbf{0} & 1\end{array}\right)^{-1} \cdot \widetilde{\mathbf{V}}^{-1} \\
&= \begin{pmatrix} \gamma\mathbf{A}\boldsymbol{r}_{n+i} - (v+\frac{1}{s})x_i^{(0)}\boldsymbol{b}^{\perp} \\ x_i^{(0)} \end{pmatrix}^{\top} \cdot \widetilde{\mathbf{V}}^{-1}
\end{aligned}$$

Finally, we use the fact that for all $\gamma \in \mathbb{Z}_p$, the distributions: $(v+\frac{1}{s}, s)$ where $s, v \leftarrow_{\mathrm{R}} \mathbb{Z}_p^*$ such that $sv+1 \neq 0$, and $(v+\gamma, s)$ where $s, v \leftarrow_{\mathrm{R}} \mathbb{Z}_p$, are statistically $\frac{3}{p}$-close. Thus, we obtain, for all $i \in [n]$ and $j \in [m]$: $\boldsymbol{c}_i := \begin{pmatrix} \gamma\mathbf{A}\boldsymbol{r}_i + (v+\gamma)x_i^{(\beta)}\boldsymbol{b}^{\perp} \\ 0 \end{pmatrix}^{\top} \widetilde{\mathbf{W}}^{-1}$, $\boldsymbol{c}_{n+i} := \begin{pmatrix} \gamma\mathbf{A}\boldsymbol{r}_{n+i} - (v+\gamma)x_i^{(0)}\boldsymbol{b}^{\perp} \\ x_i^{(0)} \end{pmatrix}^{\top} \widetilde{\mathbf{V}}^{-1}$, $\widehat{\boldsymbol{c}}_j := \widetilde{\mathbf{W}} \begin{pmatrix} \gamma\mathbf{B}\boldsymbol{s}_j + y_j^{(\beta)}\boldsymbol{a}^{\perp} \\ 0 \end{pmatrix}$, $\widehat{\boldsymbol{c}}_{m+j} := \widetilde{\mathbf{V}} \begin{pmatrix} \gamma\mathbf{B}\boldsymbol{s}_j + y_j^{(0)}\boldsymbol{a}^{\perp} \\ y_j^{(0)} \end{pmatrix}$, as in G$_3$.

This proves $|\mathsf{Adv}_2 - \mathsf{Adv}_3| \leq \frac{12}{p}$. □

Lemma 9 (G$_3$ to G$_4$). *There exists an adversary $\mathcal{B}_3$ such that*

$$|\mathsf{Adv}_3 - \mathsf{Adv}_4| \leq 2 \cdot \mathbf{Adv}_{\mathcal{G},\mathcal{B}_2}^{3\text{-}\mathsf{pddh}}(\lambda) + \frac{2}{p}.$$

Here, we change the distribution of the challenge ciphertext, using the 3-pddh assumption, as for Lemma 7.

Proof of Lemma 9. Upon receiving a 3-pddh challenge $(\mathsf{bgp}, [a]_1, [b]_2, [c]_1, [c]_2, [z]_1)$ (see Definition 3), and the challenge messages $(\boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}), (\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})$, $\mathcal{B}_1$ simulates the output of the Setup phase by picking $\mathbf{A}, \mathbf{B} \leftarrow_{\mathrm{R}} \mathcal{D}_k$; $\beta \leftarrow_{\mathrm{R}} \{0,1\}$; $\boldsymbol{a}^{\perp} \leftarrow_{\mathrm{R}} \mathsf{orth}(\mathbf{A}), \boldsymbol{b}^{\perp} \leftarrow_{\mathrm{R}} \mathsf{orth}(\mathbf{B})$ s.t. $(\boldsymbol{b}^{\perp})^{\top}\boldsymbol{a}^{\perp} = 1$, and setting:

$$[r]_1 := [a]_1, [s]_2 := [b]_2, [\gamma]_1 := [c]_1 \text{ and } [\gamma]_2 := [c]_2.$$

Then, for $i \in [2n], j \in [2m]$, $\mathcal{B}_1$ picks $\boldsymbol{r}_i \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$, $\boldsymbol{s}_j \leftarrow_{\mathrm{R}} \mathbb{Z}_p^k$ and computes

$$\mathsf{mpk} := \left\{ \left[\mathbf{A}\boldsymbol{r}_i + rx_i^{(\beta)}\boldsymbol{b}^\perp\right]_1, \left[\mathbf{A}\boldsymbol{r}_{n+i} - rx_i^{(0)}\boldsymbol{b}^\perp\right]_1, \left[\mathbf{B}\boldsymbol{s}_j + sy_j^{(\beta)}\boldsymbol{a}^\perp\right]_2, \left[\mathbf{B}\boldsymbol{s}_{m+j} + sy_j^{(0)}\boldsymbol{a}^\perp\right]_2 \right\}_{i\in[n],j\in[m]}.$$

It picks $\widetilde{\mathbf{W}}, \widetilde{\mathbf{V}} \leftarrow_{\mathrm{R}} \mathsf{GL}_{k+2}$ and implicitly sets

$$\mathbf{W} := \widetilde{\mathbf{W}} \left(\begin{array}{c|c} \mathbf{B}|s\cdot\boldsymbol{a}^\perp & 0 \\ \hline \mathbf{0} & 1 \end{array}\right)^{-1} \text{ and } \mathbf{V} := \widetilde{\mathbf{V}} \left(\begin{array}{c|c} \mathbf{B}|s\cdot\boldsymbol{a}^\perp & 0 \\ \hline \mathbf{0} & 1 \end{array}\right)^{-1}.$$

Here we use the fact that $(\mathbf{B}|\boldsymbol{a}^\perp)$ is full rank since $\boldsymbol{a}^\perp \notin \mathsf{span}(\mathbf{B})$ (this is implied by the fact that $(\boldsymbol{b}^\perp)^\top \boldsymbol{a}^\perp \neq 0$, with $\boldsymbol{b}^\perp \in \mathsf{orth}(\mathbf{B})$), and that with probability $1 - \frac{1}{p}$ over the choices of $s \leftarrow_{\mathrm{R}} \mathbb{Z}_p$, $s \neq 0$.

Then, for $i \in [n], j \in [m]$, it computes

$$[\boldsymbol{c}_i]_1 := \left[\begin{pmatrix} \gamma \boldsymbol{r}_i \\ z \cdot x_i^{(\beta)} \\ 0 \end{pmatrix}^\top \left(\begin{array}{c|c|c} \mathbf{A}^\top\mathbf{B} & 0 & 0 \\ \hline 0 & \underbrace{(\boldsymbol{b}^\perp)^\top \boldsymbol{a}^\perp}_{=1} & 0 \\ \hline 0 & 0 & 1 \end{array}\right) \widetilde{\mathbf{W}}^{-1} \right]_1 \text{ and } [\widehat{\boldsymbol{c}}_j]_2 := \left[\widetilde{\mathbf{W}} \begin{pmatrix} \boldsymbol{s}_j \\ y_j^{(\beta)} \\ 0 \end{pmatrix} \right]_2$$

$$[\boldsymbol{c}_{n+i}]_1 := \left[\begin{pmatrix} \gamma \boldsymbol{r}_{n+i} \\ -z \cdot x_i^{(0)} \\ x_i^{(0)} \end{pmatrix}^\top \left(\begin{array}{c|c|c} \mathbf{A}^\top\mathbf{B} & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 1 \end{array}\right) \widetilde{\mathbf{V}}^{-1} \right]_1 \text{ and } [\widehat{\boldsymbol{c}}_{m+j}]_2 := \left[\widetilde{\mathbf{V}} \begin{pmatrix} \boldsymbol{s}_{m+j} \\ y_j^{(0)} \\ y_j^{(0)} \end{pmatrix} \right]_2.$$

Finally, $\mathcal{B}_1$ computes $[c_0]_1 := [\gamma]_1$, $[\widehat{c}_0]_2 := [\gamma]_2$, and simulates $\mathsf{KeyGenO}$ as in G_3 (see Figure 9). Note that when $[z]_1$ is a real 3-pddh challenge, i.e $[z]_1 = [abc]_1$, then $\mathcal{B}_1$ simulates G_3; whereas it simulates G_4 when $[z]_1 \leftarrow_{\mathrm{R}} \mathbb{G}_1$. This proves $|\mathsf{Adv}_3 - \mathsf{Adv}_4| \leq 2 \cdot \mathbf{Adv}^{3-\mathsf{pddh}}_{\mathcal{G},\mathcal{B}_1}(\lambda) + \frac{2}{p}$. □

Lemma 10 **(G_4 to G_5).** *There exists an adversary $\mathcal{B}_4$ such that*

$$|\mathsf{Adv}_4 - \mathsf{Adv}_5| \leq 12 \cdot \mathbf{Adv}^{\mathcal{D}_k\text{-}\mathsf{mddh}}_{\mathcal{G},\mathcal{B}_4}(\lambda) + \frac{4}{p}.$$

Proof of Lemma 10. This transition is symmetric to that between G_0 and G_1: we use the selective, single-ciphertext security of the underlying private-key scheme (in Figure 3), to switch: $\{[\mathbf{A}\boldsymbol{r}_i + rx_i^{(\beta)}\boldsymbol{b}^\perp]_1, [\mathbf{A}\boldsymbol{r}_{n+i} - rx_i^{(0)}\boldsymbol{b}^\perp]_1, [\mathbf{B}\boldsymbol{s}_j + sy_j^{(\beta)}\boldsymbol{a}^\perp]_2, [\mathbf{B}\boldsymbol{s}_{m+j} + sy_j^{(0)}\boldsymbol{a}^\perp]_2\}_{i\in[n],j\in[m]}$ to $\{[\mathbf{A}\boldsymbol{r}_i]_1, [\mathbf{B}\boldsymbol{s}_j]_2\}_{i\in[2n],j\in[2m]}$, since $\boldsymbol{x}_i^{(\beta)\top}\mathbf{F}\boldsymbol{y}_j^{(\beta)} - \boldsymbol{x}_i^{(0)\top}\mathbf{F}y_j^{(0)} = 0$, by definition of the security game. Thus, by Theorem 2 (security of the single-ciphertext secure scheme), we obtain the lemma. □

Theorem 4 follows from Lemmas 6-10, and the fact that G_5 is independent from the bit $\beta \leftarrow_{\mathrm{R}} \{0,1\}$. □

4 Our Efficient Functional Encryption for Bilinear Maps in the GGM

In this section, we present a functional encryption scheme, $\mathsf{FE}_{\mathsf{GGM}}$, that supports the *bilinear map functionality*, and is proven secure against adaptive adversaries in the generic group model. In addition to be proven adaptive secure, this scheme enjoys a simpler structure, and is more efficient, as it admits shorter ciphertexts that comprise $2(n+m+1)$ group elements (as opposed to $6n+6m+2$ in the SXDH instantiation of the scheme of Section 3.2). For ease of exposition, the scheme is presented for the case in which the functions act over vectors of the same dimension n. It is easy to see that the case in which $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{Z}_p^n \times \mathbb{Z}_p^m$ with $n > m$ can be captured by padding $\boldsymbol{y}$ with zero entries.[8]

Technical Overview. We first provide a high-level view of the techniques used in this construction. The initial idea of the construction is to encrypt the two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ à la ElGamal in the two groups $\mathbb{G}_1$ and $\mathbb{G}_2$ respectively, i.e., the ciphertext includes $\boldsymbol{c} = [r \cdot \boldsymbol{a} + \boldsymbol{x}]_1$ and $\boldsymbol{d} = [s \cdot \boldsymbol{b} + \boldsymbol{y}]_2$ where r, s are randomly chosen and the vectors $([\boldsymbol{a}]_1, [\boldsymbol{b}]_2)$ are in the public key. At this point, we observe that, given $\boldsymbol{c}, \boldsymbol{d}$ and a function $\mathbf{F}$, one can use the bilinear map to compute $U = [(r \cdot \boldsymbol{a} + \boldsymbol{x})^\top \mathbf{F}(s \cdot \boldsymbol{b} + \boldsymbol{y})]_T$. This basic idea is similar to that of the scheme of Section 3.2. However, here we develop a different technique to enable decryption.

The basic scheme presented above is extended as follows. First, we let the secret key for function $\mathbf{F}$ be the element $[\boldsymbol{a}^\top \mathbf{F} \boldsymbol{b}]_1$. Now, if in the ciphertext we include the element $[rs]_2$, one can extract

$$[s\boldsymbol{x}^\top \mathbf{F}\boldsymbol{b} + r\boldsymbol{a}^\top \mathbf{F}\boldsymbol{y} + \boldsymbol{x}^\top \mathbf{F}\boldsymbol{y}]_T = U \cdot e([\boldsymbol{a}^\top \mathbf{F}\boldsymbol{b}]_1, [rs]_2)^{-1}.$$

Above the function's result is still "blinded" by cross terms $s(\boldsymbol{x}^\top \mathbf{F}\boldsymbol{b}) + r(\boldsymbol{a}^\top \mathbf{F}\boldsymbol{y})$. Our second idea, to solve this issue and enable full decryption, is to add to the ciphertext the ElGamal encryptions of the vectors $s \cdot \boldsymbol{x}$ and $r \cdot \boldsymbol{y}$. Namely, we add to the ciphertext the elements $\widehat{\boldsymbol{c}} = [t \cdot \boldsymbol{a} + s \cdot \boldsymbol{x}]_1$ and $\widehat{\boldsymbol{d}} = [z \cdot \boldsymbol{b} + r \cdot \boldsymbol{y}]_2$ for random t, z, and the element $[rs - t - z]_2$ (instead of $[rs]_2$). With all this information, one can compute the value U in the same way as above, and then use the public key $([\boldsymbol{a}]_1, [\boldsymbol{b}]_2)$ and the ciphertext components $\widehat{\boldsymbol{c}}, \widehat{\boldsymbol{d}}$ to compute

$$U' = [(t \cdot \boldsymbol{a} + s \cdot \boldsymbol{x})^\top \mathbf{F}\boldsymbol{b} + \boldsymbol{a}^\top \mathbf{F}(z \cdot \boldsymbol{b} + r \cdot \boldsymbol{y})]_T.$$

By a simple calculation, the function's result can be finally computed as

$$[\boldsymbol{x}^\top \mathbf{F}\boldsymbol{y}]_T = U \cdot {U'}^{-1} \cdot e([\boldsymbol{a}^\top \mathbf{F}\boldsymbol{b}]_1, [rs - z - t]_2)^{-1}.$$

As a final note, in the full scheme secret keys are slightly different, we randomize them in order to achieve collusion resistance.

Below we present our second FE scheme in detail.

$\mathsf{Setup}(1^\lambda, n)$ runs the bilinear group generator $\mathsf{bgp} \leftarrow_{\mathrm{R}} \mathcal{G}(1^\lambda)$ to obtain parameters $\mathsf{bgp} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e)$. Next, the algorithm samples a scalar $w \leftarrow_{\mathrm{R}} \mathbb{Z}_p$ and two vectors $\boldsymbol{a}, \boldsymbol{b} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^n$ uniformly at random. The message space is $\mathcal{M} \subseteq \mathbb{Z}_p^n \times \mathbb{Z}_p^n$ and the key space is the set of matrices $\mathcal{K} \subseteq \mathbb{Z}_p^{n \times n}$. It returns the master secret key $\mathsf{msk} := (w, \boldsymbol{a}, \boldsymbol{b})$, and the master public key $\mathsf{mpk} := (\mathsf{bgp}, [\boldsymbol{a}]_1, [\boldsymbol{b}]_2, [w]_2)$.

$\mathsf{KeyGen}(\mathsf{msk}, \mathbf{F})$ takes as input the master secret key msk and a matrix $\mathbf{F} \in \mathcal{K}$ and it returns a secret key $\mathsf{sk}_{\mathbf{F}} := (S_1, S_2, \mathbf{F}) \in \mathbb{G}_1^2 \times \mathcal{K}$ where S_1, S_2 are computed as follows. It samples a random $\gamma \leftarrow_{\mathrm{R}} \mathbb{Z}_p$ and computes

$$(S_1, S_2) := ([\boldsymbol{a}^\top \mathbf{F}\, \boldsymbol{b} + \gamma \cdot w]_1, [\gamma]_1).$$

[8] Furthermore, with a close look one can see that the last $n - m$ components of the vectors $[\boldsymbol{b}]_2$, $\boldsymbol{d}$ and $\widehat{\boldsymbol{d}}$ would become useless and thus can be discarded.

