Multi-Input Quadratic Functional Encryption from Pairings

Shweta Agrawal[1⋆], Rishab Goyal[2⋆⋆], and Junichi Tomida[3]

[1] IIT Madras
shweta.a@cse.iitm.ac.in
[2] MIT
goyal@utexas.edu
[3] NTT Corporation
junichi.tomida.vw@hco.ntt.co.jp

Abstract. We construct the first multi-input functional encryption (MIFE) scheme for quadratic functions from pairings. Our construction supports polynomial number of users, where user i, for $i \in [n]$, encrypts input $\mathbf{x}_i \in \mathbb{Z}^m$ to obtain ciphertext CT_i, the key generator provides a key $\mathsf{SK}_{\mathbf{c}}$ for vector $\mathbf{c} \in \mathbb{Z}^{(mn)^2}$ and decryption, given $\mathsf{CT}_1, \ldots, \mathsf{CT}_n$ and $\mathsf{SK}_{\mathbf{c}}$, recovers $\langle \mathbf{c}, \mathbf{x} \otimes \mathbf{x} \rangle$ and nothing else. We achieve indistinguishability-based (selective) security against unbounded collusions under the standard bilateral matrix Diffie-Hellman assumption. All previous MIFE schemes either support only inner products (linear functions) or rely on strong cryptographic assumptions such as indistinguishability obfuscation or multi-linear maps.

Keywords: functional encryption, multi-input, quadratic functions, pairings

1 Introduction

Functional encryption (FE) [12,29] is a novel cryptographic paradigm that moves beyond the "all or nothing" access of traditional public key encryption and enables fine grained access to encrypted data. Concretely, an FE scheme that supports a function class $\mathcal{F}$ allows an owner of a master secret to issue a secret key

⋆ Research supported by the DST "Swarnajayanti" fellowship, an Indo-French CEFIPRA project and the CCD Centre of Excellence. Part of the research corresponding to this work was conducted while visiting the Simons Institute for the Theory of Computing.

⋆⋆ Research supported in part by NSF CNS Award #1718161, an IBM-MIT grant, and by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR00112020023. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Government or DARPA. Work done in part while at the Simons Institute for the Theory of Computing, supported by Simons-Berkeley research fellowship.

SK_f for a function $f \in \mathcal{F}$. Decryption of a ciphertext CT_x for a message x with SK_f yields $f(x)$ and nothing else. Functional encryption has been extensively studied in the literature, with elegant constructions supporting various function classes, achieving different notions of security and from diverse assumptions, e.g., [3, 9, 13, 19, 20].

Multi-input functional encryption (MIFE) [22] is a natural generalization of FE, which supports functions that take multiple inputs. In MIFE, multiple parties can encrypt data independently – thus, n users may encrypt their data $x_1, \ldots, x_n$ to produce ciphertexts $\mathsf{CT}_1, \ldots, \mathsf{CT}_n$, which can be decrypted using a functional key SK_f to learn $f(x_1, \ldots, x_n)$ and nothing else.

Research in MIFE has followed two broad directions. On one hand, it was shown that for general function classes (all polynomial sized circuits), FE is powerful enough to imply MIFE (albeit with exponential loss), which in turn implies the powerful notion of indistinguishability obfuscation (iO) [8, 11]. On the other hand, for restricted function classes such as constant degree polynomials, single-input schemes do not generically imply multi-input schemes and constructing multi-input schemes directly proved significantly more challenging. Intuitively, this is because in the multi-input setting, inputs $x_1, \ldots, x_n$ encrypted using independent sources of randomness must be combined in a secure way to "emulate" the single input setting where encodings of $x_1, \ldots, x_n$ may be tied together using common randomness. Nevertheless, for the inner product functionality, several novel MIFE constructions emerged based on simple, standard polynomial hardness assumptions [1, 2, 4, 6, 15, 17, 27, 30].

Beyond Inner Products. While the inner product functionality is useful for several meaningful applications (we refer the reader to [6] for a discussion), it is evidently desirable, from the viewpoint of both theory and practice, to extend the reach of MIFE from standard assumptions beyond inner products. In the single input setting, there has been significant progress in this direction. For quadratic functions, several FE schemes have been constructed from standard assumptions on pairings [9, 21, 28][4]. Indeed, from pairings, there have also been innovative constructions for "degree 2.5" FE [7], the so-called "partially hiding functional encryption" (PHFE) schemes. Intuitively, PHFE permits part of the encryptor's input to be public and supports deeper computation on the public input as compared to the private input.

However, in the multi-input setting, constructions going beyond inner products have proved elusive. Note that unlike the single input setting, quadratic MIFE cannot be trivially constructed from inner product MIFE even with large ciphertext, since the naive idea of encrypting all quadratic monomials in advance cannot deal with quadratic terms derived from two different users. Therefore, there are currently *no* candidate constructions for MIFE supporting quadratic

[4] Note that FE for quadratic functions are trivially constructible from FE for inner products (IPFE) by linearizing and encrypting all quadratic monomials. However, FE for quadratic functions requires that the ciphertext size be linear in input length.

polynomials, from standard, polynomial hardness assumptions[5]. This is a significant gap in our understanding of MIFE, and motivates the fundamental question:

Can we construct MIFE for quadratic functions from pairings?

1.1 Our Results

In this work, we answer the above question affirmatively and construct the first MIFE scheme for quadratic functions from pairings. In more detail, we construct n-input MIFE scheme for the function class $\mathcal{F}_{m,n}$, which is defined as follows. Each function $f \in \mathcal{F}_{m,n}$ is represented by a vector $\mathbf{c} \in \mathbb{Z}^{(mn)^2}$. For inputs $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{Z}^m$, f is defined as $f(\mathbf{x}_1, \ldots, \mathbf{x}_n) := \langle \mathbf{c}, \mathbf{x} \otimes \mathbf{x} \rangle$ where $\mathbf{x} = (\mathbf{x}_1 || \cdots || \mathbf{x}_n)$ and $\otimes$ denotes the Kronecker product. In a quadratic MIFE scheme for $\mathcal{F}_{m,n}$, a user can encrypt $\mathbf{x}_i \in \mathbb{Z}^m$ to CT_i for slot $i \in [n]$, a key issuer can generate a secret key SK for $\mathbf{c} \in \mathbb{Z}^{(mn)^2}$, and decryption of $\mathsf{CT}_1, \ldots, \mathsf{CT}_n$ with SK reveals only $\langle \mathbf{c}, \mathbf{x} \otimes \mathbf{x} \rangle$ and nothing else.

To begin, we show that in the public key setting, quadratic MIFE can be generically obtained from public-key IPFE, which can be obtained even without pairings, in a relatively simple manner, as the case of public-key inner product MIFE [6]. Then we provide our main construction in the much more challenging secret-key setting[6]. Our construction relies on the bilateral matrix Diffie-Hellmen assumption [18] and achieves standard indistinguishability-based (selective) security against unbounded collusions. We observe that in the symmetric key setting, selective security is the same as "semi-adaptive" [14, 23] security. Recall that in semi-adaptive security, the adversary is permitted to see the public key before committing to the challenge. In the symmetric key setting, since the "public key" is simply public parameters of the scheme, such as group description, which may always be provided to the adversary in the first step of the game, the distinction between selective and semi-adaptive is moot. Thus, our construction achieves the same level of security as single input quadratic FE [9, 21, 28].

Our construction is built using two newly introduced primitives that we call predicated IPFE and mixed-group multi-input IPFE, which we describe next. Predicated IPFE (pIPFE) is a class of attribute-based IPFE [5], but additionally with a function hiding property. In more detail, a ciphertext pCT and a secret key pSK of a pIPFE scheme pFE are associated with two vectors $\{\mathbf{x}_1, \mathbf{x}_2\}$ and $\{\mathbf{y}_1, \mathbf{y}_2\}$, respectively. Decryption of pCT with pSK reveals $\langle \mathbf{x}_2, \mathbf{y}_2 \rangle$ iff $\langle \mathbf{x}_1, \mathbf{y}_1 \rangle = 0$. Secret keys are required to hide $\mathbf{y}_2$ but not $\mathbf{y}_1$, This scheme is the first instantiation of function-hiding attribute-based IPFE, which may be of independent

[5] In an exciting recent work, iO has been constructed from sub-exponential hardness of four well-founded assumptions [24]. However, this construction relies on a series of intricate, lossy reductions and is primarily a feasibility result. We will focus on the *polynomial hardness* of a well-founded problem in this work.

[6] Recall that public-key MIFE does not imply secret-key MIFE. Roughly speaking, a user who has CT_1 for x_1 and SK for f of a public-key scheme is allowed to learn $f(x_1, x_2, \ldots, x_n)$ for all $(x_2, \ldots, x_n)$, since this is inherent leakage, while it is not the case in secret-key MIFE.

interest. Mixed group multi input IPFE is similar to multi input IPFE but supports mixed groups, as suggested by the name. In more detail, consider a function $f : (G_1^{m_1} \times G_2^{m_2})^n \to G_T$, specified by $([\mathbf{y}_{1,1}]_2, [\mathbf{y}_{1,2}]_1, \ldots, [\mathbf{y}_{n,1}]_2, [\mathbf{y}_{n,2}]_1)$ where $\mathbf{y}_{i,1} \in \mathbb{Z}_p^{m_1}$ and $\mathbf{y}_{i,2} \in \mathbb{Z}_p^{m_2}$ and defined as $f\big(([\mathbf{x}_{1,1}]_1, [\mathbf{x}_{1,2}]_2), \ldots, ([\mathbf{x}_{n,1}]_1, [\mathbf{x}_{n,2}]_2)\big)$ $:= [\langle(\mathbf{x}_{1,1}, \mathbf{x}_{1,2}, \ldots, \mathbf{x}_{n,1}, \mathbf{x}_{n,2}), (\mathbf{y}_{1,1}, \mathbf{y}_{1,2}, \ldots, \mathbf{y}_{n,1}, \mathbf{y}_{n,2})\rangle]_T$

Mixed group multi input IPFE is also required to achieve function-hiding. We provide constructions for these primitives by leveraging a (multi-input) function-hiding IPFE scheme based on pairings [4,10,17]. These constructions may be of independent interest.

1.2 Our Techniques

As discussed above, quadratic MIFE in the public-key setting is simple to achieve due to the leakage inherent in that setting. We formalize this in the full version of this paper. Hence, as in prior work [6], we focus on the much more challenging secret key setting. In the following, we basically use m for the vector length of each user and n for the number of slots.

Lin's Single Key Quadratic FE. The starting point of our secret-key quadratic MIFE scheme is the secret-key quadratic FE scheme from pairings by Lin [28], which in turn builds upon the public key IPFE scheme from DDH by Abdalla *et al.* [3] (ABDP). We begin by recalling the ABDP scheme. In what follows, we let g_ℓ denote the generator of a cyclic group of order p and for matrix $\mathbf{A} = (a_{i,j})_{i,j}$, we denote $(g_\ell^{a_{i,j}})_{i,j}$ by $[\mathbf{A}]_\ell$. The ABDP scheme works as follows:

$\mathsf{Setup}(1^\lambda)$**:** $\mathbf{w} \leftarrow \mathbb{Z}_p^m$, $\mathsf{PK} := [\mathbf{w}]$, $\mathsf{MSK} := \mathbf{w}$.
$\mathsf{Enc}(\mathsf{PK}, \mathbf{x} \in \mathbb{Z}^m)$**:** $s \leftarrow \mathbb{Z}_p$, $\mathsf{CT} := ([s], [\mathbf{x} + s\mathbf{w}])$.
$\mathsf{KeyGen}(\mathsf{MSK}, \mathbf{c} \in \mathbb{Z}^m)$**:** $\mathsf{SK} := -\mathbf{c}^\top \mathbf{w}$.
$\mathsf{Dec}(\mathsf{CT}, \mathsf{SK})$**:** $-\mathbf{c}^\top \mathbf{w}[s] + \mathbf{c}^\top[\mathbf{x} + s\mathbf{w}] = [\langle \mathbf{c}, \mathbf{x}\rangle]$.

Lin's quadratic (secret key) FE scheme uses a clever interleaving of IPFE schemes. To compress the size of ABDP ciphertexts for quadratic terms, she leverages function-hiding IPFE, which is inherently secret-key [10]. Decryption of components in this scheme yields ciphertexts under the ABDP IPFE scheme, while secret keys of the ABDP scheme are generated using another function hiding IPFE. Finally, decryption of ABDP IPFE allows to recover the output.