$\mathsf{Encrypt}(\mathsf{mpk}, (\boldsymbol{x}, \boldsymbol{y}))$ takes as input the master public key and a message consisting of two vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{M}$, and returns a ciphertext $\mathsf{Ct} := (\boldsymbol{c}, \widehat{\boldsymbol{c}}, \boldsymbol{d}, \widehat{\boldsymbol{d}}, E, \widehat{E})$ computed as follows.

Choose $r, s, t, z \in \mathbb{Z}_p$ uniformly at random and compute

$$\begin{aligned} \boldsymbol{c} &:= [r \cdot \boldsymbol{a} + \boldsymbol{x}]_1, & \widehat{\boldsymbol{c}} &:= [t \cdot \boldsymbol{a} + s \cdot \boldsymbol{x}]_1 \\ \boldsymbol{d} &:= [s \cdot \boldsymbol{b} + \boldsymbol{y}]_2, & \widehat{\boldsymbol{d}} &:= [z \cdot \boldsymbol{b} + r \cdot \boldsymbol{y}]_2 \\ E &:= [rs - z - t]_2 & \widehat{E} &:= [w(rs - z - t)]_2 \end{aligned}$$

$\mathsf{Decrypt}(\mathsf{sk}_{\mathbf{F}}, \mathsf{Ct})$ parsing $\mathsf{sk}_{\mathbf{F}} := (S_1, S_2, \mathbf{F})$ and $\mathsf{Ct} := (\boldsymbol{c}, \widehat{\boldsymbol{c}}, \boldsymbol{d}, \widehat{\boldsymbol{d}}, E, \widehat{E})$, it computes and outputs

$$V := \boldsymbol{c}^\top \mathbf{F}\, \boldsymbol{d} - [\boldsymbol{a}]_1{}^\top \mathbf{F}\, \widehat{\boldsymbol{d}} - \widehat{\boldsymbol{c}}^\top \mathbf{F}\, [\boldsymbol{b}]_2 - e(S_1, E) + e(S_2, \widehat{E}) \in \mathbb{G}_T.$$

4.1 Correctness

To see the correctness of our scheme, let

$$\begin{aligned} A &= \boldsymbol{c}^\top \mathbf{F}\, \boldsymbol{d} = [r \cdot \boldsymbol{a} + \boldsymbol{x}]_1^\top \mathbf{F}\, [s \cdot \boldsymbol{b} + \boldsymbol{y}]_2 \\ &= [(rs) \cdot \boldsymbol{a}^\top \mathbf{F}\, \boldsymbol{b} + r \cdot \boldsymbol{a}^\top \mathbf{F}\, \boldsymbol{y} + s \cdot \boldsymbol{x}^\top \mathbf{F}\, \boldsymbol{b} + \boldsymbol{x}^\top \mathbf{F}\, \boldsymbol{y}]_T \\ B &= [\boldsymbol{a}]_1{}^\top \mathbf{F}\, \widehat{\boldsymbol{d}} + \widehat{\boldsymbol{c}}^\top \mathbf{F}\, [\boldsymbol{b}]_2 = [\boldsymbol{a}]_1^\top \mathbf{F}\, [z \cdot \boldsymbol{b} + r \cdot \boldsymbol{y}]_2 + [t \cdot \boldsymbol{a} + s \cdot \boldsymbol{x}]_1^\top \mathbf{F}\, [\boldsymbol{b}]_2 \\ &= [z \cdot \boldsymbol{a}^\top \mathbf{F}\, \boldsymbol{b} + r \cdot \boldsymbol{a}^\top \mathbf{F}\, \boldsymbol{y} + t \cdot \boldsymbol{a}^\top \mathbf{F}\, \boldsymbol{b} + s \cdot \boldsymbol{x}^\top \mathbf{F}\, \boldsymbol{b}]_T \end{aligned}$$

and note that

$$\begin{aligned} A - B &= [(rs - t - z) \cdot \boldsymbol{a}^\top \mathbf{F}\, \boldsymbol{b} + \boldsymbol{x}^\top \mathbf{F}\, \boldsymbol{y}]_T = e(S_1 - [w \cdot \gamma]_1, E) + [\boldsymbol{x}^\top \mathbf{F}\, \boldsymbol{y}]_T \\ &= e(S_1, E) - e(S_2, \widehat{E}) + [\boldsymbol{x}^\top \mathbf{F}\, \boldsymbol{y}]_T \end{aligned}$$

Since $V = A - B - e(S_1, E) + e(S_2, \widehat{E})$ it is easy to see that $V = [\boldsymbol{x}^\top \mathbf{F}\, \boldsymbol{y}]_T$.

5 Proof of Security of $\mathsf{FE}_{\mathsf{GGM}}$

In this section we state and prove the security of the functional encryption scheme $\mathsf{FE}_{\mathsf{GGM}}$ of Section 4 in the generic group model.

Theorem 5. *The functional encryption scheme $\mathsf{FE}_{\mathsf{GGM}}$ described in Section 4 satisfies security against chosen-plaintext attacks (i.e., indistinguishability-based security) in the generic bilinear group model. Precisely, for every adversary $\mathcal{A}$ which makes at most Q key derivation oracle queries and $\widetilde{Q}$ generic group oracle queries its advantage is*

$$\mathbf{Adv}^{\mathsf{ind\text{-}fe\text{-}cpa}}_{\mathsf{FE}_{\mathsf{GGM}}, \mathcal{A}}(\lambda) \leq \frac{5(6n + 6 + \widetilde{Q} + 2Q)^2}{p}$$

The proof consists of two main steps. We first state and prove a master theorem that shows hardness in the generic bilinear group model for a broad family of interactive decisional problems, notably a family which includes the indistinguishability-based experiment for our functional encryption scheme. Slightly more in detail, our master theorem states that these problems are generically hard under a certain algebraic side condition on the distribution of the elements received by the adversary. Then, following the guidelines of our master theorem, the second step of the proof consists in showing that the scheme $\mathsf{FE}_{\mathsf{GGM}}$ meets the algebraic side condition of our master theorem.

State: Lists L_1, L_2, L_T over $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ respectively.

Initialization: Lists L_1, L_2, L_T sampled according to distribution $\mathcal{D}$.

Oracles: The oracles provide black-box access to the group operations, the bilinear map and equalities:

- For all $s \in \{1, 2, T\}$: $\mathsf{add}_s(h_1, h_2)$ appends $L_s[h_1] + L_s[h_2]$ to L_s and returns its handle $(s, |L_s|)$.
- For all $s \in \{1, 2, T\}$: $\mathsf{neg}_s(h)$ appends $-L_s[h]$ to L_s and returns its handle $(s, |L_s|)$.
- $\mathsf{map}_e(h_1, h_2)$ appends $e(L_1[h_1], L_2[h_2])$ to L_T and returns its handle $(T, |L_T|)$.
- For all $s \in \{1, 2, T\}$: $\mathsf{eq}_s(h_1, h_2)$ returns 1 if $L_s[h_1] = L_s[h_2]$ and 0 otherwise.

All queries return $\perp$ when given invalid indices.

Fig. 10. Generic group model $\mathsf{GGM}_{\mathcal{D}}^{\mathsf{bgp}}$ for bilinear group setting $\mathsf{bgp} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2)$ and distribution $\mathcal{D}$.

5.1 Generic Bilinear Group Model for Interactive Problems

In this section we introduce the generic group model framework that we use to prove the security of our functional encryption scheme. We adopt the framework of Barthe et al. [8] for analyzing assumptions in generic k-linear groups, and specialize their definitions to our case of interest, that are asymmetric (Type-III) bilinear groups. In addition, since the results in [8] for interactive assumptions can only model *computational* problems, we provide extensions that allow us to deal with interactive *decisional* problems.

Generic Bilinear Group Model. Let $\mathsf{bgp} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g_1, g_2)$ be a bilinear group setting, L_1, L_2, L_T be lists of group elements in $\mathbb{G}_1, \mathbb{G}_2$ and $\mathbb{G}_T$ respectively, and let $\mathcal{D}$ be a distribution over L_1, L_2, L_T. The generic model for a bilinear group setting bgp and a distribution $\mathcal{D}$ is described compactly in Figure 5.1. In this model, the challenger first initializes the lists L_1, L_2, L_T by sampling the group elements according to $\mathcal{D}$, and the adversary receives handles for the elements in the lists. For $s \in \{1, 2, T\}$, $L_s[h]$ denotes the h-th element in the list L_s. The handle to this element is simply the pair (s, h). An adversary running in the generic bilinear group model can apply group operations and bilinear maps to the elements in the lists. To do this, the adversary has to call the appropriate oracle specifying handles for the input elements. The challenger computes the result of a query, stores it in the corresponding list, and returns to the adversary its (newly created) handle. Handles are not unique (i.e., the same group element may appear more than once in a list under different handles), but the adversary is provided with an equality oracle to check if two handles refer to the same group element. This generic group model follows closely that of Maurer [32] (which slightly differs in presentation, although it is equivalent, to that of Shoup [39]) in that the adversary has access to the state of the challenger via handles, and equality queries have "free" cost in the sense that they are not counted for measuring the adversary's computational complexity.

Below we recall a specific class of distributions on lists of group elements that is used in our work. Intuitively, it considers group elements that are generated by sampling random values $x_1, \ldots, x_n \leftarrow_{\mathrm{R}} \mathbb{Z}_p$ and by computing $[p(x_1, \ldots, x_n)]_s \in \mathbb{G}_s$ for some multivariate polynomial p.

Definition 10 (Polynomially Induced Distributions [8]). *Let $\boldsymbol{P} = (P_1, P_2, P_T)$ be three lists of polynomials in $\mathbb{Z}_p[X_1, \ldots, X_n]$ such that each list contains the constant polynomial* 1*. We define the distribution $\mathcal{D}_{\boldsymbol{P}}$ as follows: uniformly sample a vector $\boldsymbol{x} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^n$ and return three lists $\boldsymbol{L} = (L_1, L_2, L_T)$ where, for every $s \in \{1, 2, T\}$, $L_s = \{[p_1(\boldsymbol{x})]_s, \ldots, [p_{|P_s|}(\boldsymbol{x})]_s\}$ with $p_j(\boldsymbol{X})$ being the j-th polynomial in the list P_s. We compactly denote this process as $\boldsymbol{L} \leftarrow \mathcal{D}_{\boldsymbol{P}}$. A distribution $\mathcal{D}$ is called polynomially induced if $\mathcal{D} = \mathcal{D}_{\boldsymbol{P}}$ for some $\boldsymbol{P}$.*

To give an example, the input to an adversary for the computational Diffie-Hellman assumption (in $\mathbb{G}_1$) can be described as a polynomially induced distribution where $P_1 = (1, X_1, X_2)$ contains three polynomials in $\mathbb{Z}_p[X_1, X_2]$.

Definition 11 (Completion). *Given lists of polynomials $\boldsymbol{P} = (P_1, P_2, P_T)$, we define their completion $\mathcal{C}(\boldsymbol{P})$ as*

$$\mathcal{C}(\boldsymbol{P}) := P_T \cup \{p_{1,i}(\boldsymbol{X}) \cdot p_{2,j}(\boldsymbol{X}) : \forall p_{1,i} \in P_1, p_{2,j} \in P_2\}$$

Intuitively speaking, for lists of polynomials $\boldsymbol{P}$ their completion represents the list of all polynomials that can be computed by the adversary by applying bilinear maps (i.e., multiplications) to the polynomials in $\boldsymbol{P}$. Our definition given above is a specialization (which gets somewhat simplified) of the completion definition for k-linear groups given in [8].

To give an example, if $\boldsymbol{P} = (P_1, P_2, P_T)$ with $P_1 = \{1, X_1, X_2\}$, $P_2 = \{1, X_2\}$, $P_T = \{1\}$, then its completion is the list $\{1, X_1, X_2, X_1X_2, X_2^2\}$.

Symbolic Group Model. The *symbolic group model* for a bilinear group setting bgp and a polynomially induced distribution $\mathcal{D}_{\boldsymbol{P}}$, denoted as $\mathsf{SGM}^{\mathsf{bgp}}_{\mathcal{D}_{\boldsymbol{P}}}$, gives to the adversary the same interface as the corresponding generic group model, except that internally the challenger stores lists of polynomials in $\mathbb{Z}_p[X_1, \ldots, X_n]$ instead of lists of group elements. The oracles add_s, neg_s, map and eq_s compute addition, negation, multiplication, and equality in the polynomial ring. For any event $\mathcal{E}$ in the generic group model, we define a symbolic version of it, $S(\mathcal{E})$, where equalities over group elements are replaced by equalities over polynomials. In the case where $\mathcal{E}$ is an event which does not involve equality tests on group elements (e.g., in decisional problems where the finalization event can be a simple check $\beta \overset{?}{=} 1$ on the adversary's output bit) it holds $S(\mathcal{E}) = \mathcal{E}$.

Generic and Symbolic Group Model for Simple Interactive Problems. The definitions given so far work for adversaries that receive statically defined lists at the beginning of the game, and then can interact through the oracles to compute group operations and bilinear maps over them. In what follows we generalize the generic and symbolic group models in order to capture a family of interactive decisional problems which includes the indistinguishability security experiment of our functional encryption scheme. The difference in modeling interactive problems in the generic (and symbolic) group model is that the adversary is provided with access to additional oracles that compute further operations on the elements stored in the lists maintained by the challenger in its state. To formalize this setting, we build on the notion of oracles given by Barthe et al. [8] to model interactive assumptions. One difference, though, is that in our work we consider oracles that do *not* take as inputs group elements (i.e., handles to elements in the challenger's lists) from the adversary – we call these problems "*simple* interactive problems". In other words, we consider oracles that take as inputs scalar parameters in $\mathbb{Z}_p$ and return handles to group elements that are computed from these scalar parameters, values randomly sampled by $\mathcal{D}_{\boldsymbol{P}}$ and other $\mathbb{Z}_p$ values freshly sampled by the oracle itself. This restriction on the type of oracles simplifies the presentation, and allows us to state a master theorem which deals with interactive *decisional* problems, whereas the master theorem for interactive assumptions given in [8] can only deal with computational problems.

We begin by defining the notion of an oracle in the generic bilinear group model.

Definition 12 (Oracles in the generic bilinear group model). *An oracle is a tuple $\mathcal{O} = (Q', \ell, m, \boldsymbol{p}, \boldsymbol{v})$ where:*

- *Q' is the number of oracle queries that are allowed;*

- *ℓ is the number of variables $\delta_1, \ldots, \delta_\ell$ in $\mathbb{Z}_p$ that are taken as scalar parameters;*
- *m is the number of values $\omega_1, \ldots, \omega_m$ randomly sampled by the oracle in $\mathbb{Z}_p$;*
- *$\boldsymbol{p} = (p_1, \ldots, p_c)$ is a vector of polynomials in $\mathbb{Z}_p[X_1, \ldots, X_n, \Delta_1, \ldots, \Delta_\ell, \Omega_1, \ldots, \Omega_m]$ that describes the c values returned by the oracle;*
- *$\boldsymbol{v} = (v_1, \ldots, v_c)$ is a vector of indices such that every $v_i \in \{1, 2, T\}$ describes in which group the polynomial p_i belongs to.*

Basically, in the generic bilinear group model, the oracle takes as input a vector $\boldsymbol{\delta} \in \mathbb{Z}_p^\ell$ from the adversary and returns handles to group elements $[p_1(\boldsymbol{x}, \boldsymbol{\delta}, \boldsymbol{\omega})]_{v_1}, \ldots, [p_c(\boldsymbol{x}, \boldsymbol{\delta}, \boldsymbol{\omega})]_{v_c}$ computed by sampling $\boldsymbol{\omega} \leftarrow_{\mathrm{R}} \mathbb{Z}_p^m$. In the symbolic group model the oracle has the same interface, except that: instead of sampling new values ω_i, it creates new formal variables Ω_i; instead of returning handle to group elements, it returns handles to formal polynomials in the polynomial ring augmented with the newly created formal variables, i.e., $p_j(\boldsymbol{X}, \boldsymbol{\delta}, \boldsymbol{\Omega}) \in \mathbb{Z}_p[\boldsymbol{X}, \boldsymbol{\Omega}]$.

As an example, the reader may consider the key derivation oracle corresponding to the functional encryption scheme $\mathsf{FE}_{\mathsf{GGM}}$. It takes as input $\ell = n^2$ values $\{f_{i,j}\}_{i,j \in [n]}$ which are the coefficients of the bilinear form $\mathbf{F}$; it samples $m = 1$ random value $\omega_1 = \gamma$; returns $c = 2$ elements of $\mathbb{G}_1$ which can be described by polynomials $\boldsymbol{A}^\top \mathbf{F} \boldsymbol{B} + \Omega_1 W$ and Ω_1 in $\mathbb{Z}_p[\boldsymbol{A}, \boldsymbol{B}, W, \mathbf{F}, \Omega_1]$.

Now we state and prove a theorem which shows that one can switch from a generic group model experiment to a corresponding symbolic group model experiment. This theorem extends to interactive problems of Theorem 1 in [8].

Theorem 6 (From Generic to Symbol Group Model with Oracles). *Let bgp be a bilinear group setting, where p is prime, $\mathcal{D}_{\boldsymbol{P}}$ a polynomially induced distribution, $\mathcal{O} = (Q', \ell, m, \boldsymbol{p}, \boldsymbol{v})$ an oracle such that $c = |\boldsymbol{p}|$, $\mathcal{A}$ an adversary performing at most Q generic group oracle queries, and $\mathcal{E}$ an event without group equality tests. If d is an upper bound on the degree of the polynomials occurring in the internal state of $\mathsf{SGM}_{\mathcal{D}_{\boldsymbol{P}}}^{\mathsf{bgp}}$, and $N = |P_1| + |P_2| + |P_T|$ is the sum of the lists cardinalities, then*

$$\left| \Pr[\mathsf{GGM}_{\mathcal{D}_{\boldsymbol{P}}, \mathcal{O}}^{\mathsf{bgp}}(\mathcal{A}) : \mathcal{E}] - \Pr[\mathsf{SGM}_{\mathcal{D}_{\boldsymbol{P}}, \mathcal{O}}^{\mathsf{bgp}}(\mathcal{A}) : S(\mathcal{E})] \right| \leq \frac{d \cdot (N + Q + c \cdot Q')^2}{2p}$$

where the probability is taken over the coins of $\mathsf{GGM}_{\mathcal{D}_{\boldsymbol{P}}}^{\mathsf{bgp}}$ and $\mathcal{A}$.

Proof. The proof of this theorem is essentially the same as that of Theorem 1 in [8], which however does not consider oracles. Given the similarity to [8], we only provide an intuition here. The basic idea is that the adversary, who only sees handles and the outcome of equality queries, can notice a difference between the two games only if an equality query would be answered differently. For a single equality check, the probability of seeing a difference (that occurs when two polynomials $f_1 \neq f_2$ are different in SGM, but $f_1(\widetilde{\boldsymbol{x}}) = f_2(\widetilde{\boldsymbol{x}})$ for a random $\widetilde{\boldsymbol{x}}$ in GGM) is bounded using the Schwartz-Zippel lemma, and is $\leq d/p$. The final bound is then obtained by a union bound on the maximum number of equality checks between group elements (resp. polynomials) in the lists. This number is upper bounded by $T^2/2$, where T is the maximal length of the lists, which is $T = N + Q + c \cdot Q'$ for an adversary that makes at most Q queries to the generic group oracles, and has additional access to $\mathcal{O}$ which can be queried at most Q' times, each time returning c polynomials. □

Looking ahead to defining our master theorem for simple interactive decisional problems, we introduce a notion of parametric completion which works in this interactive setting where the adversary gets access to more polynomials in addition to those statically defined in the initial lists $\boldsymbol{P}$.

Definition 13 (Parametric Completion). *Given lists of polynomials $\boldsymbol{P} = (P_1, P_2, P_T)$ and an oracle $\mathcal{O} = (Q', \ell, m, \boldsymbol{p}, \boldsymbol{v})$, we define their parametric completion $\mathcal{C}^{\mathcal{O}}(\boldsymbol{P})$ as follows. Assuming that the c polynomials in $\boldsymbol{p}$ are in $\mathbb{Z}_p[\boldsymbol{X}, \boldsymbol{\varDelta}, \boldsymbol{\varOmega}]$, we define an extended set of formal variables*

$$\widehat{\boldsymbol{\varDelta}} = \{\varDelta_{i,j}\}_{i \in [\ell], j \in [Q']}, \quad \widehat{\boldsymbol{\varOmega}} = \{\varOmega_{i,j}\}_{i \in [m], j \in [Q']}$$

The parametric completion $\mathcal{C}^{\mathcal{O}}(\boldsymbol{P})$ consists of polynomials in $\mathbb{Z}_p[\boldsymbol{X}, \widehat{\boldsymbol{\varDelta}}, \widehat{\boldsymbol{\varOmega}}]$, and is computed as follows:

```
1. $\boldsymbol{P}' := \boldsymbol{P}$
2. foreach $i \in [Q']$:
3.   foreach $j \in [c]$:
4.     $p'_j := p_j(\varDelta_1 := \varDelta_{1,i}, \ldots, \varDelta_\ell := \varDelta_{\ell,i}, \varOmega_1 := \varOmega_{1,i}, \ldots, \varOmega_m := \varOmega_{m,i})$
5.     $P'_{v_j} := P'_{v_j} \cup \{p'_j\}$
6. $\mathcal{C}^{\mathcal{O}}(\boldsymbol{P}) := \mathcal{C}(\boldsymbol{P}')$
```

Basically, for every query and every polynomial which is to be returned by the oracle, line 4 redefines the polynomial by making a change of variables (so that the newly introduced variables are unique in the game instead of being only locally unique in the query), while line 5 simply adds this polynomial to the corresponding list (i.e., group) according to the index v_j. Finally, the parametric completion is just a completion (as per Definition 11) computed on the lists of polynomials $\boldsymbol{P}'$ which include the initial lists $\boldsymbol{P}$ plus all the polynomials returned by the oracle. We also note that the notion extends naturally to be parametrized by more than one oracle.

5.2 A Master Theorem for Simple Interactive Decisional Problems

Equipped with the framework and the tools introduced in the previous section, we are now ready to state our master theorem. First, we define what we call simple interactive decisional problems.

Definition 14 (Simple Interactive Decisional Problems). *A simple interactive decisional problem in the generic and symbolic bilinear group model for oracles $\mathcal{O} = (Q', \ell, m, \boldsymbol{p}, \boldsymbol{v})$, $\mathcal{O}_{ch} = (1, \ell^*, m^*, \boldsymbol{f}, \boldsymbol{v}^*)$, and $\mathcal{O}'_{ch} = (1, \ell^*, m^*, \boldsymbol{f}', \boldsymbol{v}^*)$, and a legitimacy predicate H is an experiment where:*

- *The adversary $\mathcal{A}$ gets the same input and the same oracles as in Figure 5.1.*
- *$\mathcal{A}$ can interact with two more oracles, either $\mathcal{O}$ and $\mathcal{O}_{ch}$, or $\mathcal{O}$ and $\mathcal{O}'_{ch}$, such that $\mathcal{O}_{ch}$ (resp. $\mathcal{O}'_{ch}$) can be queried only once.*
- *$\mathcal{A}$ can make (adaptive) queries to its oracles under the restriction that $\mathcal{A}$ is "legitimate", where legitimacy is defined by some predicate H over its oracle queries. Specifically, if $\widehat{\boldsymbol{\delta}}^* \in \mathbb{Z}_p^{\ell^*}$ is $\mathcal{A}$'s query to oracle $\mathcal{O}_{ch}$ (or $\mathcal{O}'_{ch}$), and $\widehat{\boldsymbol{\delta}} = (\widehat{\boldsymbol{\delta}}_1, \ldots, \widehat{\boldsymbol{\delta}}_{Q'}) \in \mathbb{Z}_p^{\ell \cdot Q'}$ are the Q' queries of $\mathcal{A}$ to oracle $\mathcal{O}$, then H is defined as a predicate $H(\widehat{\boldsymbol{\delta}}, \widehat{\boldsymbol{\delta}}^*) \in \{0, 1\}$.*
- *$\mathcal{A}$ returns a bit β, and the finalization event $\mathcal{E}$ is "$\beta \overset{?}{=} 1$".*

Note that the two oracles $\mathcal{O}_{ch}, \mathcal{O}'_{ch}$ differ only in their output polynomials. Namely, it can be $\boldsymbol{f} \neq \boldsymbol{f}'$ (while their length is clearly the same).

Below we state and prove our master theorem whose goal is to bound the difference between the probabilities of the winning event $\mathcal{E}$ in the two executions of the experiment, provided that a certain algebraic condition on the parametric completions is met.

Theorem 7 (Master Theorem for Simple Interactive Decisional Problems). *Let* bgp *be a bilinear group setting,* $\mathcal{D}_{\boldsymbol{P}}$ *be a polynomially-induced distribution, and* $\mathcal{O} = (Q', \ell, m, \boldsymbol{p}, \boldsymbol{v})$ *be an oracle. Furthermore, let* $\mathcal{O}_{ch} = (1, \ell^*, m^*, \boldsymbol{f}, \boldsymbol{v}^*)$ *and* $\mathcal{O}'_{ch} = (1, \ell^*, m^*, \boldsymbol{f}', \boldsymbol{v}^*)$ *be two other oracles. Let* $N = |P_1| + |P_2| + |P_T|$*,* $c = |\boldsymbol{p}|$*,* $c^* = |\boldsymbol{f}| = |\boldsymbol{f}'|$*,* $r = |\mathcal{C}^{\mathcal{O},\mathcal{O}_{ch}}(\boldsymbol{P})|$*, and let* d *denote an upper bound on the total degrees of the polynomials in the parametric completions. If for all vectors* $\widehat{\boldsymbol{\delta}} \in \mathbb{Z}_p^{\ell Q'}, \widehat{\boldsymbol{\delta}}^* \in \mathbb{Z}_p^{\ell^*}$ *such that* $H(\widehat{\boldsymbol{\delta}}, \widehat{\boldsymbol{\delta}}^*) = 1$ *it holds*

$$\{\boldsymbol{k} \in \mathbb{Z}_p^r \mid \boldsymbol{k} \cdot C = 0\} = \{\boldsymbol{k} \in \mathbb{Z}_p^r \mid \boldsymbol{k} \cdot C' = 0\}, \tag{1}$$

where $C = \mathcal{C}^{\mathcal{O},\mathcal{O}_{ch}}(\boldsymbol{P})(\widehat{\boldsymbol{\Delta}} = \widehat{\boldsymbol{\delta}}, \widehat{\boldsymbol{\Delta}}^* = \widehat{\boldsymbol{\delta}}^*)$ *and* $C' = \mathcal{C}^{\mathcal{O},\mathcal{O}'_{ch}}(\boldsymbol{P})(\widehat{\boldsymbol{\Delta}} = \widehat{\boldsymbol{\delta}}, \widehat{\boldsymbol{\Delta}}^* = \widehat{\boldsymbol{\delta}}^*)$*, then*

$$\left|\Pr[\mathsf{GGM}^{\mathsf{bgp}}_{\mathcal{D}_{\mathcal{P}},\mathcal{O},\mathcal{O}_{ch}}(\mathcal{A}) : \mathcal{E}] - \Pr[\mathsf{GGM}^{\mathsf{bgp}}_{\mathcal{D}_{\mathcal{P}},\mathcal{O},\mathcal{O}'_{ch}}(\mathcal{A}) : \mathcal{E}]\right| \leq \frac{(N + Q + cQ' + c^*)^2 \cdot d}{p}$$

holds for all legitimate adversaries $\mathcal{A}$ *that perform at most* Q *group operations.*

Proof. To prove the theorem we first apply Theorem 6 in order to switch the two experiments from the generic to the symbolic group model.

$$\left|\Pr[\mathsf{GGM}^{\mathsf{bgp}}_{\mathcal{D}_{\boldsymbol{P}},\mathcal{O},\mathcal{O}_{ch}}(\mathcal{A}) : \mathcal{E}] - \Pr[\mathsf{SGM}^{\mathsf{bgp}}_{\mathcal{D}_{\boldsymbol{P}},\mathcal{O},\mathcal{O}_{ch}}(\mathcal{A}) : S(\mathcal{E})]\right| \leq \frac{(N + Q + cQ' + c^*)^2 \cdot d}{2p}$$

$$\left|\Pr[\mathsf{SGM}^{\mathsf{bgp}}_{\mathcal{D}_{\boldsymbol{P}},\mathcal{O},\mathcal{O}'_{ch}}(\mathcal{A}) : S(\mathcal{E})] - \Pr[\mathsf{GGM}^{\mathsf{bgp}}_{\mathcal{D}_{\boldsymbol{P}},\mathcal{O},\mathcal{O}'_{ch}}(\mathcal{A}) : \mathcal{E}]\right| \leq \frac{(N + Q + cQ' + c^*)^2 \cdot d}{2p}$$

To complete the proof we claim that

$$\left|\Pr[\mathsf{SGM}^{\mathsf{bgp}}_{\mathcal{D}_{\mathcal{P}},\mathcal{O},\mathcal{O}_{ch}}(\mathcal{A}) : S(\mathcal{E})] - \Pr[\mathsf{SGM}^{\mathsf{bgp}}_{\mathcal{D}_{\mathcal{P}},\mathcal{O},\mathcal{O}'_{ch}}(\mathcal{A}) : S(\mathcal{E})]\right| = 0$$

Since we are quantifying over legitimate adversaries, we take for granted that $\mathcal{A}$'s queries are such that $H(\widehat{\boldsymbol{\delta}}, \widehat{\boldsymbol{\delta}}^*) = 1$. $\mathcal{A}$'s view in the symbolic game depends only on the outcome of the equality checks which are performed on the polynomials appearing in the lists stored by the challenger. At this point, the key observation is that the parametric completion $C = \mathcal{C}^{\mathcal{O},\mathcal{O}_{ch}}(\boldsymbol{P})(\widehat{\boldsymbol{\Delta}} = \widehat{\boldsymbol{\delta}}, \widehat{\boldsymbol{\Delta}}^* = \widehat{\boldsymbol{\delta}}^*)$ can be viewed as the generating set of a vector space V which describes all the polynomials computable by the adversary starting from the polynomials in $\boldsymbol{P}$ and the polynomials returned by the oracles. So, every polynomial $v \in V$ can be expressed as a linear combination of polynomials in C (i.e., $v = \boldsymbol{\lambda} \cdot C$ for some $\boldsymbol{\lambda}$) and $K = \{\boldsymbol{k} \in \mathbb{Z}_p^r \mid \boldsymbol{k} \cdot C = 0\}$ is the kernel of this linear map. Moreover, since the lists P_1, P_2 are assumed to contain the constant polynomial 1, we note that the parametric completion C in the target group is sufficient to express all polynomials in V. Therefore, the side condition on the equality of the kernels of the two linear maps (i.e. condition 1) means that the adversary sees exactly the same equalities in the two experiments. To see this, consider an execution of the SGM experiment where the adversary has two handles h_1, h_2, and assume that these point to polynomials v_1, v_2 in the left game (i.e., with oracle $\mathcal{O}_{ch}$) and v'_1, v'_2 in the right game (i.e., with oracle $\mathcal{O}'_{ch}$), such that $v_1 = v_2$ (i.e., $\mathsf{eq}_s(h_1, h_2) = 1$ in the left game) and $v'_1 \neq v'_2$ (i.e., $\mathsf{eq}_s(h_1, h_2) = 0$ in the right game). Notice that in both experiments the polynomial v_l (resp. v'_l) can be expressed using the same linear combination of elements in the respective completion, i.e., for $l = 1, 2$, in the left game we have $v_l = \boldsymbol{\lambda}_l \cdot C$ whereas in the right game we have $v'_l = \boldsymbol{\lambda}_l \cdot C'$. However, this means that $(\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2) \cdot C = 0$ whereas $(\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2) \cdot C' \neq 0$, which contradicts our side condition. □

5.3 Security of the Functional Encryption Scheme $\mathsf{FE}_{\mathsf{GGM}}$

In this section we use the generic group framework presented in the previous section to prove Theorem 5. We proceed as follows. First, we show that the indistinguishability security game for the scheme $\mathsf{FE}_{\mathsf{GGM}}$ is a simple interactive decisional problem as per Definition 14, and thus it fits our Theorem 7. Next, we give the core part of the proof which is to show that the scheme is symbolically hard, in the sense that it satisfies the side condition of equation (1) in Theorem 7.