In more detail, let $\mathsf{iFE} = (\mathsf{iSetup}, \mathsf{iEnc}, \mathsf{iKeyGen}, \mathsf{iDec})$ be a function-hiding IPFE scheme based on pairings. Note that all known function-hiding IPFE schemes based on pairings output a decryption value as an exponent of the target-group generator [10,16,26,28,31]. A simplification of her quadratic FE scheme (we omit the components of the scheme that are only required for the proof of security) is as follows:

$\mathsf{Setup}(1^\lambda)$**:** $\mathbf{w} = (w_1, \ldots, w_m), \widetilde{\mathbf{w}} = (\widetilde{w}_1, \ldots, \widetilde{w}_m) \leftarrow \mathbb{Z}_p^m$, $\mathsf{iMSK}' \leftarrow \mathsf{iSetup}(1^\lambda)$
$\mathsf{MSK} := (\mathsf{iMSK}', \mathbf{w}, \widetilde{\mathbf{w}})$.
$\mathsf{Enc}(\mathsf{MSK}, \mathbf{x} \in \mathbb{Z}^m)$**:** $s \leftarrow \mathbb{Z}_p$, $\mathsf{iCT}' \leftarrow \mathsf{iEnc}(\mathsf{iMSK}', s)$, $\mathsf{iMSK} \leftarrow \mathsf{iSetup}(1^\lambda)$
$\mathsf{iCT}_i \leftarrow \mathsf{iEnc}(\mathsf{iMSK}, (x_i, w_i)), \mathsf{iSK}_i \leftarrow \mathsf{iKeyGen}(\mathsf{iMSK}, (x_i, s\widetilde{w}_i))$.
$\mathsf{CT} := (\mathsf{iCT}', \{\mathsf{iCT}_i, \mathsf{iSK}_i\}_{i \in [m]})$.

$\mathsf{KeyGen}(\mathsf{MSK}, \mathbf{c} = \{c_{i,j}\}_{i,j\in[m]} \in \mathbb{Z}^{m^2})$**:**
$\mathsf{SK} := \mathsf{iSK}' \leftarrow \mathsf{iKeyGen}(\mathsf{MSK}', -\mathbf{c}^\top(\mathbf{w} \otimes \widetilde{\mathbf{w}}))$.
$\mathsf{Dec}(\mathsf{CT}, \mathsf{SK})$**:** $\mathsf{iDec}(\mathsf{iCT}', \mathsf{iSK}') + \sum_{i,j\in[m]} c_{i,j}\mathsf{iDec}(\mathsf{iCT}_i, \mathsf{iSK}_j) = [\langle \mathbf{c}, \mathbf{x} \otimes \mathbf{x}\rangle]_T$.

To decrypt, we compute $\mathsf{iDec}(\mathsf{iCT}_i, \mathsf{iSK}_j) = [x_i x_j + s w_i \widetilde{w}_j]_T$, which can be seen as the (i, j)-th element of the ABDP ciphertext $[\mathbf{x} \otimes \mathbf{x} + s\mathbf{w} \otimes \widetilde{\mathbf{w}}]_T$, and $\mathsf{iDec}(\mathsf{iCT}', \mathsf{iSK}') = [-s\mathbf{c}^\top(\mathbf{w} \otimes \widetilde{\mathbf{w}})]_T$, where $-\mathbf{c}^\top(\mathbf{w} \otimes \widetilde{\mathbf{w}})$ is an ABDP secret key for $\mathbf{c}$. The function-hiding property of iFE guarantees that iSK hides x_i. Since $\mathbf{w} \otimes \widetilde{\mathbf{w}}$ only appears on the exponent, one can argue that it is computationally indistinguishable from random in the security proof using the SXDH assumpetion.

IP-MIFE instead of IPFE. To generalize the above scheme to the multi-input setting, our first attempt is to modify Lin's scheme so that decryption of the function hiding IPFE scheme generates ciphertexts of a *multi-input* IPFE (IP-MIFE) scheme [4] (ACFGU) instead of a single input IPFE scheme (ABDP). Intuitively, the reason for using IP-MIFE instead of IPFE is to deal with multiple independent randomnesses derived from different users, which inherently come in when generating the IPFE ciphertext elements for quadratic terms. Now, we may hope that the key generator can provide a secret key matching the ACFGU scheme so that decryption of ciphertexts of the ACFGU scheme yields the desired result. Fortunately, the ACFGU scheme does not use pairings, so this basic template does not seem impossible. However, this starting point idea runs into several hurdles as we discuss below.

Let us recall the n-input ACFGU scheme:

$\mathsf{Setup}(1^\lambda)$**:** $\mathsf{MSK} := \mathbf{w}_1, \ldots, \mathbf{w}_n, \mathbf{u}_1, \ldots, \mathbf{u}_n \leftarrow \mathbb{Z}_p^m$.
$\mathsf{Enc}(\mathsf{MSK}, i, \mathbf{x}_i \in \mathbb{Z}^m)$**:** $s_i \leftarrow \mathbb{Z}_p$, $\mathsf{CT}_i := ([s_i], [\mathbf{x}_i + s_i\mathbf{w}_i + \mathbf{u}_i])$.
$\mathsf{KeyGen}(\mathsf{MSK}, (\mathbf{c}_1, \ldots, \mathbf{c}_n) \in \mathbb{Z}^{mn})$**:** $\mathsf{SK} := (-\sum_{i\in[n]}\langle \mathbf{c}_i, \mathbf{u}_i\rangle, \{-\mathbf{c}_i^\top \mathbf{w}_i\}_{i\in[n]})$.
$\mathsf{Dec}(\mathsf{CT}_1, \ldots, \mathsf{CT}_n, \mathsf{SK})$**:**
$\sum_{i\in[n]}(-\mathbf{c}_i^\top \mathbf{w}_i[s_i] + \mathbf{c}_i^\top[\mathbf{x}_i + s_i\mathbf{w}_i + \mathbf{u}_i]) - [\sum_{i\in[n]}\langle \mathbf{c}_i, \mathbf{u}_i\rangle] = [\sum_{i\in[n]}\langle \mathbf{c}_i, \mathbf{x}_i\rangle]$.

For intuition, we note that the ACFGU scheme may be thought of as running n instances of the ABDP scheme, where each ABDP decryption outputs the i^{th} inner product $\langle \mathbf{c}_i, \mathbf{x}_i\rangle$. Revealing each partial inner product $\langle \mathbf{c}_i, \mathbf{x}_i\rangle$ would leak too much information, so these partial decryptions are masked using $\langle \mathbf{c}_i, \mathbf{u}_i\rangle$ – this creates an extra term $\sum_{i\in[n]}\langle \mathbf{c}_i, \mathbf{u}_i\rangle$ during decryption, which, fortunately may be computed by the key generator and is compensated for by subtraction.

A First Candidate. Armed with these ideas, we construct a first candidate quadratic MIFE $\mathsf{qFE} = (\mathsf{qSetup}, \mathsf{qEnc}, \mathsf{qKeyGen}, \mathsf{qDec})$ as follows. For ease of exposition, we assume below that the dimension of each user's input vector m is set to 1.

$\mathsf{qSetup}(1^\lambda)$**:** $\mathsf{iMSK}, \mathsf{iMSK}' \leftarrow \mathsf{iSetup}(1^\lambda)$, $w_i, \widetilde{w}_i, u_i, \widetilde{u}_i \leftarrow \mathbb{Z}_p$
$\mathsf{qMSK} := (\mathsf{iMSK}, \mathsf{iMSK}', \{w_i, \widetilde{w}_i, u_i, \widetilde{u}_i\}_{i\in[n]})$.
$\mathsf{qEnc}(\mathsf{qMSK}, i, x_i \in \mathbb{Z})$**:** $s_i, \widetilde{s}_i \leftarrow \mathbb{Z}_p$
$\mathsf{iCT}'_i \leftarrow \mathsf{iEnc}(\mathsf{iMSK}', s_i)$, $\mathsf{iSK}'_i \leftarrow \mathsf{iKeyGen}(\mathsf{iMSK}', \widetilde{s}_i)$
$\mathsf{iCT}_i \leftarrow \mathsf{iEnc}(\mathsf{iMSK}, (x_i, s_i w_i, u_i))$, $\mathsf{iSK}_i \leftarrow \mathsf{iKeyGen}(\mathsf{iMSK}, (x_i, \widetilde{s}_i\widetilde{w}_i, \widetilde{u}_i))$
$\mathsf{qCT}_i := (\mathsf{iCT}'_i, \mathsf{iSK}'_i, \mathsf{iCT}_i, \mathsf{iSK}_i)$.

$\mathsf{qKeyGen}(\mathsf{MSK}, \mathbf{c}=\{c_{i,j}\}_{i,j\in[n]})$: $\mathsf{qSK} := ([-\sum_{i,j\in[n]} c_{i,j}u_i\widetilde{u}_j]_T, \{-c_{i,j}w_i\widetilde{w}_j\}_{i,j\in[n]})$.
$\mathsf{qDec}(\mathsf{qCT}_1, \ldots, \mathsf{qCT}_n, \mathsf{qSK})$:
$-\sum_{i,j\in[n]} c_{i,j}w_i\widetilde{w}_j\mathsf{iDec}(\mathsf{iCT}'_i, \mathsf{iSK}'_j) + \sum_{i,j\in[n]} c_{i,j}\mathsf{iDec}(\mathsf{iCT}_i, \mathsf{iSK}_j)$
$-[\sum_{i,j\in[n]} c_{i,j}u_i\widetilde{u}_j]_T = [\langle \mathbf{c}, \mathbf{x}\otimes\mathbf{x}\rangle]_T$

Observe that $\{\mathsf{iCT}_i, \mathsf{iSK}_i\}_{i\in[n]}$ yield $\{[x_ix_j + s_i\widetilde{s}_jw_i\widetilde{w}_j + u_i\widetilde{u}_j]_T\}_{i,j\in[n]}$ in decryption, which can be seen as ciphertexts of the n^2-input ACFGU scheme. We also remark that we decompose the ACFGU ciphertext into ciphertexts and secret keys of function-hiding IPFE so as to allow decryptors to generate ACFGU ciphertext elements for quadratic terms derived from two different users. This is in contrast to Lin's quadratic FE scheme, which uses function-hiding IPFE to compress the ciphertext size.

However, this scheme is not secure and leaks unnecessary information to the decryptor. The problem stems for the fact that the candidate scheme allows two types of mix-and-match attacks where an adversary can simultaneously use two different ciphertexts with the same index (slot) for decryption. In more detail, the adversary can learn the following information using the current scheme. Below, the superscript denotes the ciphertext index and subscript denotes the slot in a given ciphertext – thus, qCT^1_i denotes the 1^{st} ciphertext for the i^{th} slot (recall there can be multiple ciphertexts in a given slot).

1. *Attack 1:* For iCT^1_i in qCT^1_i and iSK^2_i in qCT^2_i, we have that $\mathsf{iDec}(\mathsf{iCT}^1_i, \mathsf{iSK}^2_i)$ is a valid ACFGU ciphertext and usable for the ACFGU decryption with qSK. This is problematic because it permits combining components from *different ciphertexts* qCT^1_i *and* qCT^2_i *for the same slot* i, which does not correspond to a valid combination. Recall that in an MIFE scheme, a ciphertext in slot i may be combined with multiple ciphertexts in slot $j \neq i$ but not with other ciphertexts in slot i. However, ciphertext *components* iCT^1_i and iSK^1_i from the same ciphertext and in the same slot i are allowed to be combined. Thus, to prevent this attack, we need to enforce that ciphertext components can be combined only when they come either from different slots or the same qCT_i.

2. *Attack 2:* Let $i_1 \neq i_2$. For $\{\mathsf{iCT}^1_{i_1}, \mathsf{iSK}^1_{i_1}\}$ in $\mathsf{qCT}^1_{i_1}$, $\{\mathsf{iCT}^1_{i_2}, \mathsf{iSK}^1_{i_2}\}$ in $\mathsf{qCT}^1_{i_2}$ and $\mathsf{iSK}^2_{i_2}$ in $\mathsf{qCT}^2_{i_2}$, we have that $\mathsf{iDec}(\mathsf{iCT}^1_{i_1}, \mathsf{iSK}^1_{i_1})$, $\mathsf{iDec}(\mathsf{iCT}^1_{i_1}, \mathsf{iSK}^2_{i_2})$ and $\mathsf{iDec}(\mathsf{iCT}^1_{i_2}, \mathsf{iSK}^1_{i_2})$ are valid ACFGU ciphertexts and usable for the decryption with qSK. This decryption leads to an *inconsistency* attack, where an adversary can compute a function over multiple ciphertexts for a given slot. As an example, let us consider the case where a decryptor has ciphertexts for (scalar) elements x^1_1, x^1_2, x^2_2 and a secret key for quadratic function $f = (c_{1,1}, c_{1,2}, c_{2,2})$ (w.l.o.g., we can assume $c_{2,1} = 0$). Now, the only valid function evaluations that an adversary should learn are

$$c_{1,1}x^1_1x^1_1 + c_{1,2}x^1_1x^1_2 + c_{2,2}x^1_2x^1_2, \quad \text{and} \quad c_{1,1}x^1_1x^1_1 + c_{1,2}x^1_1x^2_2 + c_{2,2}x^2_2x^2_2$$

However, the above leakage enables the adversary to additionally learn, e.g.,

$$c_{1,1}x^1_1x^1_1 + c_{1,2}x^1_1\underline{x^2_2} + c_{2,2}x^1_2\underline{x^1_2}$$

The above uses two different inputs (underlined) for the second slot for the same function evaluation, which is invalid.

More generally, valid combinations correspond to the set of superscripts (in red) $(1,1),(1,1),(1,1)$ and $(1,1),(1,2),(2,2)$. However, the adversary can learn function evaluations corresponding to $(1,1),(1,r),(s,t)$ for any $r,s,t \in [2]$ in the current candidate scheme.