Indistinguishability Security is a Simple Interactive Decisional Problem. Let us consider the indistinguishability security experiment for our scheme $\mathsf{FE}_{\mathsf{GGM}}$ in the generic bilinear group model. At the beginning the adversary is given handles for the following lists

$$\begin{aligned} L_1 &= \{[1]_1, [a_1]_1, \ldots, [a_n]_1\} \\ L_2 &= \{[1]_2, [w]_2, [b_1]_2, \ldots, [b_n]_2\} \\ L_T &= \{[1]_T\} \end{aligned}$$

which can be seen as output of the polynomially induced distribution $(L_1, L_2, L_T) \leftarrow \mathcal{D}_{\boldsymbol{P}}$, where

$$P_1 = \{1, A_1, \ldots, A_n\},\ P_2 = \{1, W, B_1, \ldots, B_n\},\ P_T = \{1\}$$

are lists of polynomials over $\mathbb{Z}_p[\boldsymbol{A}, \boldsymbol{B}, W]$.

The adversary is also given access to the key derivation oracle that we can write as $\mathcal{O} = (\cdot, n^2, 1, (p_1, p_2), (1, 1))$ since it can be queried an unbounded number of times, it takes as input the description of a quadratic form which is an $(n \times n)$-dimensional matrix $\mathbf{F} = (f_{i,j})$, samples a single value γ, and outputs two elements of $\mathbb{G}_1$ which can be described with polynomials

$$p_1 = \sum_{i,j \in [n]} f_{i,j} \cdot A_i B_j + \mathit{\Gamma} \cdot W, \quad p_2 = \mathit{\Gamma} \quad \in \mathbb{Z}_p[\boldsymbol{A}, \boldsymbol{B}, W, \mathbf{F}, \mathit{\Gamma}].$$

Also, $\mathcal{A}$ can query the challenge oracle that is either $\mathcal{O}_{ch}(1, 4n, 4, \boldsymbol{p}^*, \boldsymbol{v}^*)$ or $\mathcal{O}'_{ch}(1, 4n, 4, \boldsymbol{p}^{*\prime}, \boldsymbol{v}^*)$. To see the definition of these oracles, note that they can be queried only once, they take as input two challenge messages $(\boldsymbol{x}, \boldsymbol{y}), (\boldsymbol{x}', \boldsymbol{y}')$, sample four random values r, s, t, z, and output polynomials corresponding to the ciphertexts, that are either:

$$\begin{aligned} \boldsymbol{p}^* = (\ & \{RA_i + X_i\}_{i=1}^n, \quad \{TA_i + SX_i\}_{i=1}^n, \\ & \{SB_i + Y_i\}_{i=1}^n, \quad \{ZB_i + RY_i\}_{i=1}^n, \\ & RS - Z - T, \qquad W(RS - Z - T)\ ) \end{aligned}$$

or

$$\begin{aligned} \boldsymbol{p}^{*\prime} = (\ & \{RA_i + X'_i\}_{i=1}^n, \quad \{TA_i + SX'_i\}_{i=1}^n, \\ & \{SB_i + Y'_i\}_{i=1}^n, \quad \{ZB_i + RY'_i\}_{i=1}^n, \\ & RS - Z - T, \qquad W(RS - Z - T)\ ) \end{aligned}$$

over $\mathbb{Z}_p[\boldsymbol{A}, \boldsymbol{B}, W, \boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{X}', \boldsymbol{Y}', R, S, T, Z]$.

Moreover, for an adversary $\mathcal{A}$ that makes Q queries $\mathbf{F}_1, \ldots, \mathbf{F}_Q \in \mathbb{Z}_p^{n \times n}$ to the key derivation oracle, one query $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{x}', \boldsymbol{y}' \in \mathbb{Z}_p^n$ to the challenge oracle, and returns a bit β, then by the security definition we have that $\mathcal{A}$ is *legitimate* if "$H(\mathbf{F}_1, \ldots, \mathbf{F}_Q, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{x}', \boldsymbol{y}') = 1$", where the predicate H is true iff $\boldsymbol{x}^\top \mathbf{F}_i \boldsymbol{y} = \boldsymbol{x}'^\top \mathbf{F}_i \boldsymbol{y}'$ for all $i = 1$ to Q.

It is easy to see that the indistinguishability security experiment for the FE scheme $\mathsf{FE}_{\mathsf{GGM}}$ is a simple interactive decisional problem as per Definition 14. In order to obtain a proof of Theorem 5, then we invoke our master Theorem 7.

Instantiating the master theorem. Before focusing on the main part of the proof, which is to show the satisfaction of the side condition, we briefly show how the bound of Theorem 5 is obtained. This follows by observing that: the sum of lists cardinalities is $2(n+1)+2$, the key derivation and challenge oracles give $2Q$ and $4n+2$ polynomials respectively, and, as we shall see a bit later, the maximal total degree of polynomials in the parametric completions is $d = 5$.

Satisfaction of the master theorem side condition. The remaining part of the proof focuses on showing that the interactive decisional problem corresponding to the security of the functional encryption scheme $\mathsf{FE}_{\mathsf{GGM}}$ satisfies the side condition of equation (1). To this end, our first step is to compute the parametric completions $\mathcal{C}^{\mathcal{O},\mathcal{O}_{ch}}(\boldsymbol{P})$ and $\mathcal{C}^{\mathcal{O},\mathcal{O}'_{ch}}(\boldsymbol{P})$. In the completions computation we consider directly the adversary's queries as scalars instead of formal variables. Namely, we consider the polynomials in the parametric completions evaluated at $\boldsymbol{X} = \boldsymbol{x}, \boldsymbol{Y} = \boldsymbol{y}, \boldsymbol{X}' = \boldsymbol{x}', \boldsymbol{Y}' = \boldsymbol{y}'$ and $\widehat{\boldsymbol{F}}_k = \boldsymbol{f}^{(k)}, \forall k \in [Q]$; this is indeed what we need for analyzing the side condition of equation (1).

Notation. In the rest of the proof, for presentation's convenience we use the following vector notation to express a bilinear map:

$$\langle \boldsymbol{f}, \boldsymbol{x} \otimes \boldsymbol{y} \rangle = \sum_{i,j \in [n]} f_{i,j} x_i y_j$$

Above, $\boldsymbol{f}$ is the n^2-dimensional vector obtained by concatenating all the rows of $\mathbf{F}$, i.e., $\boldsymbol{f} = (f_{1,1}, f_{1,2}, \ldots, f_{1,n}, f_{2,1}, \ldots, f_{n,n-1}, f_{n,n})$ For any n-dimensional vectors $\boldsymbol{x}, \boldsymbol{y}$, we denote by $\boldsymbol{x} \otimes \boldsymbol{y}$ their tensor product that we write as an n^2-dimensional vector $(x_i y_j)_{i,j}$ where the entries i, j are ordered lexicographically, e.g., $\boldsymbol{x} \otimes \boldsymbol{y} = (x_1 y_1, x_1 y_2, \ldots, x_n y_{n-1}, x_n y_n)$.

Parametric Completions. Consider an adversary $\mathcal{A}$ which queries $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{x}', \boldsymbol{y}'$ to the challenge oracle $\mathcal{O}_{ch}$, and $\boldsymbol{f}^{(1)}, \ldots, \boldsymbol{f}^{(Q)}$ to the key derivation oracle. The computation of the parametric completion $\mathcal{C}^{\mathcal{O},\mathcal{O}_{ch}}(\boldsymbol{P})$ (evaluated at $\boldsymbol{X} = \boldsymbol{x}, \boldsymbol{Y} = \boldsymbol{y}, \boldsymbol{X}' = \boldsymbol{x}', \boldsymbol{X}' = \boldsymbol{y}'$ and $\widehat{\boldsymbol{F}}_k = \boldsymbol{f}^{(k)}, \forall k \in [Q]$) first builds the following lists:

$$\begin{aligned}
&P'_1 = \{1\} \cup \{A_i,\ RA_i + x_i,\ TA_i + x_i S\}_{i \in [n]} \cup \{\langle \boldsymbol{f}^{(k)}, \boldsymbol{A} \otimes \boldsymbol{B} \rangle + \mathit{\Gamma}_k W,\ \mathit{\Gamma}_k\}_{k \in [Q]} \\
&P'_2 = \{1,\ W,\ RS - Z - T,\ W(RS - Z - T)\} \cup \{B_i,\ SB_i + y_i,\ ZB_i + y_i R\}_{i \in [n]}, \\
&P'_T = \{1\}
\end{aligned}$$

The last step of the parametric completion computation, $\mathcal{C}(\boldsymbol{P'})$, then yields:

$$
\begin{aligned}
C = \{&1,\ W,\ RS - Z - T,\ W(RS - Z - T)\} \cup \\
&\{A_i, B_i,\ WA_i, RSA_i - ZA_i - TA_i,\ RSWA_i - ZWA_i - TWA_i\}_{i \in [n]} \cup \\
&\{A_iB_j\}_{i,j \in [n]} \cup \\
&\{\mathit{\Gamma}_k, \mathit{\Gamma}_k W, RS\mathit{\Gamma}_k - Z\mathit{\Gamma}_k - T\mathit{\Gamma}_k, RSW\mathit{\Gamma}_k - ZW\mathit{\Gamma}_k - TW\mathit{\Gamma}_k\}_{k \in [Q]} \cup \\
&\{\langle \boldsymbol{f}^{(k)}, \boldsymbol{A} \otimes \boldsymbol{B} \rangle + \mathit{\Gamma}_k W,\ \langle \boldsymbol{f}^{(k)}, W(\boldsymbol{A} \otimes \boldsymbol{B}) \rangle + \mathit{\Gamma}_k W^2\}_{k \in [Q]} \cup \\
&\{\langle \boldsymbol{f}^{(k)}, (RS - Z - T)(\boldsymbol{A} \otimes \boldsymbol{B}) \rangle + (RS - Z - T)W\mathit{\Gamma}_k\}_{k \in [Q]} \cup \\
&\{\langle \boldsymbol{f}^{(k)}, (RS - Z - T)W(\boldsymbol{A} \otimes \boldsymbol{B}) \rangle + (RS - Z - T)W^2\mathit{\Gamma}_k\}_{k \in [Q]} \cup \\
&\{\langle \boldsymbol{f}^{(k)}, B_j(\boldsymbol{A} \otimes \boldsymbol{B}) \rangle + B_j\mathit{\Gamma}_k W,\ B_j\mathit{\Gamma}_k\}_{j \in [n], k \in [Q]} \cup \\
&\{RA_i + x_i,\ TA_i + x_i \cdot S,\ SB_i + y_i,\ ZB_i + y_i \cdot R\}_{i \in [n]}, \cup \\
&\{RWA_i + x_i \cdot W,\ TWA_i + x_i \cdot SW\}_{i \in [n]}, \cup \\
&\{R^2SA_i - RZA_i - RTA_i + x_i \cdot (RS - Z - T)\}_{i \in [n]} \cup \\
&\{R^2SWA_i - RWZA_i - RTWA_i + x_i \cdot (RS - Z - T)W\}_{i \in [n]}, \cup \\
&\{RSTA_i - TZA_i - T^2A_i + x_i \cdot (RS^2 - SZ - ST)\}_{i \in [n]} \cup \\
&\{RSTWA_i - TWZA_i - T^2WA_i + x_i \cdot (RS^2 - SZ - ST)W\}_{i \in [n]}, \cup \\
&\{SA_iB_j + y_j \cdot A_i,\ ZA_iB_j + y_j \cdot RA_i,\ RA_iB_j + x_i \cdot B_j, TA_iB_j + x_i \cdot SB_j\}_{i,j \in [n]} \cup \\
&\{RSA_iB_j + x_iy_j + y_j \cdot RA_i + x_i \cdot SB_j\}_{i,j \in [n]} \cup \\
&\{RZA_iB_j + x_iy_j \cdot R + y_j \cdot R^2A_i + x_i \cdot ZB_j\}_{i,j \in [n]} \cup \\
&\{STA_iB_j + x_iy_j \cdot S + y_j \cdot TA_i + x_i \cdot S^2B_j\}_{i,j \in [n]} \cup \\
&\{TZA_iB_j + x_iy_j \cdot RS + y_j \cdot RTA_i + x_i \cdot SZB_j\}_{i,j \in [n]} \cup \\
&\{\langle \boldsymbol{f}^{(k)}, (SB_j + y_j)(\boldsymbol{A} \otimes \boldsymbol{B}) \rangle + SWB_j\mathit{\Gamma}_k + y_j \cdot W\mathit{\Gamma}_k\}_{j \in [n], k \in [Q]} \cup \\
&\{\langle \boldsymbol{f}^{(k)}, (ZB_j + y_j \cdot R)(\boldsymbol{A} \otimes \boldsymbol{B}) \rangle + ZWB_j\mathit{\Gamma}_k + y_j \cdot RW\mathit{\Gamma}_k\}_{j \in [n], k \in [Q]} \cup \\
&\{SB_j\mathit{\Gamma}_k + y_j \cdot \mathit{\Gamma}_k, ZB_j\mathit{\Gamma}_k + y_j \cdot R\mathit{\Gamma}_k\}_{j \in [n], k \in [Q]}
\end{aligned}
$$

The completion $C' = \mathcal{C}^{\mathcal{O},\mathcal{O}'_{ch}}(\boldsymbol{P})$ is the same as C except for replacing coefficients x_i with x'_i and y_j with y'_j, for all $i, j \in [n]$. In total, both completions consist of $r = |C| = |C'| = 4{+}15n{+}9n^2{+}8Q{+}6nQ$ polynomials in the ring $\mathbb{Z}_p[A_1, \ldots, A_n, B_1, \ldots, B_n, W, \mathit{\Gamma}_1, \ldots, \mathit{\Gamma}_Q, R, S, T, Z]$. Also it is possible to see by inspection that the maximal total degree of the polynomials in C and C' is $d = 5$ (this is the degree of the monomials R^2SWA_i).

Towards showing equality of the two kernels. Let us recall that the goal of the proof is to show that

$$K = \{\boldsymbol{k} \in \mathbb{Z}_p^r \mid \boldsymbol{k} \cdot C = 0\} \ = \ \{\boldsymbol{k} \in \mathbb{Z}_p^r \mid \boldsymbol{k} \cdot C' = 0\} = K' \tag{2}$$

under the condition that $\langle \boldsymbol{f}^{(k)}, \boldsymbol{x} \otimes \boldsymbol{y} \rangle = \langle \boldsymbol{f}^{(k)}, \boldsymbol{x'} \otimes \boldsymbol{y'} \rangle$ for all $k = 1$ to Q. One way to show this equality is to compute bases for both kernels K and K', and show that these bases generate the same space (or that they are actually the same). This is what we eventually do. However, instead of proceeding straight to computing bases for the two kernels, we first show that showing the equality in (2) is equivalent to showing a similar equality for a much simpler (smaller) vector space.

Lemma 11. *Let C and C' be the parametric completions computed above. There exist two sets of polynomials $\widetilde{C} \subset C$ and $\widetilde{C}' \subset C'$, both of cardinality $\widetilde{r}$, such that if*

$$\widetilde{K} = \{\boldsymbol{k} \in \mathbb{Z}_p^{\widetilde{r}} \mid \boldsymbol{k} \cdot \widetilde{C} = 0\} \;=\; \{\boldsymbol{k} \in \mathbb{Z}_p^{\widetilde{r}} \mid \boldsymbol{k} \cdot \widetilde{C}' = 0\} = \widetilde{K}' \tag{3}$$

is satisfied then equation (2) *is satisfied as well.*

Proof. As a first step, we show the existence of a set of indices $\mathcal{S} \subseteq [r]$ and a corresponding vector subspace $U = \{\boldsymbol{k} \in \mathbb{Z}_p^r : k_i = 0, \forall i \in \mathcal{S}\} \subset \mathbb{Z}_p^r$ such that both K and K' are contained in U, i.e., $K \subset U$ and $K' \subset U$. This fact implies that the equality of equation (2) is the same as

$$K = \{\boldsymbol{k} \in U \mid \boldsymbol{k} \cdot C = 0\} \;=\; \{\boldsymbol{k} \in U \mid \boldsymbol{k} \cdot C' = 0\} = K' \tag{4}$$

We show the existence of this set $\mathcal{S}$ by observing the specific shapes of the polynomials in C and C'. $\mathcal{S}$ is the set of indices $i \in [r]$ such that the i-th polynomial in both C and C' contains a unique monomial, i.e., a monomial which appears *only* in that polynomial. For every polynomial p_i (resp. p'_i) such that $i \in \mathcal{S}$ it holds that any vector $\boldsymbol{k} \in K$ (resp. K') must have $k_i = 0$.

By inspection, the set of such unique monomials (which implicitly determines $\mathcal{S}$) is

$$\begin{aligned}
&\{WA_i, RSA_i, ZA_i, RSTA_i, RSTWA_i, RSWA_i, ZWA_i, R^2SA_i, RZA_i\}_{i\in[n]} \cup \\
&\{R^2SWA_i, RZWA_i, RTWA_i, RWA_i, TZA_i, T^2A_i, TZWA_i, T^2WA_i\}_{i\in[n]} \cup \\
&\{STA_iB_j, TZA_iB_j, RZA_iB_j, SA_iB_j\}_{i,j\in[n]} \cup \\
&\{RS\Gamma_k, Z\Gamma_k, T\Gamma_k, W^2\Gamma_k, RSW^2\Gamma_k, ZW^2\Gamma_k, TW^2\Gamma_k\}_{k\in[Q]} \cup \\
&\{B_j\Gamma_k, SB_j\Gamma_k, WB_j\Gamma_k, ZB_j\Gamma_k, SWB_j\Gamma_k, ZWB_j\Gamma_k\}_{j\in[n],k\in[Q]}
\end{aligned}$$

Then we define $\widetilde{C}$ (resp. $\widetilde{C}'$) as the subset of C (resp. C') including all those polynomials whose index i is not in $\mathcal{S}$, i.e., $\widetilde{C} = \{p_i \in C \mid i \notin \mathcal{S}\}$ and $\widetilde{C}' = \{p_i \in C' \mid i \notin \mathcal{S}\}$. Let $\widetilde{r} = |\widetilde{C}| = |\widetilde{C}'|$.

By the definitions of U, $\widetilde{C}$ and $\widetilde{C}'$ given above, it is easy to see that if the following equality

$$\widetilde{K} = \{\boldsymbol{k} \in \mathbb{Z}_p^{\widetilde{r}} \mid \boldsymbol{k} \cdot \widetilde{C} = 0\} \;=\; \{\boldsymbol{k} \in \mathbb{Z}_p^{\widetilde{r}} \mid \boldsymbol{k} \cdot \widetilde{C}' = 0\} = \widetilde{K}'$$

is satisfied, so is the equality of equation (4), and thus that of equation (2). This completes the proof of the lemma.

For convenience, we show explicitly the simplified completion $\widetilde{C}$:

$$\begin{aligned}
\widetilde{C} = {} &\{1,\ W,\ RS - Z - T,\ W(RS - Z - T)\} \cup \\
&\{A_i, B_i\}_{i\in[n]} \cup \\
&\{A_iB_j\}_{i,j\in[n]} \cup \\
&\{\Gamma_k, \Gamma_k W, RSW\Gamma_k - ZW\Gamma_k - TW\Gamma_k\}_{k\in[Q]} \cup \\
&\{\langle \boldsymbol{f}^{(k)}, \boldsymbol{A} \otimes \boldsymbol{B}\rangle + \Gamma_k W\}_{k\in[Q]} \cup \\
&\{\langle \boldsymbol{f}^{(k)}, (RS - Z - T)(\boldsymbol{A} \otimes \boldsymbol{B})\rangle + (RS - Z - T)W\Gamma_k\}_{k\in[Q]} \cup \\
&\{RA_i + x_i,\ TA_i + x_i \cdot S,\ SB_i + y_i,\ ZB_i + y_i \cdot R\}_{i\in[n]}, \cup \\
&\{TWA_i + x_i \cdot SW\}_{i\in[n]}, \cup \\
&\{RA_iB_j + x_i \cdot B_j, ZA_iB_j + y_j \cdot RA_i,\ TA_iB_j + x_i \cdot SB_j\}_{i,j\in[n]} \cup \\
&\{RSA_iB_j + x_iy_j + y_j \cdot RA_i + x_i \cdot SB_j\}_{i,j\in[n]}
\end{aligned}$$

$\widetilde{C}'$ is the same except for having coefficients x_i', y_i' instead of x_i, y_i. $\square$

By using the result of Lemma 11, we are left with showing the equality of equation (3). To this end, we apply below an analogous simplification.

Lemma 12. *Let $\widetilde{C}$ and $\widetilde{C}'$ be the sets of polynomials as defined in Lemma 11. There exist two sets of polynomials $\widehat{C} \subset \widetilde{C}$ and $\widehat{C}' \subset \widetilde{C}'$, both of cardinality N, such that if*

$$\widehat{K} = \{\boldsymbol{k} \in \mathbb{Z}_p^N \mid \boldsymbol{k} \cdot \widehat{C} = 0\} \;=\; \{\boldsymbol{k} \in \mathbb{Z}_p^N \mid \boldsymbol{k} \cdot \widehat{C}' = 0\} = \widehat{K}' \tag{5}$$

is satisfied then equation (3) *is satisfied as well.*

Proof. The proof of this Lemma is quite similar to that of Lemma 11. As a first step, we show the existence of a set of indices $\widetilde{\mathcal{S}} \subseteq [\widetilde{r}]$ and a corresponding vector subspace $\widetilde{U} = \{\boldsymbol{k} \in \mathbb{Z}_p^{\widetilde{r}} : k_i = 0, \forall i \in \widetilde{\mathcal{S}}\} \subset \mathbb{Z}_p^{\widetilde{r}}$ such that both $\widetilde{K}$ and $\widetilde{K}'$ are contained in $\widetilde{U}$, i.e., $\widetilde{K} \subset \widetilde{U}$ and $\widetilde{K}' \subset \widetilde{U}$. This fact implies that the equality of equation (3) is the same as

$$\widetilde{K} = \{\boldsymbol{k} \in \widetilde{U} \mid \boldsymbol{k} \cdot \widetilde{C} = 0\} \;=\; \{\boldsymbol{k} \in \widetilde{U} \mid \boldsymbol{k} \cdot \widetilde{C}' = 0\} = \widetilde{K}' \tag{6}$$

To see the existence of this set $\widetilde{\mathcal{S}}$ we again look at the specific shapes of the polynomials in $\widetilde{C}$ and $\widetilde{C}'$. $\widetilde{S}$ is the set of indices $i \in [\widetilde{r}]$ such that the i-th polynomial in both $\widetilde{C}$ and $\widetilde{C}'$ contains a unique monomial, i.e., a monomial which appears only in that polynomial. For every such polynomial p_i (resp. p_i') it holds that any vector $\boldsymbol{k} \in \widetilde{K}$ (resp. $\widetilde{K}'$) must have the corresponding i-th coefficient $k_i = 0$.

By inspection, the set of such unique monomials is

$$\{W, RS, Z, T, RSW, WZ, TW\} \cup \{\Gamma_k\}_{k\in[Q]} \cup \{A_i, B_i, TA_i, TWA_i, ZB_i\}_{i\in[n]} \cup \{RA_iB_j\}_{i,j\in[n]}$$

Similarly to the previous lemma, we define $\widehat{C}$ (resp. $\widehat{C}'$) as the subset of $\widetilde{C}$ (resp. $\widetilde{C}'$) including all those polynomials whose index i is not in $\widetilde{\mathcal{S}}$, i.e., $\widehat{C} = \{p_i \in \widetilde{C} \mid i \notin \widetilde{\mathcal{S}}\}, \quad \widehat{C}' = \{p_i \in \widetilde{C}' \mid i \notin \widetilde{\mathcal{S}}\}$. Let $N = |\widehat{C}| = |\widehat{C}'|$.

By the definitions of $\widetilde{U}$, $\widehat{C}$ and $\widehat{C}'$, it is easy to see that if the following equality

$$\widehat{K} = \{\boldsymbol{k} \in \mathbb{Z}_p^N \mid \boldsymbol{k} \cdot \widehat{C} = 0\} \;=\; \{\boldsymbol{k} \in \mathbb{Z}_p^N \mid \boldsymbol{k} \cdot \widehat{C}' = 0\} = \widehat{K}'$$

is satisfied, so is the equality of equation (6), and thus that of equation (3). This completes the proof of the lemma.

For convenience, we show the simplified completion $\widehat{C}$:

$$\begin{aligned}
\widehat{C}_0 &= \{1\}\\
\widehat{C}_{1,i} &= \{RA_i + x_i\}_{i\in[n]},\\
\widehat{C}_{2,i} &= \{SB_i + y_i\}_{i\in[n]}\\
\widehat{C}_{3,i,j} &= \{A_iB_j\}_{i,j\in[n]}\\
\widehat{C}_{4,i,j} &= \{RSA_iB_j + x_iy_j + y_j \cdot RA_i + x_i \cdot SB_j\}_{i,j\in[n]}\\
\widehat{C}_{5,i,j} &= \{TA_iB_j + x_i \cdot SB_j\}_{i,j\in[n]}\\
\widehat{C}_{6,i,j} &= \{ZA_iB_j + y_j \cdot RA_i\}_{i,j\in[n]}\\
\widehat{C}_{7,k} &= \{W\varGamma_k\}_{k\in[Q]}\\
\widehat{C}_{8,k} &= \{RSW\varGamma_k - ZW\varGamma_k - TW\varGamma_k\}_{k\in[Q]}\\
\widehat{C}_{9,k} &= \{\langle \boldsymbol{f}^{(k)}, \boldsymbol{A} \otimes \boldsymbol{B}\rangle + W\varGamma_k\}_{k\in[Q]}\\
\widehat{C}_{10,k} &= \{\langle \boldsymbol{f}^{(k)}, (RS - Z - T)(\boldsymbol{A} \otimes \boldsymbol{B})\rangle + (RS - Z - T)W\varGamma_k\}_{k\in[Q]}
\end{aligned}$$

$\widehat{C}'$ is defined analogously, except for having values x'_i and y'_i instead of x_i and y_i respectively. □

By using the result of Lemma 12, we are left with showing the equality of equation (5) that we recall below

$$\widehat{K} = \{\boldsymbol{k} \in \mathbb{Z}_p^N \mid \boldsymbol{k} \cdot \widehat{C} = 0\} \;=\; \{\boldsymbol{k} \in \mathbb{Z}_p^N \mid \boldsymbol{k} \cdot \widehat{C}' = 0\} = \widehat{K}'$$

All the polynomials in the completions $\widehat{C}$ and $\widehat{C}'$ can be seen as linear combinations of the following set of monomials, that we call the monomials basis

$$\begin{array}{cccc}
H_0 = 1 & \{H_{1,i} = RA_i\}_{i\in[n]} & \{H_{2,i} = SB_i\}_{i\in[n]} & \\
\{H_{3,i,j} = A_iB_j\}_{i,j\in[n]} & \{H_{4,i,j} = RSA_iB_j\}_{i,j\in[n]} & \{H_{5,i,j} = TA_iB_j\}_{i,j\in[n]} & \{H_{6,i,j} = ZA_iB_j\}_{i,j\in[n]}\\
\{H_{7,k} = W\varGamma_k\}_{k\in[Q]} & \{H_{8,k} = RSW\varGamma_k\}_{k\in[Q]} & \{H_{9,k} = TW\varGamma_k\}_{k\in[Q]} & \{H_{10,k} = ZW\varGamma_k\}_{k\in[Q]}
\end{array}$$

Let us write the above monomials basis as a vector $\boldsymbol{H}$ of N entries. Then, $\boldsymbol{H}$ is a monomial basis in the sense that for every polynomial $p \in \widehat{C}$ (resp. $p' \in \widehat{C}'$) there exists a vector $\boldsymbol{v} \in \mathbb{Z}_p^N$ (resp. $\boldsymbol{v}'$) such that $p = \langle \boldsymbol{v}, \boldsymbol{H}\rangle$ (resp. $p' = \langle \boldsymbol{v}', \boldsymbol{H}\rangle$). (Precisely, $\boldsymbol{v}$ has coefficients in $\{0,1\} \cup \{x_i, y_i\}_{i\in[n]} \cup \{x_iy_j\}_{i,j\in[n]} \cup \{f_{i,j}^{(k)}\}_{i,j\in[n],k\in[Q]}$ while $\boldsymbol{v}'$ has coefficients in $\{0,1\} \cup \{x'_i, y'_i\}_{i\in[n]} \cup \{x'_iy'_j\}_{i,j\in[n]} \cup \{f_{i,j}^{(k)}\}_{i,j\in[n],k\in[Q]}$.)

Let $\mathbf{M} \in \mathbb{Z}_p^{N\times N}$ be the matrix obtained by concatenating, row after row, all these vectors $\boldsymbol{v}_1, \ldots \boldsymbol{v}_N$, i.e., such that all polynomials in the completion can be compactly expressed as $\widehat{C} = \mathbf{M} \cdot \boldsymbol{H}$. And let us define analogously $\mathbf{M}'$ such that $\widehat{C}' = \mathbf{M}' \cdot \boldsymbol{H}$.

Using this representation in the monomial basis, then showing the equality in (5) is the same as showing

$$\{\boldsymbol{k} \in \mathbb{Z}_p^N : \boldsymbol{k}^\top \cdot \mathbf{M} = \mathbf{0}\} = \{\boldsymbol{k} \in \mathbb{Z}_p^N : \boldsymbol{k}^\top \cdot \mathbf{M}' = \mathbf{0}\}$$

namely that $\mathbf{M}$ and $\mathbf{M}'$ have the same left kernel.

We finalize the proof of Theorem 5 by proving the following lemma.

Lemma 13. *Let* $\mathbf{M}$ *and* $\mathbf{M}'$ *be the matrices defined above. Then* $\ker(\mathbf{M}^\top) = \ker(\mathbf{M}'^\top)$*.*

Proof. We prove the lemma by computing bases for the kernels of both transposed matrices $\mathbf{M}^\top$ and $\mathbf{M}'^\top$. Below we write the matrix $\mathbf{M}^\top$ using a "block representation" that we explain slightly below:

$$\mathbf{M}^\top = \begin{array}{lr||ccc|cccc|cccc}
 & & \widehat{C}_0 & \widehat{\mathbf{C}}_1 & \widehat{\mathbf{C}}_2 & \widehat{\mathbf{C}}_3 & \widehat{\mathbf{C}}_4 & \widehat{\mathbf{C}}_5 & \widehat{\mathbf{C}}_6 & \widehat{\mathbf{C}}_7 & \widehat{\mathbf{C}}_8 & \widehat{\mathbf{C}}_9 & \widehat{\mathbf{C}}_{10} \\
\hline\hline
H_0 & 1 & 1 & \boldsymbol{x} & \boldsymbol{y} & \mathbf{0} & \boldsymbol{x} \otimes \boldsymbol{y} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_1 & R\boldsymbol{A} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{I} \otimes \boldsymbol{y} & \mathbf{0} & \mathbf{I} \otimes \boldsymbol{y} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_2 & S\boldsymbol{B} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \boldsymbol{x} \otimes \mathbf{I} & \boldsymbol{x} \otimes \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\hline
\mathbf{H}_3 & \boldsymbol{A} \otimes \boldsymbol{B} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F} & \mathbf{0} \\
\mathbf{H}_4 & RS(\boldsymbol{A} \otimes \boldsymbol{B}) & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F} \\
\mathbf{H}_5 & T(\boldsymbol{A} \otimes \boldsymbol{B}) & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{F} \\
\mathbf{H}_6 & Z(\boldsymbol{A} \otimes \boldsymbol{B}) & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{F} \\
\hline
\mathbf{H}_7 & W\boldsymbol{\Gamma} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\
\mathbf{H}_8 & RSW\boldsymbol{\Gamma} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{I} \\
\mathbf{H}_9 & TW\boldsymbol{\Gamma} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & -\mathbf{I} \\
\mathbf{H}_{10} & ZW\boldsymbol{\Gamma} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & -\mathbf{I}
\end{array}$$

Above, the elements on the left and above the double rules || are intended as labels for the rows and columns of the matrix.