Thus, both attacks leverage the decomposable structure of the quadratic ciphertext to mix and match invalid components to obtain leakage. While both attacks have the similarity that they combine different ciphertexts for the same slot in a given evaluation, the technical treatment to handle them needs to differ. This is because to address the first attack, we must prevent the attacker from combining $(1,1),(1,r),(s,t)$ for $s \neq t$ while for the second, we must prevent the same for $r \neq t$. Intuitively, r and t are the indices related to the ciphertexts of iFE while s is the index related to the secret keys of iFE, and thus prohibiting the case of $s \neq t$ and that of $r \neq t$ are essentially different things, which must be handled separately. Next, we describe how each of these attacks may be prevented.

Preventing Attack 1. Recall that Lin's quadratic FE scheme does not allow attack 1 since the encryption algorithm generates a new iMSK for each ciphertext. On the other hand, our candidate uses the same iMSK for all ciphertexts so that decryptors can generate ACFGU ciphertext elements for quadratic terms from two different users. To prevent this attack, we need a function-hiding IPFE scheme where iCT is decryptable with iSK *if and only if they come from either different slots or the same* qCT_i. Thus, we need to extend the functionality of function-hiding IPFE to check the above condition prior to computation. Although this primitive is reminiscent of "attribute-based IPFE" [5], we also need the function-hiding property which has not been considered in prior works.

To address this need, we define and construct a function-hiding "predicated IPFE" (pIPFE), which can be seen as a combination of inner product encryption [25] and IPFE. Informally, a ciphertext pCT and a secret key pSK of a pIPFE scheme pFE are associated with two vectors $\{\mathbf{x}_1, \mathbf{x}_2\}$ and $\{\mathbf{y}_1, \mathbf{y}_2\}$, respectively. Here, the secret key must hide $\mathbf{y}_2$ but do not $\mathbf{y}_1$. Decryption of pCT with pSK reveals $\langle \mathbf{x}_2, \mathbf{y}_2 \rangle$ iff $\langle \mathbf{x}_1, \mathbf{y}_1 \rangle = 0$.

To see how function-hiding predicated IPFE yields the desired functionality, let us set $\mathbf{x}_1 = (0^{2(i-1)}, 1, L, 0^{2(n-i)})$, $\mathbf{y}_1 = (0^{2(i-1)}, L, -1, 0^{2(n-i)})$ where $L \in \mathbb{Z}_p$ is sampled randomly for each encryption, and $i \in [n]$. Let (i_1, L_1) (resp. (i_2, L_2)) be a pair of a slot index and random element of $\mathbf{x}_1$ (resp. $\mathbf{y}_1$). It is easy to see that $\langle \mathbf{x}_1, \mathbf{y}_1 \rangle = 0$ iff $i_1 \neq i_2$ or $L_1 = L_2$. Since L is chosen from an exponentially large space, we have that $L_1 \neq L_2$ with overwhelming probability. We construct a function-hiding predicated IPFE scheme pFE from a function-hiding IPFE scheme iFE in a generic way. Please see Section 3 for details.

Preventing Attack 2. Attack 2 is much more tricky to handle. A problematic aspect of this attack is the fact that $\mathsf{iDec}(\mathsf{iCT}^1_{i_1}, \mathsf{iSK}^1_{i_1})$ and $\mathsf{iDec}(\mathsf{iCT}^1_{i_2}, \mathsf{iSK}^1_{i_2})$ are necessary for decryption of ciphertexts $\mathsf{qCT}^1_{i_1}, \mathsf{qCT}^1_{i_2}$ respectively, and $\mathsf{iDec}(\mathsf{iCT}^2_{i_2},$

$\mathsf{iSK}^1_{i_1}$) is necessary for combined decryption of the pair $\mathsf{qCT}^1_{i_1}, \mathsf{qCT}^2_{i_2}$. However, they leak inappropriate information if both of them are used in decryption simultaneously. Thus, we cannot solve the problem by building in some sort of access control into iFE decryption as in the case of attack 1.

Our solution is to bind ACFGU ciphertexts generated from the iFE decryption with common random elements. That is, iCT_i in qCT_i is changed to encryption of $(x_i, s_i w_i, u_i, t_i v_i)$, and iSK_i is changed to a secret key of $(x_i, \widetilde{s}_i\widetilde{w}_i, r_i\widetilde{u}_i, \widetilde{v}_i)$ where $v_i, \widetilde{v}_i$ are new elements in qMSK and r_i, t_i are the common random elements for binding ACFGU ciphertexts, which are chosen by qEnc. Then, decryption with $\{\mathsf{iCT}_i, \mathsf{iSK}_i\}_{i\in[n]}$ yields $\{[x_i x_j + s_i\widetilde{s}_j w_i\widetilde{w}_j + r_j u_i\widetilde{u}_j + t_i v_i\widetilde{v}_j]_T\}_{i,j\in[n]}$.

According to the change of $\mathsf{iCT}, \mathsf{iSK}$, the first element of an ACFGU secret key should be modified as $\mathsf{qSK}_1 = [-\sum_{i,j\in[n]} c_{i,j}(r_j u_i\widetilde{u}_j + t_i v_i\widetilde{v}_j)]_T$. By this construction, we cannot simultaneously use $\mathsf{iDec}(\mathsf{iCT}^1_{i_1}, \mathsf{iSK}^1_{i_1})$, $\mathsf{iDec}(\mathsf{iCT}^1_{i_2}, \mathsf{iSK}^1_{i_2})$ and $\mathsf{iDec}(\mathsf{iCT}^2_{i_2}, \mathsf{iSK}^1_{i_1})$ for ACFGU decryption. Intuitively, qSK_1 must involve $t^1_{i_2}$ and $t^2_{i_2}$ (randomnesses used in $\mathsf{iCT}^1_{i_2}$ and $\mathsf{iCT}^2_{i_2}$, respectively) to decrypt the ACFGU ciphertexts generated from $\mathsf{iDec}(\mathsf{iCT}^1_{i_1}, \mathsf{iSK}^1_{i_1})$, $\mathsf{iDec}(\mathsf{iCT}^1_{i_2}, \mathsf{iSK}^1_{i_2})$ and $\mathsf{iDec}(\mathsf{iCT}^2_{i_2}, \mathsf{iSK}^1_{i_1})$ together, but in fact qSK_1 can involve only one of $t^1_{i_2}$ and $t^2_{i_2}$.

How to Generate the Modified Secret Key. The last challenge is how to generate the modified secret key. It is obvious that $\mathsf{qKeyGen}$ cannot generate the modified key since it contains random elements r_i, t_i used in ciphertexts. We solve the problem by employing an additional function-hiding IP-MIFE scheme, denoted by miFE, into the candidate scheme. That is, qEnc additionally generates an IP-MIFE ciphertext miCT_i for (r_i, t_i), and $\mathsf{qKeyGen}$ generates an IP-MIFE secret key miSK for $\{(\sum_{j\in[n]} c_{j,i}u_j\widetilde{u}_i, \sum_{j\in[n]} c_{i,j}v_i\widetilde{v}_j)\}_{i\in[n]}$. Then, a decryptor can generate the secret-key element $-\sum_{i,j\in[n]} c_{i,j}(r_j u_i\widetilde{u}_j + t_i v_i\widetilde{v}_j)$ from $\mathsf{miCT}_1, \ldots, \mathsf{miCT}_n, \mathsf{miSK}$ without knowing unnecessary information. This technique is similar to Gay's technique in [21], which uses (partially) function-hiding IPFE to generate a "decryption key" consisting of both elements inherently derived from a ciphertext and a secret key. Note that our actual scheme needs mixed-group multi-input IPFE instead of IP-MIFE, which we construct in Sec. 4.

Putting it all Together. Putting together the ideas discussed above, we now present a second version of our scheme.

$\mathsf{qSetup}(1^\lambda)$**:** $\mathsf{iMSK}' \leftarrow \mathsf{iSetup}(1^\lambda), \mathsf{pMSK} \leftarrow \mathsf{pSetup}(1^\lambda), \mathsf{miMSK} \leftarrow \mathsf{miSetup}(1^\lambda)$
$w_i, \widetilde{w}_i, u_i, \widetilde{u}_i, v_i, \widetilde{v}_i \leftarrow \mathbb{Z}_p$
$\mathsf{qMSK} := (\mathsf{iMSK}', \mathsf{pMSK}, \mathsf{miMSK}, \{w_i, \widetilde{w}_i, u_i, \widetilde{u}_i, v_i, \widetilde{v}_i\}_{i\in[n]})$.
$\mathsf{qEnc}(\mathsf{qMSK}, i, x_i \in \mathbb{Z})$**:** $s_i, \widetilde{s}_i, r_i, t_i, L \leftarrow \mathbb{Z}_p$, $\boldsymbol{\ell}_1 = (0^{2(i-1)}, 1, L, 0^{2(n-i)})$
$\boldsymbol{\ell}_2 = (0^{2(i-1)}, L, -1, 0^{2(n-i)})$, $\mathsf{iCT}'_i \leftarrow \mathsf{iEnc}(\mathsf{iMSK}', s_i)$, $\mathsf{iSK}'_i \leftarrow \mathsf{iKeyGen}(\mathsf{iMSK}', \widetilde{s}_i)$
$\mathsf{pCT}_i \leftarrow \mathsf{pEnc}(\mathsf{pMSK}, \boldsymbol{\ell}_1, (x_i, s_i w_i, r_i u_i, v_i))$
$\mathsf{pSK}_i \leftarrow \mathsf{pKeyGen}(\mathsf{pMSK}, \boldsymbol{\ell}_2, (x_i, \widetilde{s}_i\widetilde{w}_i, \widetilde{u}_i, t_i\widetilde{v}_i))$
$\mathsf{miCT}_i \leftarrow \mathsf{miEnc}(\mathsf{miMSK}, (r_i, t_i)), \mathsf{qCT}_i := (\mathsf{iCT}'_i, \mathsf{iSK}'_i, \mathsf{pCT}_i, \mathsf{pSK}_i, \mathsf{miCT}_i)$.
$\mathsf{qKeyGen}(\mathsf{MSK}, \mathbf{c} = \{c_{i,j}\}_{i,j\in[n]})$**:**
$\mathsf{miSK} \leftarrow \mathsf{miKeyGen}(\mathsf{miMSK}, \{(\sum_{j\in[n]} c_{j,i}u_j\widetilde{u}_i, \sum_{j\in[n]} c_{i,j}v_i\widetilde{v}_j)\}_{i\in[n]})$
$\mathsf{qSK} := (\mathsf{miSK}, \{-c_{i,j}w_i\widetilde{w}_j\}_{i,j\in[n]})$.

$\mathsf{qDec}(\mathsf{qCT}_1, \ldots, \mathsf{qCT}_n, \mathsf{qSK})$:
$-\sum_{i,j\in[n]} c_{i,j} w_i \widetilde{w}_j \mathsf{iDec}(\mathsf{iCT}'_i, \mathsf{iSK}'_j) + \sum_{i,j\in[n]} c_{i,j} \mathsf{pDec}(\mathsf{pCT}_i, \mathsf{pSK}_j)$
$-\mathsf{miDec}(\mathsf{miCT}_1, \ldots, \mathsf{miCT}_n, \mathsf{miSK}) = [\langle \mathbf{c}, \mathbf{x} \otimes \mathbf{x} \rangle]_T$

However, while the above candidate satisfies functionality and resists the aforementioned attacks, we are still far from a proof of security. For instance, one hurdle is that we must argue that $\{w_i \widetilde{w}_j\}_{i,j\in[n]}$ is pseudorandom, which is not true because qSK contains these elements not as exponents of group elements but as elements in $\mathbb{Z}_p$. Moreover, since we have already "used up" our pairing, we cannot move these to the exponent as in [28]. Another hurdle is that the underlying IPFE schemes satisfy only indistinguishability based security rather than simulation based security. To arrive at a security proof, we must address several such challenges, which we describe next.

Overview of Proof of Security. For ease of exposition, we outline our ideas for the warm-up case of two input quadratic MIFE described in Sec. 5. The general case is handled in Sec. 6.

First, we briefly recall the definition for indistinguishability based security of secret-key MIFE. Intuitvely, security requires that all PPT adversaries cannot guess a randomly chosen bit β with meaningful probability in the following game: the adversary first outputs a set of challenge messages $\{i, x_i^{j,0}, x_i^{j,1}\}_{i\in[n],j\in[q_{\mathsf{CT}}]}$ and obtains ciphertexts for $\{i, x_i^{j,\beta}\}$. After that, the adversary can query a key generation oracle on any functions f such that for all $(j_1, \ldots, j_n) \in [q_{\mathsf{CT}}]^n$, it holds that $f(x_1^{j_1,0}, \ldots, x_n^{j_n,0}) = f(x_1^{j_1,1}, \ldots, x_n^{j_n,1})$. The goal of the security proof is to show that ciphertexts for $\{i, x_i^{j,0}\}$ and $\{i, x_i^{j,1}\}$ are indistinguishable.

The first challenge in the security proof is how to design a series of hybrids between the real games G^β for $\beta = 0$ and $\beta = 1$. A naive strategy is to change each ciphertext from $\beta = 0$ to $\beta = 1$ one by one, that is, in hybrid H_ι^η for $\iota \in [2], \eta \in [q_{\mathsf{CT}}]$, the adversary is given the ciphertext for $x_i^{j,1}$ if $(i,j) \le (\iota, \eta)$ and that for $x_i^{j,0}$ otherwise, where $(i,j) \le (\iota,\eta) \Leftrightarrow (i-1)q_{\mathsf{CT}} + j \le (\iota-1)q_{\mathsf{CT}} + \eta$. Then, we may hope to prove that $\mathsf{G}^0 \approx_c \mathsf{H}_1^1 \approx_c \cdots \approx_c \mathsf{H}_1^{q_{\mathsf{CT}}} \approx_c \mathsf{H}_2^1 \approx_c \cdots \approx_c \mathsf{H}_2^{q_{\mathsf{CT}}} \approx_c \mathsf{G}^1$. However, it quickly becomes evident that this strategy does not work. This is since the queried function f does not necessarily satisfy $f(x_1^{1,0}, x_2^{j_2,0}) = f(x_1^{1,1}, x_2^{j_2,0})$, and thus the adversary can trivially distinguish G^0 from H_1^1. Even worse, when we change some input from $\beta = 0$ to $\beta = 1$, the change affects the quadratic terms that contain an input from another slot such as $x_1^{1,1} x_2^{j_2,0}$. This correlation does not appear in IP-MIFE and makes the proof much more complex.

We address this issue as follows. Recall that our quadratic MIFE decryption first generates modified ACFGU ciphertexts $\{\mathsf{aCT}_{i,\ell}\}_{i,\ell\in[2]}$ and a secret key element aSK where

$$\mathsf{aCT}_{i,\ell} = \mathsf{pDec}(\mathsf{pCT}_i, \mathsf{pSK}_\ell) = [x_i x_\ell + s_i \widetilde{s}_\ell w_i \widetilde{w}_\ell + r_\ell u_i \widetilde{u}_\ell + t_i v_i \widetilde{v}_\ell]_T$$
$$\mathsf{aSK} = \mathsf{miDec}(\mathsf{miCT}_1, \mathsf{miCT}_2, \mathsf{miSK}) = [-\sum_{i,\ell\in[2]} c_{i,\ell}(r_\ell u_i \widetilde{u}_\ell + t_i v_i \widetilde{v}_\ell)]_T.$$

Our first idea is to define H_ι^η so that $\mathsf{qDec}(\mathsf{qCT}_1^{j_1}, \mathsf{qCT}_2^{j_2}, \mathsf{qSK})$ in H_ι^η yields $(\{\mathsf{aCT}_{i,\ell}^{j_i,j_\ell}\}_{i,\ell\in[2]}, \mathsf{aSK}^{j_1,j_2})$ where

$$\mathsf{aCT}_{i,\ell}^{j_i,j_\ell} = \begin{cases} [x_i^1 x_\ell^1 + s_i\widetilde{s}_\ell w_i\widetilde{w}_\ell + r_\ell u_i\widetilde{u}_\ell + t_i v_i\widetilde{v}_\ell]_T & (\ell, j_\ell) \le (\iota, \eta) \\ [x_i^0 x_\ell^0 + s_i\widetilde{s}_\ell w_i\widetilde{w}_\ell + r_\ell u_i\widetilde{u}_\ell + t_i v_i\widetilde{v}_\ell]_T & (\ell, j_\ell) > (\iota, \eta) \end{cases}$$

$$\mathsf{aSK}^{j_1,j_2} = [-\sum_{i,\ell\in[2]} c_{i,\ell}(r_\ell u_i\widetilde{u}_\ell + t_i v_i\widetilde{v}_\ell) - \sum_{\substack{i\in[2] \\ \ell\in\{k\in[2]\mid(k,j_k)\le(\iota,\eta)\}}} c_{i,\ell}(x_i^1 x_\ell^1 - x_i^0 x_\ell^0)]_T.$$

Note that variables $x, s, \widetilde{s}, r, t$ are also indexed by j_1, j_2, but we often omit j_1, j_2 for conciseness if it is clear in context. Observe that, in hybrid H_ι^η, $\sum_{i,\ell\in[2]} c_{i,\ell}\mathsf{aCT}_{i,\ell}^{j_i,j_\ell} + \mathsf{aSK}^{j_1,j_2} = \sum_{i,\ell\in[2]} c_{i,\ell}[x_i^0 x_\ell^0 + s_i\widetilde{s}_\ell w_i\widetilde{w}_\ell]_T$ for all $(\iota, \eta, j_1, j_2) \in [2] \times [q_{\mathsf{CT}}]^3$. Therefore, the adversary always obtains $f(x_1^0, x_2^0)$ by decryption in all hybrids and cannot trivially distinguish them. Since the second term of aSK^{j_1,j_2}, $\sum_{i,\ell\in[2]} c_{i,\ell}(x_i^1 x_\ell^1 - x_i^0 x_\ell^0) = 0$ due to the query condition, $\mathsf{H}_2^{q_{\mathsf{CT}}}$ almost can be seen as G^1. Thanks to the function-hiding property of pFE and miFE, information encoded in ciphertexts and secret keys is not revealed other than $\mathsf{aCT}_{i,\ell}, \mathsf{aSK}$.

Next we must define encoded vectors in ciphertexts and secret keys in pFE and miFE in each hybrid so that they are indistinguishable in the hybrid sequence. First, let us consider vectors encoded in pFE that yield $\mathsf{aCT}_{i,\ell}$. In G^0, recall that $\mathbf{b}_i = (x_i^0, s_i w_i, u_i, t_i v_i)$ and $\widetilde{\mathbf{b}}_i = (x_i^0, \widetilde{s}_i\widetilde{w}_i, r_i\widetilde{u}_i, \widetilde{v}_i)$ are encoded in pCT_i and pSK_i, respectively. To make $[\langle \mathbf{b}_i^{j_i}, \widetilde{\mathbf{b}}_\ell^{j_\ell}\rangle]_T = \mathsf{aCT}_{i,\ell}^{j_i,j_\ell}$ in all hybrids, we introduce a free space, used for only the security proof, and define $\mathbf{b}_i^{j_i}, \widetilde{\mathbf{b}}_i^{j_i}$ in H_ι^η as follows:

$$\mathbf{b}_i^{j_i} = (x_i^0, \underline{x_i^1}, s_i w_i, u_i, t_i v_i), \quad \widetilde{\mathbf{b}}_i^{j_i} = \begin{cases} (\underline{0}, x_i^1, \widetilde{s}_i\widetilde{w}_i, r_i\widetilde{u}_i, \widetilde{v}_i) & (i, j_i) \le (\iota, \eta) \\ (x_i^0, \underline{0}, \widetilde{s}_i\widetilde{w}_i, r_i\widetilde{u}_i, \widetilde{v}_i) & (i, j_i) > (\iota, \eta) \end{cases}.$$

Then, we need to prove that $\{\mathbf{b}_i^{j_i}, \widetilde{\mathbf{b}}_i^{j_i}\}_{i\in[2],j_i\in[q_{\mathsf{CT}}]}$ in $\mathsf{H}_\iota^{\eta-1}$ and that in H_ι^η are indistinguishable. Initially, it appears that we can prove it similarly to Lin's technique [28], that is, we introduce a more free space and consider an intermediate hybrid in which we define

$$\mathbf{b}_i^{j_i} = (x_i^{j_i,0}, x_i^{j_i,1}, s_i w_i, u_i, t_i v_i, \underline{x_i^{j_i,0} x_\iota^{\eta,0} + s_i\widetilde{s}_\iota w_i\widetilde{w}_\iota + r_\iota u_i\widetilde{u}_\iota + t_i v_i\widetilde{v}_\iota}) \tag{1.1}$$

$$\widetilde{\mathbf{b}}_i^{j_i} = \begin{cases} (0, x_i^{j_i,1}, \widetilde{s}_i\widetilde{w}_i, r_i\widetilde{u}_i, \widetilde{v}_i, \underline{0}) & (i, j_i) < (\iota, \eta) \\ (\underline{0, 0, 0, 0, 0, 1}) & (i, j_i) = (\iota, \eta) \\ (x_i^{j_i,0}, 0, \widetilde{s}_i\widetilde{w}_i, r_i\widetilde{u}_i, \widetilde{v}_i, \underline{0}) & (i, j_i) > (\iota, \eta) \end{cases}$$

Now, we may hope to change $x_i^{j_i,0} x_\iota^{\eta,0}$ in the last entry of $\mathbf{b}_i^{j_i}$ to $x_i^{j_i,1} x_\iota^{\eta,1}$ by the indistinguishability-based security of the (modified) ACFGU IP-MIFE scheme.

However, we get stuck here; the relation between $\{x_i^{j_i,0} x_\iota^{\eta,0}\}_{i\in[2],j_i\in[q_{\mathsf{CT}}]}$ and $\{x_i^{j_i,1} x_\iota^{\eta,1}\}_{i\in[2],j_i\in[q_{\mathsf{CT}}]}$ implied by the query condition $f(x_1^{j_1,0}, x_2^{j_2,0}) = f(x_1^{j_1,1}, x_2^{j_2,1})$ is unclear. This is because, in the reduction to ACFGU IP-MIFE, the simulator is expected to simulate pCT for $\mathbf{b}_i^{j_i}$ and qSK for quadratic function f using

ACFGU ciphertexts for $\{x_i^{j_i,\beta} x_\iota^{\eta,\beta}\}_{i\in[2],j_i\in[q_{\mathsf{CT}}]}$ and secret keys for linear functions f_ι, respectively, such that $f_\iota(x_1^{j_1,0}x_\iota^{\eta,0}, x_2^{j_2,0}x_\iota^{\eta,0}) = f_\iota(x_1^{j_1,1}x_\iota^{\eta,1}, x_2^{j_2,1}x_\iota^{\eta,1})$. Note that f_ι comprises coefficients of f that are related to the ι-th input. Unfortunately, we cannot derive the above relation on f_ι from the query condition. The critical observation we make here is that we have an alternative equality on f_ι that are implied by the condition: for all $(j_1, j_2, \eta) \in [q_{\mathsf{CT}}]^3$, we have

$$f_1(x_1^{\eta,0}x_1^{\eta,0} - x_1^{1,0}x_1^{1,0}, x_2^{j_2,0}x_1^{\eta,0} - x_2^{j_2,0}x_1^{1,0}) = f_1(x_1^{\eta,1}x_1^{\eta,1} - x_1^{1,1}x_1^{1,1}, x_2^{j_2,1}x_1^{\eta,1} - x_2^{j_2,1}x_1^{1,1}) \tag{1.2}$$

$$f_2(x_1^{j_1,0}x_2^{\eta,0} - x_1^{j_1,0}x_2^{1,0}, x_2^{\eta,0}x_2^{\eta,0} - x_2^{1,0}x_2^{1,0}) = f_2(x_1^{j_1,1}x_2^{\eta,1} - x_1^{j_1,1}x_2^{1,1}, x_2^{\eta,1}x_2^{\eta,1} - x_2^{1,1}x_2^{1,1}). \tag{1.3}$$

Eq. (1.2) and (1.3) can be obtained by Eq. (1.4) − Eq. (1.5) where

$$f(x_1^{\eta,0}, x_2^{j_2,0}) = f(x_1^{\eta,1}, x_2^{j_2,1}) \qquad f(x_1^{j_1,0}, x_2^{\eta,0}) = f(x_1^{j_1,1}, x_2^{\eta,1}) \tag{1.4}$$

$$f(x_1^{1,0}, x_2^{j_2,0}) = f(x_1^{1,1}, x_2^{j_2,1}) \qquad f(x_1^{j_1,0}, x_2^{1,0}) = f(x_1^{j_1,1}, x_2^{1,1}). \tag{1.5}$$

The last challenge is to somehow change $x_i^{j_i,0}x_\iota^{\eta,0}$ in the last entry of Eq. (1.1) in to $x_i^{j_i,1}x_\iota^{\eta,1}$ leveraging Eq. (1.2) or Eq. (1.3). We first observe that

$$\begin{aligned} x_i^{j_i,0}x_\iota^{\eta,0} + s_i^{j_i}\widetilde{s}_\iota^{j_\iota} w_i\widetilde{w}_\iota + r_\iota^{j_\iota}u_i\widetilde{u}_\iota + t_i^{j_i}v_i\widetilde{v}_\iota &\approx_c x_i^{j_i,0}x_\iota^{\eta,0} + \widehat{s}_{i,\iota}^{j_i}\widehat{w}_{i,\iota} + \widehat{u}_i + \widehat{v}_i^{j_i} \\ &= \underbrace{x_i^{j_i,0}x_\iota^{\eta,0} - x_i^{j_i,0}x_\iota^{1,0} + \widehat{s}_{i,\iota}^{j_i}\widehat{w}_{i,\iota} + \widehat{u}_i}_{\text{ACFGU ciphertext}} + \ddot{v}_i^{j_i} \end{aligned}$$

where $\widehat{s}_{i,\iota}^{j_i}, \widehat{w}_{i,\iota}, \widehat{u}_i, \widehat{v}_i^{j_i}, \ddot{v}_i^{j_i}$ are fresh random elements. The computational indistinguishability is implied by the SXDH assumption, and the equality follows by implicitly defining $\widehat{v}_i^{j_i} = \ddot{v}_i^{j_i} - x_i^{j_i,0}x_\iota^{1,0}$. We can see that the last part of the above equation is exactly the ACFGU ciphertext of $x_i^{j_i,0}x_\iota^{\eta,0} - x_i^{j_i,0}x_\iota^{1,0}$ plus $\ddot{v}_i^{j_i}$. At this point, we can use the security of the ACFGU IP-MIFE scheme to change $x_i^{j_i,0}x_\iota^{\eta,0} - x_i^{j_i,0}x_\iota^{1,0}$ to $x_i^{j_i,1}x_\iota^{\eta,1} - x_i^{j_i,1}x_\iota^{1,1}$. This is because they satisfy Eq. (1.2) or Eq. (1.3), and thus the reduction can follow the query condition of IP-MIFE. Perceptive readers may notice that if $i = \iota$, then $x_i^{j_i,0}x_\iota^{\eta,0} - x_i^{j_i,0}x_\iota^{1,0} = x_i^{j_i,1}x_\iota^{\eta,1} - x_i^{j_i,1}x_\iota^{1,1}$ holds only when $j_i = \eta$. This is not a problem since we can deal with the terms for $i = \iota, j_i \neq \eta$ leveraging the security of predicated IPFE.