In our "block representation" we have that:

- $\widehat{C}_0$ is a single column, $\widehat{\mathbf{C}}_1, \widehat{\mathbf{C}}_2$ consist of n columns each, $\widehat{\mathbf{C}}_3, \ldots, \widehat{\mathbf{C}}_6$ have n^2 columns each, and $\widehat{\mathbf{C}}_7, \ldots, \widehat{\mathbf{C}}_{10}$ have Q columns each.
- Similarly to above, H_0 is a single row, $\mathbf{H}_1, \mathbf{H}_2$ consist of n rows each, $\mathbf{H}_3, \ldots, \mathbf{H}_6$ have n^2 rows each, and $\mathbf{H}_7, \ldots, \mathbf{H}_{10}$ have Q rows each.
- $\boldsymbol{x}, \boldsymbol{y}$ are n-dimensional row vectors.
- $\mathbf{I}$ is the identity matrix of dimension $n \times n$, or $n^2 \times n^2$ or $Q \times Q$.
- $\mathbf{0}$ denotes a vector or a matrix of zeros whose dimension is easily extrapolated from its position.
- $\mathbf{F}$ is the $(n^2 \times Q)$-dimensional matrix $\mathbf{F} = \left[\boldsymbol{f}^{(1)} \mid \cdots \mid \boldsymbol{f}^{(Q)}\right]$, which essentially represents a concatenation, column after column, of all the queried functions, each represented as a column vector.
- Tensoring notation: For any vectors $\boldsymbol{x}, \boldsymbol{y}$ of dimension n, we denote by $\boldsymbol{x} \otimes \boldsymbol{y}$ their tensor product that we write as an n^2-dimensional *row vector* $(x_i y_j)_{i,j}$ where the entries i, j are ordered lexicographically, e.g., $\boldsymbol{x} \otimes \boldsymbol{y} := (x_1 y_1, x_1 y_2, \ldots, x_n y_{n-1}, x_n y_n)$. Clearly, $\otimes$ *is not commutative.*

  Moreover, abusing notation, we define the tensor product between an $(\ell \times n)$-dimensional matrix $\mathbf{A}$ and an n-dimensional vector $\boldsymbol{y}$ as the component-wise tensor product of every row of $\mathbf{A}$ with $\boldsymbol{y}$, i.e., letting $\mathbf{A}_i$ be the i-th row of $\mathbf{A}$, we define

$$\mathbf{A} \otimes \boldsymbol{y} := \begin{bmatrix} \mathbf{A}_1 \otimes \boldsymbol{y} \\ \vdots \\ \mathbf{A}_\ell \otimes \boldsymbol{y} \end{bmatrix} \quad \text{and similarly} \quad \boldsymbol{y} \otimes \mathbf{A} := \begin{bmatrix} \boldsymbol{y} \otimes \mathbf{A}_1 \\ \vdots \\ \boldsymbol{y} \otimes \mathbf{A}_\ell \end{bmatrix}$$

As an example, using the just introduced notation, one can take a block-column such as $\widehat{\mathbf{C}}_1 \in \mathbb{Z}_p^{N\times n}$ in $\mathbf{M}^\top$, and compactly write

$$\widehat{\mathbf{C}}_1 \otimes \boldsymbol{y} = \begin{bmatrix} \boldsymbol{x} \otimes \boldsymbol{y} \\ \mathbf{I} \otimes \boldsymbol{y} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix} \in \mathbb{Z}_p^{N\times n^2}$$

This block representation of $\mathbf{M}^\top$ is convenient as it allows us to perform gaussian elimination on $\mathbf{M}^\top$ by expressing multiple column operations with single block-of-columns operations (i.e., intuitively treating every block as if being of constant size). Namely, we will express operations using blocks and observe that these get easily translated into corresponding column operations as follows:

- swap of column-blocks is translated into component-wise swapping of columns,
- addition/subtraction of two column-blocks becomes a component-wise addition/subtraction of the corresponding columns,
- tensoring of a column-block by a vector is translated into (simultaneously) multiplying several columns by field constants.

Now we proceed to computing a basis for the kernel of $\mathbf{M}^\top$. To this end, we first extend below $\mathbf{M}^\top$ with the identity matrix. This gives us the following matrix $\mathbf{T}_1$:

$$\mathbf{T}_1 \quad = \quad \begin{array}{r||ccc|cccc|cccc}
 & \widehat{C}_0 & \widehat{\mathbf{C}}_1 & \widehat{\mathbf{C}}_2 & \widehat{\mathbf{C}}_3 & \widehat{\mathbf{C}}_4 & \widehat{\mathbf{C}}_5 & \widehat{\mathbf{C}}_6 & \widehat{\mathbf{C}}_7 & \widehat{\mathbf{C}}_8 & \widehat{\mathbf{C}}_9 & \widehat{\mathbf{C}}_{10} \\
\hline
H_0 & 1 & \boldsymbol{x} & \boldsymbol{y} & \mathbf{0} & \boldsymbol{x}\otimes\boldsymbol{y} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_1 & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{I}\otimes\boldsymbol{y} & \mathbf{0} & \mathbf{I}\otimes\boldsymbol{y} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_2 & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \boldsymbol{x}\otimes\mathbf{I} & \boldsymbol{x}\otimes\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\hline
\mathbf{H}_3 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F} & \mathbf{0} \\
\mathbf{H}_4 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F} \\
\mathbf{H}_5 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{F} \\
\mathbf{H}_6 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{F} \\
\hline
\mathbf{H}_7 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\
\mathbf{H}_8 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{I} \\
\mathbf{H}_9 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & -\mathbf{I} \\
\mathbf{H}_{10} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & -\mathbf{I} \\
\hline
 & 1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}
\end{array}$$

In what follows we perform gaussian elimination on the above matrix via a series of column transformations until the upper matrix gets in column echelon form. In order, we apply the following column transformations (expressed in block notation):

1. $\widehat{\mathbf{C}}_1 = \widehat{\mathbf{C}}_1 - \boldsymbol{x} \otimes \widehat{C}_0$ and $\widehat{\mathbf{C}}_2 = \widehat{\mathbf{C}}_2 - \boldsymbol{y} \otimes \widehat{C}_0$; this yields matrix $\mathbf{T}_2$.
2. $\widehat{\mathbf{C}}_4 = \widehat{\mathbf{C}}_4 - (\boldsymbol{x} \otimes \boldsymbol{y}) \otimes \widehat{C}_0 - \widehat{\mathbf{C}}_5 - \widehat{\mathbf{C}}_6$; this yields matrix $\mathbf{T}_3$.
3. $\widehat{\mathbf{C}}_5 = \widehat{\mathbf{C}}_5 - \boldsymbol{x} \otimes \widehat{\mathbf{C}}_2$ and $\widehat{\mathbf{C}}_6 = \widehat{\mathbf{C}}_6 - \widehat{\mathbf{C}}_1 \otimes \boldsymbol{y}$; this yields matrix $\mathbf{T}_4$.
4. $\widehat{\mathbf{C}}_9 = \widehat{\mathbf{C}}_9 - \widehat{\mathbf{C}}_3 \cdot \mathbf{F} - \widehat{\mathbf{C}}_7$ and $\widehat{\mathbf{C}}_{10} = \widehat{\mathbf{C}}_{10} - \widehat{\mathbf{C}}_4 \cdot \mathbf{F}$; this yields matrix $\mathbf{T}_5$.
5. $\widehat{\mathbf{C}}_{10} = \widehat{\mathbf{C}}_{10} - \widehat{\mathbf{C}}_8$ and $\widehat{\mathbf{C}}_4 = \widehat{\mathbf{C}}_4 + \widehat{\mathbf{C}}_5 + \widehat{\mathbf{C}}_6$; this yields matrix $\mathbf{T}_6$.

The matrices $\mathbf{T}_1$–$\mathbf{T}_6$ appear in the following.

$$\mathbf{T}_2 \quad = \quad \begin{array}{r||ccc|cccc|cccc}
 & \widehat{C}_0 & \widehat{\mathbf{C}}_1 & \widehat{\mathbf{C}}_2 & \widehat{\mathbf{C}}_3 & \widehat{\mathbf{C}}_4 & \widehat{\mathbf{C}}_5 & \widehat{\mathbf{C}}_6 & \widehat{\mathbf{C}}_7 & \widehat{\mathbf{C}}_8 & \widehat{\mathbf{C}}_9 & \widehat{\mathbf{C}}_{10} \\
\hline\hline
H_0 & 1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{x}\otimes\boldsymbol{y} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_1 & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{I}\otimes\boldsymbol{y} & \mathbf{0} & \mathbf{I}\otimes\boldsymbol{y} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_2 & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \boldsymbol{x}\otimes\mathbf{I} & \boldsymbol{x}\otimes\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\hline
\mathbf{H}_3 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F} & \mathbf{0} \\
\mathbf{H}_4 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F} \\
\mathbf{H}_5 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{F} \\
\mathbf{H}_6 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{F} \\
\hline
\mathbf{H}_7 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\
\mathbf{H}_8 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{I} \\
\mathbf{H}_9 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & -\mathbf{I} \\
\mathbf{H}_{10} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & -\mathbf{I} \\
\hline
 & 1 & -\boldsymbol{x} & -\boldsymbol{y} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}
\end{array}$$

$$
\mathbf{T}_3 \quad = \quad
\begin{array}{r||ccc|cccc|cccc}
 & \widehat{C}_0 & \widehat{\mathbf{C}}_1 & \widehat{\mathbf{C}}_2 & \widehat{\mathbf{C}}_3 & \widehat{\mathbf{C}}_4 & \widehat{\mathbf{C}}_5 & \widehat{\mathbf{C}}_6 & \widehat{\mathbf{C}}_7 & \widehat{\mathbf{C}}_8 & \widehat{\mathbf{C}}_9 & \widehat{\mathbf{C}}_{10} \\
\hline\hline
H_0 & 1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_1 & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}\otimes\boldsymbol{y} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_2 & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \boldsymbol{x}\otimes\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\hline
\mathbf{H}_3 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F} & \mathbf{0} \\
\mathbf{H}_4 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F} \\
\mathbf{H}_5 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{F} \\
\mathbf{H}_6 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{F} \\
\hline
\mathbf{H}_7 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\
\mathbf{H}_8 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{I} \\
\mathbf{H}_9 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & -\mathbf{I} \\
\mathbf{H}_{10} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & -\mathbf{I} \\
\hline
 & 1 & -\boldsymbol{x} & -\boldsymbol{y} & \mathbf{0} & -\boldsymbol{x}\otimes\boldsymbol{y} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}
\end{array}
$$

$$\mathbf{T}_4 \quad = \quad \begin{array}{r||ccc|cccc|cccc}
 & \widehat{C}_0 & \widehat{\mathbf{C}}_1 & \widehat{\mathbf{C}}_2 & \widehat{\mathbf{C}}_3 & \widehat{\mathbf{C}}_4 & \widehat{\mathbf{C}}_5 & \widehat{\mathbf{C}}_6 & \widehat{\mathbf{C}}_7 & \widehat{\mathbf{C}}_8 & \widehat{\mathbf{C}}_9 & \widehat{\mathbf{C}}_{10} \\
\hline\hline
H_0 & 1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_1 & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_2 & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\hline
\mathbf{H}_3 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F} & \mathbf{0} \\
\mathbf{H}_4 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F} \\
\mathbf{H}_5 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{F} \\
\mathbf{H}_6 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{F} \\
\hline
\mathbf{H}_7 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\
\mathbf{H}_8 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{I} \\
\mathbf{H}_9 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & -\mathbf{I} \\
\mathbf{H}_{10} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & -\mathbf{I} \\
\hline
 & 1 & -\boldsymbol{x} & -\boldsymbol{y} & \mathbf{0} & -\boldsymbol{x}\otimes\boldsymbol{y} & \boldsymbol{x}\otimes\boldsymbol{y} & \boldsymbol{x}\otimes\boldsymbol{y} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I}\otimes\boldsymbol{y} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & -\boldsymbol{x}\otimes\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}
\end{array}$$

$$\mathbf{T}_5 \quad = \quad \begin{array}{r||ccc|cccc|cccc}
 & \widehat{C}_0 & \widehat{\mathbf{C}}_1 & \widehat{\mathbf{C}}_2 & \widehat{\mathbf{C}}_3 & \widehat{\mathbf{C}}_4 & \widehat{\mathbf{C}}_5 & \widehat{\mathbf{C}}_6 & \widehat{\mathbf{C}}_7 & \widehat{\mathbf{C}}_8 & \widehat{\mathbf{C}}_9 & \widehat{\mathbf{C}}_{10} \\
\hline\hline
H_0 & 1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_1 & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_2 & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\hline
\mathbf{H}_3 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_4 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_5 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_6 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\hline
\mathbf{H}_7 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_8 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{I} \\
\mathbf{H}_9 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & -\mathbf{I} \\
\mathbf{H}_{10} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & -\mathbf{I} \\
\hline
 & 1 & -\boldsymbol{x} & -\boldsymbol{y} & \mathbf{0} & -\boldsymbol{x}\otimes\boldsymbol{y} & \boldsymbol{x}\otimes\boldsymbol{y} & \boldsymbol{x}\otimes\boldsymbol{y} & \mathbf{0} & \mathbf{0} & \mathbf{0} & (\boldsymbol{x}\otimes\boldsymbol{y})\mathbf{F} \\
 & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I}\otimes\boldsymbol{y} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & -\boldsymbol{x}\otimes\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{F} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{F} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & -\mathbf{I} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}
\end{array}$$

$$\mathbf{T}_6 \quad = \quad \begin{array}{c||ccccccccc||c||c}
 & \widehat{C}_0 & \widehat{\mathbf{C}}_1 & \widehat{\mathbf{C}}_2 & \widehat{\mathbf{C}}_3 & \widehat{\mathbf{C}}_4 & \widehat{\mathbf{C}}_5 & \widehat{\mathbf{C}}_6 & \widehat{\mathbf{C}}_7 & \widehat{\mathbf{C}}_8 & \widehat{\mathbf{C}}_9 & \widehat{\mathbf{C}}_{10} \\
\hline\hline
H_0 & 1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_1 & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_2 & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_3 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_4 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_5 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_6 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_7 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_8 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_9 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{0} \\
\mathbf{H}_{10} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{0} \\
\hline
 & 1 & -\boldsymbol{x} & -\boldsymbol{y} & \mathbf{0} & \boldsymbol{x} \otimes \boldsymbol{y} & \boldsymbol{x} \otimes \boldsymbol{y} & \boldsymbol{x} \otimes \boldsymbol{y} & \mathbf{0} & \mathbf{0} & \mathbf{0} & (\boldsymbol{x} \otimes \boldsymbol{y})\mathbf{F} \\
 & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & -\mathbf{I} \otimes \boldsymbol{y} & \mathbf{0} & -\mathbf{I} \otimes \boldsymbol{y} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & -\boldsymbol{x} \otimes \mathbf{I} & -\boldsymbol{x} \otimes \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{F} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{F} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & -\mathbf{I} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & -\mathbf{I} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\
 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}
\end{array}$$