Next we give some intuition for how to define vectors in miFE. Similarly to $\mathbf{b}_i^{j_i}, \widetilde{\mathbf{b}}_i^{j_i}$, we want to define $\mathbf{f}_i^{j_i}, \widetilde{\mathbf{f}}_i$ in H_ι^η, which are encoded in miFE and yield aSK, but this approach quickly runs into cumbersome issues. The first problem is that the second term of aSK^{j_1,j_2}, $\mathsf{aSK}^{j_1,j_2}[2] = \sum c_{i,\ell}(x_i^{j_i,1}x_\ell^{j_\ell,1} - x_i^{j_i,0}x_\ell^{j_\ell,0})$, in the current definition depends on both $x_1^{j_1}$ and $x_2^{j_2}$. Thus, we must somehow encode $x_1^{j_1}$ and $x_2^{j_2}$ in $\mathsf{miCT}_1^{j_1}$ and $\mathsf{miCT}_2^{j_2}$, respectively. However, we cannot generate the term $x_1^{j_1}x_2^{j_2}$ via miFE, which can only compute linear functions! A naive idea may be to program all quadratic terms into additional free spaces in miCT. It immediately ends in failure; we cannot program q_{CT}^2 values into $O(q_{\mathsf{CT}})$ spaces.

Our solution is to use Eq. (1.2) and Eq. (1.3) to compress the q_{CT}^2 values into q_{CT} values. For instance, Eq. (1.2) implies

$$f_1(x_1^{j_1,1}x_1^{j_1,1} - x_1^{j_1,0}x_1^{j_1,0}, x_2^{j_2,1}x_1^{j_1,1} - x_2^{j_2,0}x_1^{j_1,0}) = f_1(x_1^{1,1}x_1^{1,1} - x_1^{1,0}x_1^{1,0}, x_2^{j_2,1}x_1^{1,1} - x_2^{j_2,0}x_1^{1,0})$$

since f_1 is a linear function (we change η to j_1). This means that $\sum_{\ell=1} c_{i,\ell}(x_i^{j_i,1}x_\ell^{j_\ell,1} - x_i^{j_i,0}x_\ell^{j_\ell,0}) = \sum_{\ell=1} c_{i,\ell}(x_i^{j_i,1}x_\ell^{1,1} - x_i^{j_i,0}x_\ell^{1,0})$ for all j_i. Similarly, we can handle the case for $\ell = 2$. Thus, we can program $\mathsf{aSK}^{j_1,j_2}[2]$ in $\mathsf{miCT}_1^{j_1}$ and $\mathsf{miCT}_2^{j_2}$ as:

$$\mathbf{f}_i^{j_i} = \begin{cases} (r_i, t_i, \underline{x_i^{j_i,1}x_1^{1,1} - x_i^{j_i,0}x_1^{1,0}}, 0) & \iota = 1 \\ (r_i, t_i, \underline{x_i^{j_i,1}x_1^{1,1} - x_i^{j_i,0}x_1^{1,0}, x_i^{j_i,1}x_2^{1,1} - x_i^{j_i,0}x_2^{1,0}}) & \iota = 2 \end{cases}$$
$$\widetilde{\mathbf{f}}_i = (\sum_{\ell\in[2]} c_{\ell,i}u_\ell\widetilde{u}_i, \sum_{\ell\in[2]} c_{i,\ell}v_i\widetilde{v}_\ell, \underline{c_{i,1}, c_{i,2}}).$$

The second problem is the fact that

$$\overline{\mathsf{aSK}^{j_1,j_2}}[2] = \langle \mathbf{f}_i^{j_i}, \widetilde{\mathbf{f}}_i\rangle - \sum_{i,\ell\in[2]} c_{i,\ell}(r_\ell u_i\widetilde{u}_\ell + t_i v_i\widetilde{v}_\ell) = \sum_{i\in[2],\ell\in[\iota]} c_{i,\ell}(x_i^1x_\ell^1 - x_i^0x_\ell^0)$$

in the current definition of $\mathbf{f}_i^{j_i}, \widetilde{\mathbf{f}}_i$, while $\mathsf{aSK}^{j_1,j_2}[2]$ should be

$$\mathsf{aSK}^{j_1,j_2}[2] = \sum_{\substack{i\in[2] \\ \ell\in\{k\in[2]\mid(k,j_k)\le(\iota,\eta)\}}} c_{i,\ell}(x_i^1x_\ell^1 - x_i^0x_\ell^0).$$

We adjust them by modifying aCT as $\overline{\mathsf{aCT}_{i,\ell}^{j_i,j_\ell}} = \mathsf{aCT}_{i,\ell}^{j_i,j_\ell} + Q(\mathbf{x})$ so that $\sum_{i,\ell\in[2]} c_{i,\ell}\overline{\mathsf{aCT}_{i,\ell}^{j_i,j_\ell}} + \overline{\mathsf{aSK}^{j_1,j_2}} = \sum_{i,\ell\in[2]} c_{i,\ell}[x_i^0x_\ell^0 + s_i\widetilde{s}_\ell w_i\widetilde{w}_\ell]_T$ holds, where Q is a quadratic polynomial over variables $\mathbf{x} = \{x_i^{j_i,\beta}\}_{i\in[2],j_i\in[q_{\mathsf{CT}}],\beta\in\{0,1\}}$. The additional term $Q(\mathbf{x})$ in $\overline{\mathsf{aCT}_{i,\ell}^{j_i,j_\ell}}$ can be programed into pCT and pSK by introducing more additional space. Please see Section 5 for a detailed argument.

2 Preliminaries

In this section, we define some notation and preliminaries that we require. For vectors $\mathbf{v}_1, \ldots, \mathbf{v}_n$, $(\mathbf{v}_1, \ldots, \mathbf{v}_n)$ denotes the vector concatenation as row vectors *regardless of* whether each $\mathbf{v}_i$ is a row or column vector. We use $\otimes$ for the Kronecker product. We denote an n-dimensional unit vector $(0^{i-1}, 1, 0^{n-1})$ by $\mathbf{e}_{i/n}$. We use standard cryptographic bilinear groups where the matrix decisional Diffie-Hellman assumption (MDDH) holds [18].

2.1 Multi-Input Functional Encryption

Definition 2.1 (Multi-Input Functional Encryption). Let $\mathcal{F}$ be a function family such that, for all $f \in \mathcal{F}$, $f : \mathcal{X}_1 \times \cdots \times \mathcal{X}_n \to \mathcal{Z}$. An MIFE scheme for $\mathcal{F}$, MIFE, consists of four algorithms.

$\mathsf{Setup}(1^\lambda)$**:** It takes a security parameter 1^λ and outputs a public parameter PP and a master secret key MSK. The other algorithms implicitly take PP.

$\mathsf{Enc}(\mathsf{MSK}, i, x_i)$**:** It takes MSK, an index $i \in [n]$, and $x_i \in \mathcal{X}_i$ and outputs a ciphertext CT_i.

$\mathsf{KeyGen}(\mathsf{MSK}, f)$**:** It takes MSK, and $f \in \mathcal{F}$, and outputs a secret key SK.

$\mathsf{Dec}(\mathsf{CT}_1, \ldots, \mathsf{CT}_n, \mathsf{SK})$**:** It takes $\mathsf{CT}_1, \ldots, \mathsf{CT}_n$ and SK, and outputs a decryption value $d \in \mathcal{Z}$ or a symbol $\perp$.

When $n = 1$, we call it just a functional encryption (FE) scheme and omit the second argument of Enc.

Correctness. MIFE is *correct* if it satisfies the following condition. For all $\lambda \in \mathbb{N}$, $(x_1, \ldots, x_n) \in \mathcal{X}_1 \times \cdots \times \mathcal{X}_n$, $f \in \mathcal{F}$, we have

$$\Pr\left[d = f(x_1, \ldots, x_n) \;\middle|\; \begin{array}{l} \mathsf{PP}, \mathsf{MSK} \leftarrow \mathsf{Setup}(1^\lambda) \\ \mathsf{CT}_i \leftarrow \mathsf{Enc}(\mathsf{MSK}, i, x_i) \\ \mathsf{SK} \leftarrow \mathsf{KeyGen}(\mathsf{MSK}, f) \\ d := \mathsf{Dec}(\mathsf{CT}_1, \ldots, , \mathsf{CT}_n, \mathsf{SK}) \end{array} \right] = 1.$$

Selective Security. We define two indistinguishability-based security definitions for MIFE, namely, message-hiding and function-hiding. For a stateful PPT adversary $\mathcal{A}$ and $\lambda \in \mathbb{N}$, let

$$\mathsf{P}^{\mathsf{MIFE},\beta}_{\mathcal{A},\mathsf{mh}}(\lambda) := \Pr\left[\beta' = 1 \;\middle|\; \begin{array}{l} \{i, x_i^{j,0}, x_i^{j,1}\}_{i\in[n], j\in[q_{\mathsf{CT},i}]} \leftarrow \mathcal{A}(1^\lambda) \\ \mathsf{PP}, \mathsf{MSK} \leftarrow \mathsf{Setup}(1^\lambda), \\ \mathsf{CT}_i^j \leftarrow \mathsf{Enc}(\mathsf{MSK}, i, x_i^{j,\beta}) \\ \beta' \leftarrow \mathcal{A}^{\mathsf{KeyGen}(\mathsf{MSK},\cdot)}(\mathsf{PP}, \{\mathsf{CT}_i^j\}_{i\in[n], j\in[q_{\mathsf{CT},i}]}) \end{array} \right].$$

Let q_{SK} be a number of queries to KeyGen. We say $\mathcal{A}$ is *admissible* if, in case of $q_{\mathsf{CT},1}, \ldots, q_{\mathsf{CT},n}, q_{\mathsf{SK}} \geq 1$, $\mathcal{A}$'s queries satisfy $f^\ell(x_1^{j_1,0}, \ldots, x_n^{j_n,0}) = f^\ell(x_1^{j_1,1}, \ldots, x_n^{j_n,1})$ for all $(j_1, \ldots, j_n) \in [q_{\mathsf{CT},1}] \times \cdots \times [q_{\mathsf{CT},n}]$ and $\ell \in [q_{\mathsf{SK}}]$. MIFE is *message-hiding* if, for all admissible PPT adversaries $\mathcal{A}$, the following advantage of $\mathcal{A}$ is negligible in λ: $\mathsf{Adv}^{\mathsf{MIFE}}_{\mathcal{A},\mathsf{mh}}(\lambda) := |\mathsf{P}^{\mathsf{MIFE},0}_{\mathcal{A},\mathsf{mh}}(\lambda) - \mathsf{P}^{\mathsf{MIFE},1}_{\mathcal{A},\mathsf{mh}}(\lambda)|$.

Next, we define a function-hiding property. Let $\mathsf{P}^{\mathsf{MIFE},\beta}_{\mathcal{A},\mathsf{fh}}(\lambda)$ be defined the same as $\mathsf{P}^{\mathsf{MIFE},\beta}_{\mathcal{A},\mathsf{mh}}(\lambda)$ except that $\mathcal{A}$'s oracle is $\mathcal{O}_{\mathsf{SK}}(\beta, \cdot)$ instead of KeyGen, where $\mathcal{O}_{\mathsf{SK}}(\beta, \cdot)$ takes (f^0, f^1) and outputs $\mathsf{KeyGen}(\mathsf{MSK}, f^\beta)$. This time, $\mathcal{A}$ is *admissible* if, in case of $q_{\mathsf{CT},1}, \ldots, q_{\mathsf{CT},n}, q_{\mathsf{SK}} \geq 1$, $\mathcal{A}$'s queries satisfy $f^{\ell,0}(x_1^{j_1,0}, \ldots, x_n^{j_n,0}) = f^{\ell,1}(x_1^{j_1,1}, \ldots, x_n^{j_n,1})$ for all $(j_1, \ldots, j_n) \in [q_{\mathsf{CT},1}] \times \cdots \times [q_{\mathsf{CT},n}]$ and $\ell \in [q_{\mathsf{SK}}]$. Then, MIFE is *function-hiding* if, for all admissible PPT adversaries $\mathcal{A}$, the following advantage of $\mathcal{A}$ is negligible in λ: $\mathsf{Adv}^{\mathsf{MIFE}}_{\mathcal{A},\mathsf{fh}}(\lambda) := |\mathsf{P}^{\mathsf{MIFE},0}_{\mathcal{A},\mathsf{fh}}(\lambda) - \mathsf{P}^{\mathsf{MIFE},1}_{\mathcal{A},\mathsf{fh}}(\lambda)|$.

Remark 2.1. In this paper, we assume that $q_{\mathsf{CT},i} \geq 1$ for all $i \in [n]$ and that $q_{\mathsf{CT},1} = \cdots = q_{\mathsf{CT},n} (= q_{\mathsf{CT}})$. This is w.l.o.g as discussed in [6, 17].

We next define quadratic functions.

Definition 2.2 (Bounded-Norm Multi-Input Quadratic functions over $\mathbb{Z}$). A function family $\mathcal{F}^{\mathsf{MQF}}_{m,n,X,C}$ for bounded-norm multi-input quadratic functions consist of functions $f : (\mathcal{X}^m)^n \to \mathbb{Z}$ where $\mathcal{X} = \{i \mid i \in \mathbb{Z}, |i| \leq X\}$. Each $f \in \mathcal{F}^{\mathsf{MQF}}_{m,n,X,C}$ is specified by $\mathbf{c} = \{c_{\mu,\nu}\}_{\mu,\nu\in[mn]} \in \mathbb{Z}^{(mn)^2}$ s.t. $||\mathbf{c}||_\infty \leq C$ and $c_{\mu,\nu} = 0$ if $\mu > \nu$. Let x_μ be the μ-th element of $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_n) \in (\mathcal{X}^m)^n$. Then, f specified by $\mathbf{c}$ is defined as $f(\mathbf{x}_1, \ldots, \mathbf{x}_n) := \sum_{\mu,\nu\in[mn]} c_{\mu,\nu} x_\mu x_\nu$.

3 Predicated Inner Product Functional Encryption

We define and construct predicated inner product functional encryption.

Definition 3.1 (Inner Products over Bilinear Groups). Let $\mathbb{G} = (p, G_1, G_2, G_T, g_1, g_2, e)$ be bilinear groups. A function family $\mathcal{F}^{\mathsf{IP}}_{m,\mathbb{G}}$ for inner products over bilinear groups consists of functions $f : G_1^m \to G_T$. Each $f \in \mathcal{F}^{\mathsf{IP}}_{m,\mathbb{G}}$ is specified by $[\mathbf{y}]_2$ where $\mathbf{y} \in \mathbb{Z}_p^m$ and defined as $f([\mathbf{x}]_1) := [\langle \mathbf{x}, \mathbf{y} \rangle]_T$.

Definition 3.2 (Predicated Inner Products over Bilinear Groups). A function family $\mathcal{F}^{\mathsf{PIP}}_{d,m,\mathbb{G}}$ for predicated inner products over bilinear groups consists of functions $f : \mathbb{Z}_p^d \times G_1^m \to G_T \cup \{\bot\}$. Each $f \in \mathcal{F}^{\mathsf{PIP}}_{d,m,\mathbb{G}}$ is specified by $\mathbf{y}_1 \in \mathbb{Z}_p^d$ and $[\mathbf{y}_2]_2$ where $\mathbf{y}_2 \in \mathbb{Z}_p^m$ and defined as $f(\mathbf{x}_1, [\mathbf{x}_2]_1) := \begin{cases} [\langle \mathbf{x}_2, \mathbf{y}_2 \rangle]_T & \text{if } \langle \mathbf{x}_1, \mathbf{y}_1 \rangle = 0 \\ \bot & \text{if } \langle \mathbf{x}_1, \mathbf{y}_1 \rangle \neq 0 \end{cases}$.

We refer to FE for $\mathcal{F}^{\mathsf{IP}}_{m,\mathbb{G}}$ and $\mathcal{F}^{\mathsf{PIP}}_{d,m,\mathbb{G}}$ as IPFE and predicated IPFE, respectively. We define partially function-hiding security of FE for $\mathcal{F}^{\mathsf{PIP}}_{d,m,\mathbb{G}}$. Partially function-hiding security guarantees that secret keys hide $\mathbf{y}_2$ (but do not $\mathbf{y}_1$).

Partially Function-Hiding Security. Let $\mathsf{pFE} = (\mathsf{pSetup}, \mathsf{pEnc}, \mathsf{pKeyGen}, \mathsf{pDec})$ be a FE scheme for $\mathcal{F}^{\mathsf{PIP}}_{d,m,\mathbb{G}}$. For a stateful PPT adversary $\mathcal{A}$ and $\lambda \in \mathbb{N}$, let

$$\mathsf{P}^{\mathsf{pFE},\beta}_{\mathcal{A},\mathsf{pfh}}(\lambda) := \Pr\left[\beta' = 1 \;\middle|\; \begin{array}{l} \{\mathbf{x}_1^j, [\mathbf{x}_2^{j,0}]_1, [\mathbf{x}_2^{j,1}]_1\}_{j \in [q_{\mathsf{CT}}]} \leftarrow \mathcal{A}(1^\lambda) \\ \mathsf{pPP}, \mathsf{pMSK} \leftarrow \mathsf{pSetup}(1^\lambda), \\ \mathsf{pCT}^j \leftarrow \mathsf{pEnc}(\mathsf{pMSK}, (\mathbf{x}_1^j, [\mathbf{x}_2^{j,\beta}]_1)) \\ \beta' \leftarrow \mathcal{A}^{\mathcal{O}_{\mathsf{SK}}(\beta,\cdot)}(\mathsf{pPP}, \{\mathsf{pCT}^j\}_{j \in [q_{\mathsf{CT}}]}) \end{array}\right]$$

where $\mathcal{O}_{\mathsf{SK}}$ takes $(\mathbf{y}_1, [\mathbf{y}_2^0]_2, [\mathbf{y}_2^1]_2)$ and outputs $\mathsf{pKeyGen}(\mathsf{MSK}, (\mathbf{y}_1, [\mathbf{y}_2^\beta]_2))$. Let q_{SK} be a number of queries to $\mathcal{O}_{\mathsf{SK}}$. We say $\mathcal{A}$ is *admissible* if $\mathcal{A}$'s queries satisfy $\langle \mathbf{x}_2^{j,0}, \mathbf{y}_2^{\ell,0} \rangle = \langle \mathbf{x}_2^{j,1}, \mathbf{y}_2^{\ell,1} \rangle$ *when* $\langle \mathbf{x}_1^j, \mathbf{y}_1^\ell \rangle = 0$ for all $j \in [q_{\mathsf{CT}}]$ and $\ell \in [q_{\mathsf{SK}}]$. pFE is *partially function-hiding* if, for all admissible PPT adversaries $\mathcal{A}$, the following advantage of $\mathcal{A}$ is negligible in λ: $\mathsf{Adv}^{\mathsf{pFE}}_{\mathcal{A},\mathsf{pfh}}(\lambda) := |\mathsf{P}^{\mathsf{pFE},0}_{\mathcal{A},\mathsf{pfh}}(\lambda) - \mathsf{P}^{\mathsf{pFE},1}_{\mathcal{A},\mathsf{pfh}}(\lambda)|$.

3.1 Predicated IPFE from IPFE

We construct a partially function-hiding FE scheme for $\mathcal{F}^{\mathsf{PIP}}_{d,m,\mathbb{G}}$ from a function-hiding FE scheme for $\mathcal{F}^{\mathsf{IP}}_{kd+2m+1,\mathbb{G}}$ generically. Note that k is a parameter for the MDDH assumption. A function-hiding FE scheme for $\mathcal{F}^{\mathsf{IP}}_{m,\mathbb{G}}$ based on MDDH is implied by the function-hiding IPFE scheme described in [30, Appx. A] [7]. Let $\mathsf{iFE} = (\mathsf{iSetup}, \mathsf{iEnc}, \mathsf{iKeyGen}, \mathsf{iDec})$ be a function-hiding FE scheme for $\mathcal{F}^{\mathsf{IP}}_{kd+2m+1,\mathbb{G}}$. Then, our partially function-hiding FE scheme $\mathsf{pFE} = (\mathsf{pSetup}, \mathsf{pEnc}, \mathsf{pKeyGen}, \mathsf{pDec})$ for $\mathcal{F}^{\mathsf{PIP}}_{d,m,\mathbb{G}}$ is constructed as shown in Fig 1.

[7] In more detail, this follows since the scheme remains correct and secure even if input vectors for Enc and KeyGen consist of group elements, and Dec first obtains decryption values on the exponent of a target-group generator and then computes its discrete log.

$\underline{\mathsf{pSetup}(1^\lambda)} \to \mathsf{pPP}, \mathsf{pMSK}$
$(\mathsf{pPP}, \mathsf{pMSK}) := (\mathsf{iPP}, \mathsf{iMSK}) \leftarrow \mathsf{iSetup}(1^\lambda)$

$\underline{\mathsf{pEnc}(\mathsf{MSK}, (\mathbf{x}_1, [\mathbf{x}_2]_1))} \to \mathsf{pCT}$
$\mathbf{z} \leftarrow \mathbb{Z}_p^k,\ \mathbf{x} := (\mathbf{z} \otimes \mathbf{x}_1, \mathbf{x}_2, 0^m, 0) \in \mathbb{Z}_p^{kd+2m+1},\ \mathsf{iCT} \leftarrow \mathsf{iEnc}(\mathsf{iMSK}, [\mathbf{x}]_1),\ \mathsf{pCT} := (\mathbf{x}_1, \mathsf{iCT})$

$\underline{\mathsf{pKeyGen}(\mathsf{pMSK}, (\mathbf{y}_1, [\mathbf{y}_2]_2))} \to \mathsf{pSK}$
$\mathbf{a} \leftarrow \mathbb{Z}_p^k,\ \mathbf{y} := (\mathbf{a} \otimes \mathbf{y}_1, \mathbf{y}_2, 0^m, 0) \in \mathbb{Z}_p^{kd+2m+1},\ \mathsf{iSK} \leftarrow \mathsf{iKeyGen}(\mathsf{iMSK}, [\mathbf{y}]_2),\ \mathsf{pSK} := (\mathbf{y}_1, \mathsf{iSK})$

$\underline{\mathsf{pDec}(\mathsf{pCTpSK})} \to z$
If $\langle \mathbf{x}_1, \mathbf{y}_1 \rangle \neq 0$, outputs $z = \perp$. Otherwise, outputs $z = \mathsf{iDec}(\mathsf{iCT}, \mathsf{iSK})$.

Fig 1: Our predicated IPFE scheme.

Correctness. Since $\langle \mathbf{z} \otimes \mathbf{x}_1, \mathbf{a} \otimes \mathbf{y}_1 \rangle = \langle \mathbf{z}, \mathbf{a} \rangle \cdot \langle \mathbf{x}_1, \mathbf{y}_1 \rangle$, $\mathsf{iDec}(\mathsf{iCT}, \mathsf{iSK})$ outputs $[\langle \mathbf{x}, \mathbf{y} \rangle]_T = [\langle \mathbf{x}_2, \mathbf{y}_2 \rangle]_T$ if $\langle \mathbf{x}_1, \mathbf{y}_1 \rangle = 0$. This follows from the correctness of iFE.

For security, we have the following theorem.

Theorem 3.1. *If* iFE *is function-hiding, and the MDDH assumption holds in* $\mathbb{G}$*, then* pFE *is partially function-hiding. More precisely, for all PPT adversaries* $\mathcal{A}$*, there exist PPT adversaries* $\mathcal{B}_1, \mathcal{B}_2$ *such that*

$$\mathsf{Adv}^{\mathsf{pFE}}_{\mathcal{A},\mathsf{pfh}}(\lambda) \leq q_{\mathsf{CT}}(3\mathsf{Adv}^{\mathsf{iFE}}_{\mathcal{B}_1,\mathsf{fh}}(\lambda) + 2\mathsf{Adv}^{\mathcal{D}_k\text{-}\mathsf{MDDH}}_{\mathcal{B}_2}(\lambda)).$$

Due to space constraints, the proof is provided in the full version.

4 Mixed-Group Multi-Input IPFE

In this section, we define and construct our mixed-group multi-input inner product functional encryption (mixed-group IP-MIFE).

Definition 4.1 (Multi-Input Inner Products over Bilinear Groups). Let $\mathbb{G} = (p, G_1, G_2, G_T, g_1, g_2, e)$ be bilinear groups. A function family $\mathcal{F}^{\mathsf{MIP}}_{m,n,\mathbb{G}}$ for multi-input inner products over bilinear groups consists of functions $f : (G_1^m)^n \to G_T$. Each $f \in \mathcal{F}^{\mathsf{MIP}}_{m,n,\mathbb{G}}$ is specified by $[\mathbf{y}_1]_2, \ldots, [\mathbf{y}_n]_2$ where $\mathbf{y}_i \in \mathbb{Z}_p^m$ and defined as $f([\mathbf{x}]_1, \ldots, [\mathbf{x}]_n) := [\sum_{i \in [n]} \langle \mathbf{x}_i, \mathbf{y}_i \rangle]_T$.