As one can see, in the above matrix $\mathbf{T}_6$ the upper part is in column echelon form. Hence, the basis $\mathcal{K}$ of the kernel of $\mathbf{M}^\top$ is represented by the two rightmost block-columns of the lower matrix. These columns are a collection of $2Q$ N-dimensional vectors as follows

$$\mathcal{K} = \left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ -\mathbf{F} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ -\mathbf{I} \\ \mathbf{0} \\ \mathbf{I} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} (\boldsymbol{x} \otimes \boldsymbol{y})\mathbf{F} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ -\mathbf{F} \\ \mathbf{F} \\ \mathbf{F} \\ \mathbf{0} \\ -\mathbf{I} \\ \mathbf{0} \\ \mathbf{I} \end{bmatrix} \right\} \in \mathbb{Z}_p^{N \times 2Q}$$

It is easy to see that when applying the analogous set of transformations on $\mathbf{M'}^\top$ (where $\boldsymbol{x}$ and $\boldsymbol{y}$ are replaced by $\boldsymbol{x}'$ and $\boldsymbol{y}'$ respectively) one obtains the *same* basis $\mathcal{K}$. Precisely, the analogous transformations lead to the same vectors of the kernel except for having $(\boldsymbol{x}' \otimes \boldsymbol{y}')\mathbf{F}$ instead of $(\boldsymbol{x} \otimes \boldsymbol{y})\mathbf{F}$. However, by the legitimacy condition of the security game it holds $(\boldsymbol{x} \otimes \boldsymbol{y})\mathbf{F} = (\boldsymbol{x}' \otimes \boldsymbol{y}')\mathbf{F}$. Hence, $\mathbf{M}$ and $\mathbf{M}'$ have the same basis for their left kernels, which completes the proof. □

6 Predicate Encryption for Bilinear Maps Evaluation

Here we show how to use our functional encryption schemes to build a Predicate Encryption (PE) scheme for the evaluation of bilinear maps over attributes. Specifically, we give a scheme for the predicate $\mathsf{P} : \mathcal{X} \times \mathcal{Y} \to \{0, 1\}$ where $\mathcal{X} \subset \mathbb{Z}_p^n \times \mathbb{Z}_p^m$, $\mathcal{Y} \subset \mathbb{Z}_p^{n \times m}$, and for all $(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{X}$ and $\mathbf{F} \in \mathcal{Y}$:

$$\boldsymbol{x}^\top \mathbf{F} \boldsymbol{y} \in \{0, 1\} \text{ and } \mathsf{P}((\boldsymbol{x}, \boldsymbol{y}), \mathbf{F}) = 1 \text{ iff } \boldsymbol{x}^\top \mathbf{F} \boldsymbol{y} = 1.$$

In Figure 11, we present a generic construction of PE for P from any functional encryption scheme FE for the bilinear maps functionality $F : \mathcal{K} \times \mathcal{M}' \to \mathcal{Y}'$, where $\mathcal{M}' := \mathbb{Z}_p^n \times \mathbb{Z}_p^m$, $\mathcal{K} := \mathbb{Z}_p^{n \times m}$, $\mathcal{Y}' := \mathbb{G}_T$ and for all $(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{M}'$, $\mathbf{F} \in \mathcal{K}$

$$F(\mathbf{F}, (\boldsymbol{x}, \boldsymbol{y})) = [\boldsymbol{x}^\top \mathbf{F} \boldsymbol{y}]_T \,.$$

The PE scheme can be instantiated by using one of our FE constructions presented in Sections 3 and 4.

```
$\mathsf{Setup}(1^\lambda, \mathsf{P}, \mathcal{M} := \mathbb{G}_T)$:
Return $(\mathsf{mpk}, \mathsf{msk}) \leftarrow_{\text{R}} \mathsf{Setup}_{\mathsf{FE}}(1^\lambda, F)$

$\mathsf{KeyGen}(\mathsf{msk}, \mathbf{F} \in \mathcal{Y})$:
Return $\mathsf{sk}_{\mathbf{F}} := \mathsf{KeyGen}_{\mathsf{FE}}(\mathsf{msk}, \mathbf{F})$
```

```
$\mathsf{Encrypt}(\mathsf{mpk}, (\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{X}, M \in \mathbb{G}_T)$:
$w \leftarrow_{\text{R}} \mathbb{Z}_p$; $C_0 := [w]_T + M$
$C_1 := \mathsf{Encrypt}_{\mathsf{FE}}(\mathsf{mpk}, (w \cdot \boldsymbol{x}, \boldsymbol{y}))$
Return $\mathsf{Ct}_{(\boldsymbol{x}, \boldsymbol{y})} := (C_0, C_1)$

$\mathsf{Decrypt}(\mathsf{mpk}, \mathsf{Ct}_{(\boldsymbol{x}, \boldsymbol{y})} := (C_0, C_1), \mathsf{sk}_{\mathbf{F}})$:
$K := \mathsf{Decrypt}_{\mathsf{FE}}(\mathsf{mpk}, C_1, \mathsf{sk}_{\mathbf{F}})$
Return $C_0 - K$.
```

Fig. 11. PE, a predicate encryption scheme, selectively (resp. adaptively) secure if the underlying FE scheme $(\mathsf{Setup}_{\mathsf{FE}}, \mathsf{KeyGen}_{\mathsf{FE}}, \mathsf{Encrypt}_{\mathsf{FE}}, \mathsf{Decrypt}_{\mathsf{FE}})$ is selectively (resp. adaptively) secure.

Theorem 8 (Correctness). *If* $\mathsf{FE} := (\mathsf{Setup}_{\mathsf{FE}}, \mathsf{KeyGen}_{\mathsf{FE}}, \mathsf{Encrypt}_{\mathsf{FE}}, \mathsf{Decrypt}_{\mathsf{FE}})$ *is a perfectly correct functional encryption scheme for functionality F, then so is the predicate encryption scheme* PE *defined in Figure 11.*

Proof of Theorem 8. By correctness of FE, we have for all $(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{X}$, $w \in \mathbb{Z}_p$, $\mathbf{F} \in \mathcal{Y}$:

$$F(\mathbf{F}, (w \cdot \boldsymbol{x}, \boldsymbol{y})) = [w \cdot \boldsymbol{x}^\top \mathbf{F} \boldsymbol{y}]_T = [w \cdot \mathsf{P}((\boldsymbol{x}, \boldsymbol{y}), \mathbf{F})]_T \,.$$

Thus, when $\mathsf{P}((\boldsymbol{x}, \boldsymbol{y}), \mathbf{F}) = 1$, decryption recovers the encapsulation key $[w]_T$. □

Theorem 9 (Security). *If* $\mathsf{FE} := (\mathsf{Setup}_{\mathsf{FE}}, \mathsf{KeyGen}_{\mathsf{FE}}, \mathsf{Encrypt}_{\mathsf{FE}}, \mathsf{Decrypt}_{\mathsf{FE}})$ *is an adaptively (resp. selectively) secure encryption scheme for F, then so is the predicate encryption scheme* PE *defined in Figure 11. Namely, for any PPT adversary $\mathcal{A}$, there exists a PPT adversary $\mathcal{B}$ such that:*

$$\mathbf{Adv}^{\mathsf{ind\text{-}fe\text{-}cpa}}_{\mathsf{PE},\mathcal{A}}(\lambda) \leq 4 \cdot \mathbf{Adv}^{\mathsf{ind\text{-}fe\text{-}cpa}}_{\mathsf{PE},\mathcal{B}}(\lambda).$$

Similarly, in the selective case, for any PPT adversary $\mathcal{A}$, there exists a PPT adversary $\mathcal{B}$ such that:

$$\mathbf{Adv}^{\mathsf{sel\text{-}ind\text{-}fe\text{-}cpa}}_{\mathsf{PE},\mathcal{A}}(\lambda) \leq 4 \cdot \mathbf{Adv}^{\mathsf{sel\text{-}ind\text{-}fe\text{-}cpa}}_{\mathsf{PE},\mathcal{B}}(\lambda).$$

```
$\mathrm{G}_0, \boxed{\mathrm{G}_1, \boxed{\mathrm{G}_2}}$:

$\beta \leftarrow_{\mathrm{R}} \{0,1\}$, $(\mathsf{mpk}, \mathsf{msk}) \leftarrow_{\mathrm{R}} \mathsf{Setup}_{\mathsf{FE}}(1^{\lambda}, F)$
$\beta' \leftarrow \mathcal{A}^{\mathsf{KeyGenO}(\cdot), \mathsf{EncO}(\cdot,\cdot,\cdot,\cdot)}(\mathsf{mpk})$
Return 1 if $\beta' = \beta$, 0 otherwise.

$\underline{\mathsf{EncO}((\boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}), M_0, (\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)}), M_1)}$:
$w \leftarrow_{\mathrm{R}} \mathbb{Z}_p$, $C_0 := [w]_T + M_\beta$, $C_1 := \mathsf{Encrypt}_{\mathsf{FE}}(\mathsf{mpk}, (w \cdot \boldsymbol{x}^{(\beta)}, \boldsymbol{y}^{(\beta)}))$
$\boxed{\text{If } M_0 \neq M_1,\ C_1 := \mathsf{Encrypt}_{\mathsf{FE}}(\mathsf{mpk}, (\mathbf{0}, \mathbf{0}))}$
$\boxed{\text{If } M_0 = M_1,\ C_1 := \mathsf{Encrypt}_{\mathsf{FE}}(\mathsf{mpk}, (w \cdot \boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}))}$
Return $\mathsf{Ct} := (C_0, C_1)$

$\underline{\mathsf{KeyGenO}(\mathbf{F} \in \mathbb{Z}_p^{n \times m})}$:
Return $\mathsf{sk}_{\mathbf{F}} := \mathsf{KeyGen}_{\mathsf{FE}}(\mathsf{msk}, \mathbf{F})$
```

Fig. 12. Games G_i, for $i = 0, 1, 2$ for the proof of adaptive security of PE in Figure 11. In each procedure, the components inside a solid (dotted) frame are only present in the games marked by a solid (dotted) frame.

Proof of Theorem 9, adaptive security. We prove the adaptive security of PE via a series of games described in Figure 12 and we use Adv_i to denote the advantage of $\mathcal{A}$ in game G_i, that is $\mathsf{Adv}_i := |1 - 2\Pr[\mathrm{G}_i \text{ returns } 1]|$. G_0 is defined as:

$$\mathrm{G}_0 : \begin{array}{l} \beta \leftarrow_{\mathrm{R}} \{0,1\} \\ \beta' \leftarrow \mathbf{Exp}^{\mathsf{ind\text{-}pe\text{-}cpa\text{-}}\beta}_{\mathsf{PE},\mathcal{A}}(\lambda) \\ \text{Return 1 if } \beta' = \beta, \text{ 0 otherwise.} \end{array}$$

Where $\mathbf{Exp}^{\mathsf{ind\text{-}pe\text{-}cpa\text{-}}\beta}_{\mathsf{PE},\mathcal{A}}(\lambda)$ is the experiment used in Definition 9 of fully attribute-hiding security for predicate encryption. In particular, we have $\mathsf{Adv}_0 = \mathbf{Adv}^{\mathsf{ind\text{-}pe\text{-}cpa}}_{\mathsf{PE},\mathcal{A}}(\lambda)$. We explain in Remark 1 how to obtain the same results for selective security.

Lemma 14 **(G_0 to G_1).** *There exists a PPT adversary $\mathcal{B}_0$:*

$$|\mathsf{Adv}_0 - \mathsf{Adv}_1| \leq 2 \cdot \mathbf{Adv}^{\mathsf{ind\text{-}fe\text{-}cpa}}_{\mathsf{PE},\mathcal{B}_0}(\lambda).$$

Proof of Lemma 14. By definition of the security game, we know that if $M_0 \neq M_1$, then it must be that for all queries $\mathbf{F}$ to $\mathsf{KeyGenO}(\cdot)$, $\boldsymbol{x}^{(\beta)\top}\mathbf{F}\boldsymbol{y}^{(\beta)} = 0$ (i.e., the predicate over the challenge attributes is false). Therefore, using the adaptive security of the underlying FE scheme, we can switch: $\mathsf{Encrypt}(\mathsf{mpk}, (w \cdot \boldsymbol{x}^{(\beta)}, \boldsymbol{y}^{(\beta)}))$, computed by EncO when $M_0 \neq M_1$, to $\mathsf{Encrypt}(\mathsf{mpk}, (\mathbf{0}, \mathbf{0}))$. □

Lemma 15 **(G_1 to G_2).** *There exists a PPT adversary $\mathcal{B}_1$:*

$$|\mathsf{Adv}_1 - \mathsf{Adv}_2| \leq 2 \cdot \mathbf{Adv}^{\mathsf{ind\text{-}fe\text{-}cpa}}_{\mathsf{PE},\mathcal{B}_1}(\lambda).$$

Proof of Lemma 15. By definition of the security game, we know that for all queries $\mathbf{F}$ to $\mathsf{KeyGenO}(\cdot)$, $\mathsf{P}\big((\boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}), \mathbf{F}\big) = \mathsf{P}\big((\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)}), \mathbf{F}\big)$. Together with the fact that for all $(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{X}$ and $\mathbf{F} \in \mathcal{Y}$:

$\boldsymbol{x}^\top \mathbf{F} \boldsymbol{y} \in \{0,1\}$, we obtain that: $\boldsymbol{x}^{(0)\top} \mathbf{F} \boldsymbol{y}^{(0)} = \boldsymbol{x}^{(1)\top} \mathbf{F} \boldsymbol{y}^{(1)}$. Therefore, using the adaptive security of the underlying FE scheme, we can switch: $\mathsf{Encrypt}(\mathsf{mpk}, (w \cdot \boldsymbol{x}^{(\beta)}, \boldsymbol{y}^{(\beta)}))$, computed by EncO when $M_0 = M_1$, to $\mathsf{Encrypt}(\mathsf{mpk}, (w \cdot \boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}))$. □

Lemma 16 (G$_2$)**.** $\mathsf{Adv}_2 = 0$.

Proof of Lemma 16. We show that the $\mathcal{A}$'s view is independent of $\beta \leftarrow_{\mathrm{R}} \{0,1\}$ in this game. If $M_0 \neq M_1$, the challenge ciphertext is of the form (C_0, C_1) where $C_0 := [w]_T + M_\beta$ for $w \leftarrow_{\mathrm{R}} \mathbb{Z}_p$, and C_1 is independent of w and β. Thus, the message M_β is completely hidden by the one-time pad $[w]_T$, and the ciphertext is independent of β.

If $M_0 = M_1$, the challenge ciphertext is of the form (C_0, C_1) where $C_0 := [w]_T + M_\beta$ for $w \leftarrow_{\mathrm{R}} \mathbb{Z}_p$, which is independent of β since $M_0 = M_1$; and $C_1 := \mathsf{Encrypt}(\mathsf{mpk}, (w \cdot \boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}))$, which is also independent of β. □

Theorem 9 follows readily from Lemmas 14, 15, and 16. □

Remark 1 (Selective FE ⇒ selective PE). We can adapt straightforwardly the proof of Theorem 9, to the selective setting, simply by constructing PPT adversaries $\mathcal{B}_0$ and $\mathcal{B}_1$ against the selective security of the underlying FE, exactly as those in Lemmas 14 and 15, except that those adversaries first receive a challenge $(\boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}), (\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})$ from the adversary $\mathcal{A}$, playing against the selective security of the PE, upon which they sample $w \leftarrow_{\mathrm{R}} \mathbb{Z}_p$, and send $(w \cdot \boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}), (w \cdot \boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)})$ as their selective challenge. Finally, we use the statistical argument from Lemma 16, which works exactly in the same way for the selective setting.

6.1 Applications of PE for Bilinear Maps Evaluation

In this section, we discuss two applications of our fully attribute-hiding PE scheme supporting bilinear maps evaluation.

PE for constant depth boolean formulas. As a first application, we can use the PE scheme in Figure 11 to handle boolean functions of constant degree d in n variables. This yields a solution where ciphertexts comprise $O(n^{d/2})$ group elements, in contrast to $O(n^d)$ group elements in [28] (the asymptotic is taken for large n, constant d).