Definition 4.2 (Multi-Input Mixed-Group Inner Products over Bilinear Groups). Let $\mathbb{G} = (p, G_1, G_2, G_T, g_1, g_2, e)$ be bilinear groups. A function family $\mathcal{F}^{\mathsf{MGIP}}_{m_1,m_2,n,\mathbb{G}}$ for multi-input mixed-group inner products over bilinear groups consists of functions $f : (G_1^{m_1} \times G_2^{m_2})^n \to G_T$. Each $f \in \mathcal{F}^{\mathsf{MGIP}}_{m_1,m_2,n,\mathbb{G}}$ is specified by $([\mathbf{y}_{1,1}]_2, [\mathbf{y}_{1,2}]_1, \ldots, [\mathbf{y}_{n,1}]_2, [\mathbf{y}_{n,2}]_1)$ where $\mathbf{y}_{i,1} \in \mathbb{Z}_p^{m_1}$ and $\mathbf{y}_{i,2} \in \mathbb{Z}_p^{m_2}$ and defined as $f(([\mathbf{x}_{1,1}]_1, [\mathbf{x}_{1,2}]_2), \ldots, ([\mathbf{x}_{n,1}]_1, [\mathbf{x}_{n,2}]_2)) := [\langle \mathbf{x}, \mathbf{y} \rangle]_T$ where $\mathbf{x} := (\mathbf{x}_{1,1}, \mathbf{x}_{1,2}, \ldots, \mathbf{x}_{n,1}, \mathbf{x}_{n,2})$ and $\mathbf{y} := (\mathbf{y}_{1,1}, \mathbf{y}_{1,2}, \ldots, \mathbf{y}_{n,1}, \mathbf{y}_{n,2})$.

We refer to MIFE for $\mathcal{F}^{\mathsf{MIP}}_{m,n,\mathbb{G}}$ and $\mathcal{F}^{\mathsf{MGIP}}_{m_1,m_2,n,\mathbb{G}}$ as IP-MIFE and mixed-group IP-MIFE, respectively. We require mixed-group IP-MIFE to satisfy the standard function-hiding security in Def. 2.1.

$\underline{\mathsf{gSetup}(1^\lambda)} \to \mathsf{gPP}, \mathsf{gMSK}$
$\mathsf{miPP}, \mathsf{miMSK} \leftarrow \mathsf{miSetup}(1^\lambda),\ (\mathsf{iPP}_1, \mathsf{iMSK}_1), \ldots, (\mathsf{iPP}_n, \mathsf{iMSK}_n) \leftarrow \mathsf{iSetup}(1^\lambda)$
$\mathsf{gPP} := (\mathsf{miPP}, \mathsf{iPP}_1, \ldots, \mathsf{iPP}_n),\ \mathsf{gMSK} := (\mathsf{miMSK}, \mathsf{iMSK}_1, \ldots, \mathsf{iMSK}_n)$

$\underline{\mathsf{gEnc}(\mathsf{MSK}, i, ([\mathbf{x}_{i,1}]_1, [\mathbf{x}_{i,2}]_2))} \to \mathsf{gCT}_i$
$\mathbf{z} \leftarrow \mathbb{Z}_p^k,\ \widetilde{\mathbf{x}}_{i,1} := (\mathbf{x}_{i,1}, 0^{m_2}, \mathbf{z}, 0) \in \mathbb{Z}_p^{m_1+m_2+k+1},\ \widetilde{\mathbf{x}}_{i,2} := (\mathbf{x}_{i,2}, -\mathbf{z}, 0) \in \mathbb{Z}_p^{m_2+k+1}$
$\mathsf{miCT}_i \leftarrow \mathsf{miEnc}(\mathsf{miMSK}, i, [\widetilde{\mathbf{x}}_{i,1}]_1),\ \mathsf{iCT}_i \leftarrow \mathsf{iEnc}(\mathsf{iMSK}_i, [\widetilde{\mathbf{x}}_{i,2}]_2),\ \mathsf{gCT}_i := (\mathsf{miCT}_i, \mathsf{iCT}_i)$

$\underline{\mathsf{gKeyGen}(\mathsf{MSK}, \{[\mathbf{y}_{i,1}]_2, [\mathbf{y}_{i,2}]_1\}_{i\in[n]})} \to \mathsf{gSK}$
$\mathbf{a} \leftarrow \mathbb{Z}_p^k,\ \widetilde{\mathbf{y}}_{i,1} := (\mathbf{y}_{i,1}, 0^{m_2}, \mathbf{a}, 0) \in \mathbb{Z}_p^{m_1+m_2+k+1},\ \widetilde{\mathbf{y}}_{i,2} := (\mathbf{y}_{i,2}, \mathbf{a}, 0) \in \mathbb{Z}_p^{m_2+k+1},\ \widetilde{\mathbf{y}} := (\widetilde{\mathbf{y}}_{1,1}, \ldots, \widetilde{\mathbf{y}}_{n,1})$
$\mathsf{miSK} \leftarrow \mathsf{miKeyGen}(\mathsf{miMSK}, [\widetilde{\mathbf{y}}]_2),\ \mathsf{iSK}_i \leftarrow \mathsf{iKeyGen}(\mathsf{iMSK}_i, [\widetilde{\mathbf{y}}_{i,2}]_1),\ \mathsf{gSK} := (\mathsf{miSK}, \{\mathsf{iSK}_i\}_{i\in[n]})$

$\underline{\mathsf{gDec}(\mathsf{gCT}_1, \ldots, \mathsf{gCT}_n, \mathsf{gSK})} \to z$
Outputs $\mathsf{miDec}(\mathsf{miCT}_1, \ldots, \mathsf{miCT}_n, \mathsf{miSK}) \prod_{i\in[n]} \mathsf{iDec}(\mathsf{iCT}_i, \mathsf{iSK}_i)$

Fig 2: Our mixed-group IP-MIFE scheme.

4.1 Construction

Let $\mathcal{F}^{\mathsf{IP}'}_{m,\mathbb{G}}$ be a function class defined the same as $\mathcal{F}^{\mathsf{IP}}_{m,\mathbb{G}}$ in Def. 3.1 except that G_1 and G_2 are switched, that is, each $f : G_2^m \to G_T$ is specified by $[\mathbf{y}]_1$. We construct a function-hiding MIFE scheme for $\mathcal{F}^{\mathsf{MGIP}}_{m_1,m_2,n,\mathbb{G}}$ from a function-hiding MIFE scheme for $\mathcal{F}^{\mathsf{MIP}}_{m_1+m_2+k+1,n,\mathbb{G}}$ and function-hiding FE scheme for $\mathcal{F}^{\mathsf{IP}'}_{m_2+k+1,\mathbb{G}}$ in a generic way. Note that k is a parameter for the MDDH assumption. A function-hiding MIFE scheme for $\mathcal{F}^{\mathsf{MIP}}_{m,n,\mathbb{G}}$ based on MDDH is easily obtained from a function-hiding multi-input IPFE schemes in [4,17,30]. This is since these schemes in the literatures work even if input vectors for Enc and KeyGen consist of group elements, and Dec first obtains decryption values on the exponent of a target-group generator and then computes its discrete log.

Let $\mathsf{miFE} = (\mathsf{miSetup}, \mathsf{miEnc}, \mathsf{miKeyGen}, \mathsf{miDec})$ be a function-hiding MIFE scheme for $\mathcal{F}^{\mathsf{MIP}}_{m_1+m_2+k+1,n,\mathbb{G}}$ and $\mathsf{iFE} = (\mathsf{iSetup}, \mathsf{iEnc}, \mathsf{iKeyGen}, \mathsf{iDec})$ be a function-hiding FE scheme for $\mathcal{F}^{\mathsf{IP}'}_{m_2+k+1,\mathbb{G}}$. Then, our function-hiding MIFE scheme $\mathsf{gFE} = (\mathsf{gSetup}, \mathsf{gEnc}, \mathsf{gKeyGen}, \mathsf{gDec})$ for $\mathcal{F}^{\mathsf{MGIP}}_{m_1,m_2,n,\mathbb{G}}$ is constructed as shown in Fig 2.

Correctness. Due to the correctness of miFE and iFE, gDec outputs

$$\left[\sum_{i\in[n]} (\langle \widetilde{\mathbf{x}}_{i,1}, \widetilde{\mathbf{y}}_{i,1}\rangle + \langle \widetilde{\mathbf{x}}_{i,2}, \widetilde{\mathbf{y}}_{i,2}\rangle)\right]_T = \left[\sum_{i\in[n]} (\langle \mathbf{x}_{i,1}, \mathbf{y}_{i,1}\rangle + \langle \mathbf{x}_{i,2}, \mathbf{y}_{i,2}\rangle)\right]_T .$$

For security, we have the following theorem.

Theorem 4.1. *If* miFE *and* iFE *are function-hiding, and the bilateral MDDH assumption holds in* $\mathbb{G}$*, then* gFE *is function-hiding. More precisely, for all PPT adversaries* $\mathcal{A}$*, there exist PPT adversaries* $\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3$ *such that*

$$\mathsf{Adv}^{\mathsf{gFE}}_{\mathcal{A},\mathsf{fh}}(\lambda) \le (4q_{\mathsf{CT}}+1)\mathsf{Adv}^{\mathsf{miFE}}_{\mathcal{B}_1,\mathsf{fh}}(\lambda) + n(4q_{\mathsf{CT}}+1)\mathsf{Adv}^{\mathsf{iFE}}_{\mathcal{B}_2,\mathsf{fh}}(\lambda) + 4nq_{\mathsf{CT}}\mathsf{Adv}^{\mathsf{bi}\text{-}\mathcal{D}_k\text{-}\mathsf{MDDH}}_{\mathcal{B}_3}(\lambda).$$

Due to space constraints, the proof is provided in the full version.

$\underline{\mathsf{qSetup}(1^\lambda) \to \mathsf{qPP}, \mathsf{qMSK}}$
$\mathbb{G} \leftarrow \mathcal{G}_{\mathsf{BG}}(1^\lambda),\ w_{1,1}, w_{1,2}, w_{2,1}, w_{2,2}, u_1, u_2, v_1, v_2 \leftarrow \mathbb{Z}_p$
$\mathsf{pPP}, \mathsf{pMSK} \leftarrow \mathsf{pSetup}(1^\lambda),\ \mathsf{iPP}, \mathsf{iMSK} \leftarrow \mathsf{iSetup}(1^\lambda),\ \mathsf{gPP}, \mathsf{gMSK} \leftarrow \mathsf{gSetup}(1^\lambda)$
$\mathsf{qPP} := (\mathbb{G}, \mathsf{pPP}, \mathsf{iPP}, \mathsf{gPP}),\ \mathsf{qMSK} := (\{w_{i,j}\}_{i,j\in[2]}, \{u_i, v_i\}_{i\in[2]}, \mathsf{pMSK}, \mathsf{iMSK}, \mathsf{gMSK})$

$\underline{\mathsf{qEnc}(\mathsf{qMSK}, i, x_i) \to \mathsf{qCT}_i}$
$s, \widetilde{s}, r, t, L \leftarrow \mathbb{Z}_p,\ \mathbf{l} := \mathbf{e}_{i/2} \otimes (1, L) \in \mathbb{Z}_p^4,\ \widetilde{\mathbf{l}} := \mathbf{e}_{i/2} \otimes (L, -1) \in \mathbb{Z}_p^4$
$\mathbf{b} := (x_i, 0, sw_{1,i}, sw_{2,i}, u_i, t, 0, 0) \in \mathbb{Z}_p^8,\ \widetilde{\mathbf{b}} := (x_i, 0, \widetilde{s}\mathbf{e}_{i/2}, r, v_i, 0, 0) \in \mathbb{Z}_p^8$
$\mathbf{d} := (s, 0) \in \mathbb{Z}_p^2,\ \widetilde{\mathbf{d}} := (\widetilde{s}, 0) \in \mathbb{Z}_p^2,\ \mathbf{f} := (r, t, 0, 0) \in \mathbb{Z}_p^4,\ h := 0$
$\mathsf{pCT}_i \leftarrow \mathsf{pEnc}(\mathsf{pMSK}, (\mathbf{l}, [\mathbf{b}]_1)),\ \mathsf{pSK}_i \leftarrow \mathsf{pKeyGen}(\mathsf{pMSK}, (\widetilde{\mathbf{l}}, [\widetilde{\mathbf{b}}]_2))$
$\mathsf{iCT}_i \leftarrow \mathsf{iEnc}(\mathsf{iMSK}, [\mathbf{d}]_1),\ \mathsf{iSK}_i \leftarrow \mathsf{iKeyGen}(\mathsf{iMSK}, [\widetilde{\mathbf{d}}]_2)$
$\mathsf{gCT}_i \leftarrow \mathsf{gEnc}(\mathsf{gMSK}, i, ([\mathbf{f}]_1, [h]_2)),\ \mathsf{qCT}_i := (\mathsf{pCT}_i, \mathsf{pSK}_i, \mathsf{iCT}_i, \mathsf{iSK}_i, \mathsf{gCT}_i)$

$\underline{\mathsf{qKeyGen}(\mathsf{qMSK}, \mathbf{c} = \{c_{\mu,\nu}\}_{\mu,\nu\in[2]}) \to \mathsf{qSK}}$
$\widetilde{\mathbf{f}}_i := \left(\sum_{\mu\in[2]} c_{i,\mu}u_\mu, \sum_{\mu\in[2]} c_{\mu,i}v_\mu, 0, 0\right) \in \mathbb{Z}_p^4,\ \widetilde{h}_i := 0,\ \sigma_{i,\theta} := c_{i,\theta}w_{i,\theta}$
$\mathsf{gSK} \leftarrow \mathsf{gKeyGen}(\mathsf{gMSK}, \{[\widetilde{\mathbf{f}}_i]_2, [\widetilde{h}_i]_1\}_{i\in[2]}),\ \mathsf{qSK} := (\mathbf{c}, \mathsf{gSK}, \{\sigma_{i,\theta}\}_{i,\theta\in[2]})$

$\underline{\mathsf{qDec}(\mathsf{qCT}_1, \mathsf{qCT}_2, \mathsf{qSK}) \to z}$
$[z_1]_T := \prod_{\mu,\nu\in[2]} \mathsf{pDec}(\mathsf{pCT}_\nu, \mathsf{pSK}_\mu)^{c_{\mu,\nu}},\ [z_2]_T := \prod_{i,\theta\in[2]} \mathsf{iDec}(\mathsf{iCT}_\theta, \mathsf{iSK}_i)^{\sigma_{i,\theta}}$
$[z_3]_T := \mathsf{gDec}(\mathsf{gCT}_1, \mathsf{gCT}_2, \mathsf{gSK}),\ [z]_T := [z_1 - z_2 - z_3]_T$
Searches for z within the range of $z \le |4CX^2|$

Fig 3: Our two-input quadratic MIFE scheme.