The idea is to encode a predicate for boolean formulas into a predicate for bilinear maps evaluation. This can be done as follows. Consider the following predicate $\mathsf{P} : \mathcal{X} \times \mathcal{Y} \rightarrow \{0,1\}$, with $\mathcal{X} := \mathbb{Z}_2^n$ and $\mathcal{Y} := \{T \in \mathbb{Z}_2[X_1, \ldots, X_n], \deg(T) \leq d\}$, such that for all $\boldsymbol{x} \in \mathcal{X}$, $T \in \mathcal{X}$, $\mathsf{P}(\boldsymbol{x}, T) = 1$ iff $T(\boldsymbol{x}) = 1$. Below we describe how to encode $\boldsymbol{x} \in \mathcal{X}$ and $T \in \mathcal{Y}$ into a vector $\widetilde{\boldsymbol{x}}$ and a matrix $\widetilde{\mathbf{T}}$ such that $\mathsf{P}(\boldsymbol{x}, T) = 1$ iff $\widetilde{\boldsymbol{x}}^\top \widetilde{\mathbf{T}} \widetilde{\boldsymbol{x}} = 1$.

To see this, assume for simplicity that d is even, and let us consider the setting where $n \geq \frac{d}{2}$. First, we map every $\boldsymbol{x} \in \mathcal{X}$ to $\widetilde{\boldsymbol{x}} := (M_1(\boldsymbol{x}), \ldots, M_{\widetilde{d}}(\boldsymbol{x})) \in \mathbb{Z}_2^{\widetilde{d}}$, where $\widetilde{d} := \sum_{i=0}^{\frac{d}{2}} \binom{n}{i}$, and for all $j \in \left[\binom{n}{\frac{d}{2}}\right]$, M_j is the j-th monomial of degree at most $\frac{d}{2}$ on n variables (there are exactly $\widetilde{d}$ such monomials, which we order arbitrarily). Second, we write every $T \in \mathcal{Y}$ as $\sum_{i,j \in [\widetilde{d}]} T_{i,j} M_i M_j$, where for all $i, j \in [\widetilde{d}]$, $T_{i,j} \in \mathbb{Z}_2$, and we map $T \in \mathcal{Y}$ to $\widetilde{\mathbf{T}} \in \mathbb{Z}_2^{\widetilde{d} \times \widetilde{d}}$ such that for all $i, j \in [\widetilde{d}]$, $\widetilde{T}_{i,j} := T_{i,j}$. This way, for all $\boldsymbol{x} \in \mathcal{X}$ and $T \in \mathcal{Y}$, we have $\mathsf{P}(\boldsymbol{x}, T) = 1$ iff $\widetilde{\boldsymbol{x}}^\top \widetilde{\mathbf{T}} \widetilde{\boldsymbol{x}} = T(\boldsymbol{x}) = 1$.

Therefore, using the PE which supports bilinear maps evaluation presented in Section 6, we obtain a PE for boolean formulas with ciphertexts of size $O(\widetilde{d})$. Using a similar encoding to the PE from [28] that support linear maps evaluation yields a solution with ciphertexts of dimension $O(\widehat{d})$ where $\widehat{d} := \sum_{i=0}^{d} \binom{n}{i}$. When considering asymptotic for large n, constant d, our ciphertext size is $O(n^{d/2})$, against $O(n^d)$ for [28].

Finally, we note that boolean formulas can be arithmetized into a polynomial over $\mathbb{Z}_2$, à la [38]. Namely, for boolean variables $x, y \in \mathbb{Z}_2$, $\mathsf{AND}(x,y)$ is encoded as $x \cdot y$, $\mathsf{OR}(x,y)$ is encoded as $x + y - xy$, and $\mathsf{NOT}(x) = 1 - x$.

PE for comparison. Let us consider the comparison predicate $\mathsf{P}_{\leq} : [N] \times [N] \to \{0,1\}$ that for all $x, y \in [N]$ is defined by

$$\mathsf{P}_{\leq}(x,y) = 1 \text{ iff } x \leq y.$$

We can reduce this predicate to a polynomial of degree two, as done (implicitly) in [12], as follows. First, any integer $x \in [N]$ is canonically mapped to the lexicographically ordered pair $(x_1, x_2) \in [\sqrt{N}] \times [\sqrt{N}]$ (we assume $\sqrt{N}$ is an integer for simplicity). Then x_1 is mapped to vectors $\widetilde{\boldsymbol{x}} := \begin{pmatrix} \mathbf{0}^{x_1} \\ \mathbf{1}^{\sqrt{N}-x_1} \end{pmatrix} \in \{0,1\}^{\sqrt{N}}$ where $\mathbf{1}^{\ell}$, $\mathbf{0}^{\ell}$ denote the all-one and all-zero vectors in $\{0,1\}^{\ell}$, respectively; and $\widehat{\boldsymbol{x}} := \boldsymbol{e}_{x_1} \in \{0,1\}^{\sqrt{N}}$, where for all $i \in [\sqrt{N}]$, $\boldsymbol{e}_i$ denotes the i'th vector of the canonical basis of $\mathbb{Z}_p^{\sqrt{N}}$. Finally, $x_2 \in [\sqrt{N}]$ is mapped to $\bar{\boldsymbol{x}} := \begin{pmatrix} \mathbf{0}^{x_2-1} \\ \mathbf{1}^{\sqrt{N}-x_2+1} \end{pmatrix}$. For all $(x_1, x_2), (y_1, y_2) \in [\sqrt{N}] \times [\sqrt{N}]$:

$$\mathsf{P}_{\leq}((x_1,x_2),(y_1,y_2)) = 1 \text{ iff } \widetilde{x}_{y_1} + \widehat{x}_{y_1} \cdot \bar{x}_{y_2} = 1,$$

where for any vector $\boldsymbol{z} \in \mathbb{Z}_p^{\sqrt{N}}$, and any $i \in [\sqrt{N}]$, we denote by $z_i \in \mathbb{Z}_p$ the i-th coordinate of $\boldsymbol{z}$.

This means that by using the above encoding, for an integer attribute $x \in [N]$ one can use a PE for bilinear maps evaluation to encrypt the pair of vectors

$$\left(\begin{pmatrix} \widetilde{\boldsymbol{x}} \\ \widehat{\boldsymbol{x}} \end{pmatrix}, \begin{pmatrix} 1 \\ \bar{\boldsymbol{x}} \end{pmatrix} \right) \in \mathbb{Z}_p^{2\sqrt{N}} \times \mathbb{Z}_p^{1+\sqrt{N}}$$

This gives a PE for comparison with ciphertexts of $O(\sqrt{N})$ group elements, as in [12,21]. More precisely, by instantiating our PE scheme with the FE of Section 3.2, we obtain a PE for comparison with ciphertext size $(12\sqrt{N}+1) \cdot |\mathbb{G}_1| + (6\sqrt{N}+7) \cdot |\mathbb{G}_2|$, and secret-key size $|\mathbb{G}_1| + |\mathbb{G}_2|$, compared to ciphertext size $5\sqrt{N} \cdot |\mathbb{G}_1| + 4\sqrt{N} \cdot |\mathbb{G}_2| + |\mathbb{G}_T|$ and secret-key size $|\mathbb{G}_2|$ for [21], where both schemes are selectively-secure based on SXDH. When using our FE of Section 4, we obtain a PE for comparisons that is adaptive secure in the generic group model and that has shorter ciphertexts of size $(4\sqrt{N}+1) \cdot |\mathbb{G}_1| + (2\sqrt{N}+3) \cdot |\mathbb{G}_2|$.

References

1. M. Abdalla, M. Bellare, D. Catalano, E. Kiltz, T. Kohno, T. Lange, J. Malone-Lee, G. Neven, P. Paillier, and H. Shi. Searchable encryption revisited: Consistency properties, relation to anonymous IBE, and extensions. In V. Shoup, editor, *CRYPTO 2005*, volume 3621 of *LNCS*, pages 205–222. Springer, Aug. 2005.
2. M. Abdalla, F. Bourse, A. De Caro, and D. Pointcheval. Simple functional encryption schemes for inner products. In *PKC 2015*, LNCS, pages 733–751. Springer, 2015.
3. M. Abdalla, F. Bourse, A. De Caro, and D. Pointcheval. Better security for functional encryption for inner product evaluations. Cryptology ePrint Archive, Report 2016/011, 2016. `http://eprint.iacr.org/2016/011`.

4. M. Abdalla, M. Raykova, and H. Wee. Multi-input inner-product functional encryption from pairings. *IACR Cryptology ePrint Archive*, 2016:425, 2016.
5. S. Agrawal, D. M. Freeman, and V. Vaikuntanathan. Functional encryption for inner product predicates from learning with errors. In D. H. Lee and X. Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 21–40. Springer, Dec. 2011.
6. S. Agrawal, B. Libert, and D. Stehlé. Fully secure functional encryption for inner products, from standard assumptions. LNCS, pages 333–362. Springer, Aug. 2016.
7. P. Ananth and A. Sahai. Projective arithmetic functional encryption and indistinguishability obfuscation from degree-5 multilinear maps. Cryptology ePrint Archive, Report 2016/1097, 2016. `http://eprint.iacr.org/2016/1097`.
8. G. Barthe, E. Fagerholm, D. Fiore, J. C. Mitchell, A. Scedrov, and B. Schmidt. Automated analysis of cryptographic assumptions in generic group models. In J. A. Garay and R. Gennaro, editors, *CRYPTO 2014, Part I*, volume 8616 of *LNCS*, pages 95–112. Springer, Aug. 2014.
9. A. Bishop, A. Jain, and L. Kowalczyk. Function-hiding inner product encryption. LNCS, pages 470–491. Springer, Dec. 2015.
10. D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano. Public key encryption with keyword search. In C. Cachin and J. Camenisch, editors, *EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 506–522. Springer, May 2004.
11. D. Boneh and M. K. Franklin. Identity-based encryption from the Weil pairing. In J. Kilian, editor, *CRYPTO 2001*, volume 2139 of *LNCS*, pages 213–229. Springer, Aug. 2001.
12. D. Boneh, A. Sahai, and B. Waters. Fully collusion resistant traitor tracing with short ciphertexts and private keys. In S. Vaudenay, editor, *EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 573–592. Springer, May / June 2006.
13. D. Boneh, A. Sahai, and B. Waters. Functional encryption: Definitions and challenges. In Y. Ishai, editor, *TCC 2011*, volume 6597 of *LNCS*, pages 253–273. Springer, Mar. 2011.
14. D. Boneh and B. Waters. A fully collusion resistant broadcast, trace, and revoke system. In A. Juels, R. N. Wright, and S. Vimercati, editors, *ACM CCS 06*, pages 211–220. ACM Press, Oct. / Nov. 2006.
15. D. Boneh and B. Waters. Conjunctive, subset, and range queries on encrypted data. In S. P. Vadhan, editor, *TCC 2007*, volume 4392 of *LNCS*, pages 535–554. Springer, Feb. 2007.
16. J. Chen, R. Gay, and H. Wee. Improved dual system ABE in prime-order groups via predicate encodings. LNCS, pages 595–624. Springer, 2015.
17. J. Chen, H. W. Lim, S. Ling, H. Wang, and H. Wee. Shorter IBE and signatures via asymmetric pairings. In M. Abdalla and T. Lange, editors, *PAIRING 2012*, volume 7708 of *LNCS*, pages 122–140. Springer, May 2013.
18. A. Escala, G. Herold, E. Kiltz, C. Ràfols, and J. Villar. An algebraic framework for Diffie-Hellman assumptions. In R. Canetti and J. A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 129–147. Springer, Aug. 2013.
19. S. D. Galbraith, K. G. Paterson, and N. P. Smart. Pairings for cryptographers. *Discrete Appl. Math.*, 156(16):3113–3121, Sept. 2008.
20. S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B. Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *54th FOCS*, pages 40–49. IEEE Computer Society Press, Oct. 2013.
21. S. Garg, A. Kumarasubramanian, A. Sahai, and B. Waters. Building efficient fully collusion-resilient traitor tracing and revocation schemes. In E. Al-Shaer, A. D. Keromytis, and V. Shmatikov, editors, *ACM CCS 10*, pages 121–130. ACM Press, Oct. 2010.
22. C. Gentry. Practical identity-based encryption without random oracles. In S. Vaudenay, editor, *EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 445–464. Springer, May / June 2006.
23. S. Goldwasser, Y. T. Kalai, R. A. Popa, V. Vaikuntanathan, and N. Zeldovich. Reusable garbled circuits and succinct functional encryption. In D. Boneh, T. Roughgarden, and J. Feigenbaum, editors, *45th ACM STOC*, pages 555–564. ACM Press, June 2013.
24. S. Gorbunov, V. Vaikuntanathan, and H. Wee. Functional encryption with bounded collusions via multi-party computation. In R. Safavi-Naini and R. Canetti, editors, *CRYPTO 2012*, volume 7417 of *LNCS*, pages 162–179. Springer, Aug. 2012.
25. S. Gorbunov, V. Vaikuntanathan, and H. Wee. Attribute-based encryption for circuits. In D. Boneh, T. Roughgarden, and J. Feigenbaum, editors, *45th ACM STOC*, pages 545–554. ACM Press, June 2013.
26. S. Gorbunov, V. Vaikuntanathan, and H. Wee. Predicate encryption for circuits from LWE. In *CRYPTO 2015, Part II*, LNCS, pages 503–523. Springer, Aug. 2015.

27. V. Goyal, O. Pandey, A. Sahai, and B. Waters. Attribute-based encryption for fine-grained access control of encrypted data. In A. Juels, R. N. Wright, and S. Vimercati, editors, *ACM CCS 06*, pages 89–98. ACM Press, Oct. / Nov. 2006. Available as Cryptology ePrint Archive Report 2006/309.
28. J. Katz, A. Sahai, and B. Waters. Predicate encryption supporting disjunctions, polynomial equations, and inner products. In N. P. Smart, editor, *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 146–162. Springer, Apr. 2008.
29. J. Katz, A. Sahai, and B. Waters. Predicate encryption supporting disjunctions, polynomial equations, and inner products. *Journal of Cryptology*, 26(2):191–224, Apr. 2013.
30. A. B. Lewko. Tools for simulating features of composite order bilinear groups in the prime order setting. In D. Pointcheval and T. Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 318–335. Springer, Apr. 2012.
31. H. Lin. Indistinguishability obfuscation from ddh on 5-linear maps and locality-5 prgs. Cryptology ePrint Archive, Report 2016/1096, 2016. `http://eprint.iacr.org/2016/1096`.
32. U. M. Maurer. Abstract models of computation in cryptography (invited paper). In N. P. Smart, editor, *10th IMA International Conference on Cryptography and Coding*, volume 3796 of *LNCS*, pages 1–12. Springer, Dec. 2005.
33. T. Okamoto and K. Takashima. Homomorphic encryption and signatures from vector decomposition. In S. D. Galbraith and K. G. Paterson, editors, *PAIRING 2008*, volume 5209 of *LNCS*, pages 57–74. Springer, Sept. 2008.
34. T. Okamoto and K. Takashima. Hierarchical predicate encryption for inner-products. In M. Matsui, editor, *ASIACRYPT 2009*, volume 5912 of *LNCS*, pages 214–231. Springer, Dec. 2009.
35. A. O'Neill. Definitional issues in functional encryption. Cryptology ePrint Archive, Report 2010/556, 2010. `http://eprint.iacr.org/2010/556`.
36. A. Sahai and B. R. Waters. Fuzzy identity-based encryption. In R. Cramer, editor, *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 457–473. Springer, May 2005.
37. A. Shamir. Identity-based cryptosystems and signature schemes. In G. R. Blakley and D. Chaum, editors, *CRYPTO'84*, volume 196 of *LNCS*, pages 47–53. Springer, Aug. 1984.
38. A. Shamir. IP=PSPACE. In *31st FOCS*, pages 11–15. IEEE Computer Society Press, Oct. 1990.
39. V. Shoup. Lower bounds for discrete logarithms and related problems. In W. Fumy, editor, *EUROCRYPT'97*, volume 1233 of *LNCS*, pages 256–266. Springer, May 1997.
40. B. Waters. Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In S. Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 619–636. Springer, Aug. 2009.