5 Warm-up: Two Input Quadratic MIFE

Since our general quadratic MIFE scheme (Sec. 6) is quite complex, we first present a simpler scheme as a warm-up. This scheme is a MIFE scheme for $\mathcal{F}^{\mathsf{MQF}}_{1,2,X,C}$ from the SXDH assumption, that is $m = 1, n = 2$. For ease of exposition, we also restrict the number of ciphertext queries to 2 per slot. The SXDH assumption is captured as the $\mathcal{D}_k$ assumption where $\mathcal{D}_k$ consists of all matrices with the form of $(a, 1)^\top \in \mathbb{Z}_p^2$.

Let $\mathsf{pFE} = (\mathsf{pSetup}, \mathsf{pEnc}, \mathsf{pKeyGen}, \mathsf{pDec})$ be an FE scheme for $\mathcal{F}^{\mathsf{PIP}}_{4,8,\mathbb{G}}$ (Def.3.2), $\mathsf{iFE} = (\mathsf{iSetup}, \mathsf{iEnc}, \mathsf{iKeyGen}, \mathsf{iDec})$ be an FE scheme for $\mathcal{F}^{\mathsf{IP}}_{2,\mathbb{G}}$ (Def. 3.1), and $\mathsf{gFE} = (\mathsf{gSetup}, \mathsf{gEnc}, \mathsf{gKeyGen}, \mathsf{gDec})$ be an FE scheme for $\mathcal{F}^{\mathsf{MGIP}}_{4,1,2,\mathbb{G}}$ (Def. 4.2). The warm-up scheme $\mathsf{qFE} = (\mathsf{qSetup}, \mathsf{qEnc}, \mathsf{qKeyGen}, \mathsf{qDec})$ is constructed from pFE, iFE, and gFE as shown in Fig 3. Since gFE cannot be instantiated from SXDH, the warm-up scheme needs an additional assumption such as XDLIN (bilateral 2-Lin).

Correctness. Let $s_i, \widetilde{s}_i, r_i, t_i, \mathbf{l}_i, \widetilde{\mathbf{l}}_i, \mathbf{b}_i, \widetilde{\mathbf{b}}_i$ for $i \in [2]$ be random elements used to generate qCT_i. Observe that $\langle \mathbf{l}_i, \widetilde{\mathbf{l}}_I \rangle = 0$ for all $i, I \in [2]$, and thus $\mathsf{pDec}(\mathsf{pCT}_i, \mathsf{pSK}_I) = \langle \mathbf{b}_i, \widetilde{\mathbf{b}}_I \rangle$. Due to the correctness of $\mathsf{pFE}, \mathsf{iFE}, \mathsf{gEF}$, we have

$$z_1 = \sum_{\mu,\nu\in[2]} c_{\mu,\nu}(x_\mu x_\nu + s_\nu \widetilde{s}_\mu w_{\mu,\nu} + r_\mu u_\nu + t_\nu v_\mu)$$

$$z_2 = \sum_{\mu,\nu\in[2]} c_{\mu,\nu} s_\nu \widetilde{s}_\mu w_{\mu,\nu},\ z_3 = \sum_{\mu,\nu\in[2]} c_{\mu,\nu}(r_\mu u_\nu + t_\nu v_\mu).$$

Hence, we have $z = \sum_{\mu,\nu\in[2]} c_{\mu,\nu} x_\mu x_\nu$.

5.1 Multi-input IPFE Scheme for Security Analysis

Before going to the security analysis of our quadratic MIFE scheme, we introduce a message-hiding IP-MIFE scheme, i.e. an MIFE scheme for $\mathcal{F}^{\mathsf{MIP}}_{m,n,\mathbb{G}}$, denoted by $\mathsf{miFE} = (\mathsf{miSetup}, \mathsf{miEnc}, \mathsf{miKeyGen}, \mathsf{miDec})$ that we use for the security proof. The scheme is obtained by applying the conversion of single to multi-input IPFE by Abdalla *et al.* [4, Sec. 4.1], to the single-input IPFE scheme by Abdalla *et al.* [3, Sec. 5]. The resulting scheme satisfies the message-hiding security under the DDH assumption. Note that although Abdalla *et al.* considered the conversion in the adaptive setting, it is not hard to see that the conversion works in the selective setting. The original scheme in [3] uses a pairing-free group for the construction, but it is easy to see that their scheme can be similarly built on pairing groups where the SXDH assumption holds. The scheme is described in Fig 4.

$\underline{\mathsf{miSetup}(1^\lambda) \to \mathsf{miPP}, \mathsf{miMSK}}$
$\mathbb{G} \leftarrow \mathcal{G}_{\mathsf{BG}}(1^\lambda),\ \mathbf{w}_1, \dots, \mathbf{w}_n \leftarrow \mathbb{Z}_p^m,\ \mathbf{u}_1, \dots, \mathbf{u}_n \leftarrow \mathbb{Z}_p^m$
$\mathsf{miPP} := (\mathbb{G}, [\mathbf{w}_1]_1, \dots, [\mathbf{w}_n]_1),$
$\mathsf{miMSK} := (\mathbf{w}_1, \dots, \mathbf{w}_n, \mathbf{u}_1, \dots, \mathbf{u}_n)$

$\underline{\mathsf{miEnc}(\mathsf{miMSK}, i, \mathbf{x}_i) \to \mathsf{miCT}_i}$
$s \leftarrow \mathbb{Z}_p,\ \mathsf{miCT}_i := [\mathbf{c}_i]_1 = ([s]_1, [s\mathbf{w}_i + \mathbf{u}_i + \mathbf{x}_i]_1)$

$\underline{\mathsf{miKeyGen}(\mathsf{miMSK}, \mathbf{y}_1, \dots, \mathbf{y}_n) \to \mathsf{miSK}}$
$\mathsf{miSK}_0 := -\sum_{i\in[n]} \langle \mathbf{y}_i, \mathbf{u}_i \rangle,\ \mathsf{miSK}_i := (-\mathbf{y}_i^\top \mathbf{w}_i, \mathbf{y}_i),$
$\mathsf{miSK} := (\mathsf{miSK}_0, \{\mathsf{miSK}_i\}_{i\in[n]})$

$\underline{\mathsf{miDec}(\mathsf{miCT}_1, \dots, \mathsf{miCT}_n, \mathsf{miSK}) \to z}$
$[z]_1 = [\sum_{i\in[n]} \langle \mathbf{c}_i, \mathsf{miSK}_i \rangle + \mathsf{miSK}_0]_1$

Fig 4: IP-MIFE scheme by Abdalla *et al.*

$\underline{\mathsf{G}_\beta}$
$\{i, \mathbf{x}_i^{j,0}, \mathbf{x}_i^{j,1}\}_{i\in[2],j\in[2]} \leftarrow \mathcal{A}(1^\lambda)$
$\mathsf{qPP}, \mathsf{qMSK} \leftarrow \mathsf{qSetup}(1^\lambda)$
$\mathsf{qCT}_i^j \leftarrow \mathsf{qEnc}(\mathsf{qMSK}, i, \mathbf{x}_i^{j,\beta})$
$\mathbf{c} \leftarrow \mathcal{A}(\mathsf{qPP}, \{\mathsf{qCT}_i^j\}_{i\in[2],j\in[2]})$
$\mathsf{qSK} \leftarrow \mathsf{qKeyGen}(\mathsf{qMSK}, \mathbf{c})$
$\beta' \leftarrow \mathcal{A}(\mathsf{qSK})$

Fig 5: qFE warmup security game.

5.2 Proof of Security

Theorem 5.1. *If* pFE *is partially function-hiding,* iFE *and* gFE *are function-hiding, and* $\mathcal{G}_{\mathsf{BG}}$ *outputs bilinear groups where the SXDH assumption holds, then* qFE *is message-hiding as long as* $q_{\mathsf{CT}} = 2$ *and* $q_{\mathsf{SK}} = 1$.

Proof. For ease of exposition, we prove security in the restricted game where an adversary makes two ciphertext queries per slot and one secret key query.

$\underline{\mathsf{qCT}_1^1}$ $\mathbf{b}_1^1 := (x_1^{1,\beta},\ 0,\ s_1^1 w_{1,1},\ s_1^1 w_{2,1},\ u_1,\ t_1^1,\ 0,\ 0)$ $\widetilde{\mathbf{b}}_1^1 := (x_1^{1,\beta},\ 0,\ \widetilde{s}_1^1,\ 0,\ r_1^1,\ v_1,\ 0,\ 0)$ $\mathbf{d}_1^1 := (s_1^1, 0),\ \widetilde{\mathbf{d}}_1^1 := (\widetilde{s}_1^1, 0)$ $\mathbf{f}_1^1 := (r_1^1, t_1^1, 0, 0),\ h_1^1 := 0$	$\underline{\mathsf{qCT}_2^1}$ $\mathbf{b}_2^1 := (x_2^{1,\beta},\ 0,\ s_2^1 w_{1,2},\ s_2^1 w_{2,2},\ u_2,\ t_2^1,\ 0,\ 0)$ $\widetilde{\mathbf{b}}_2^1 := (x_2^{1,\beta},\ 0,\ 0,\ \widetilde{s}_2^1,\ r_2^1,\ v_2,\ 0,\ 0)$ $\mathbf{d}_2^1 := (s_2^1, 0),\ \widetilde{\mathbf{d}}_2^1 := (\widetilde{s}_2^1, 0)$ $\mathbf{f}_2^1 := (r_2^1, t_2^1, 0, 0),\ h_2^1 := 0$
$\underline{\mathsf{qCT}_1^2}$ $\mathbf{b}_1^2 := (x_1^{2,\beta},\ 0,\ s_1^2 w_{1,1},\ s_1^2 w_{2,1},\ u_1,\ t_1^2,\ 0,\ 0)$ $\widetilde{\mathbf{b}}_1^2 := (x_1^{2,\beta},\ 0,\ \widetilde{s}_1^2,\ 0,\ r_1^2,\ v_1,\ 0,\ 0)$ $\mathbf{d}_1^2 := (s_1^2, 0),\ \widetilde{\mathbf{d}}_1^2 := (\widetilde{s}_1^2, 0)$ $\mathbf{f}_1^2 := (r_1^2, t_1^2, 0, 0),\ h_1^2 := 0$	$\underline{\mathsf{qCT}_2^2}$ $\mathbf{b}_2^2 := (x_2^{2,\beta},\ 0,\ s_2^2 w_{1,2},\ s_2^2 w_{2,2},\ u_2,\ t_2^2,\ 0,\ 0)$ $\widetilde{\mathbf{b}}_2^2 := (x_2^{2,\beta},\ 0,\ 0,\ \widetilde{s}_2^2,\ r_2^2,\ v_2,\ 0,\ 0)$ $\mathbf{d}_2^2 := (s_2^2, 0),\ \widetilde{\mathbf{d}}_2^2 := (\widetilde{s}_2^2, 0)$ $\mathbf{f}_2^2 := (r_2^2, t_2^2, 0, 0),\ h_2^2 := 0$
$\underline{\mathsf{qSK}}$ $\widetilde{\mathbf{f}}_1 := (\sum_{\mu\in[2]} c_{1,\mu} u_\mu, \sum_{\mu\in[2]} c_{\mu,1} v_\mu, 0, 0)$ $\widetilde{h}_1 := 0$	$\widetilde{\mathbf{f}}_2 := (\sum_{\mu\in[2]} c_{2,\mu} u_\mu, \sum_{\mu\in[2]} c_{\mu,2} v_\mu, 0, 0)$ $\widetilde{h}_2 := 0$

Fig 6: Vectors in G_β.

$\underline{\mathsf{qCT}_1^1}$ $\mathbf{b} := (\ x_1^{1,0},\ \boxed{x_1^{1,1}},\ s_1^1 w_{1,1},\ s_1^1 w_{2,1},\ u_1,\ t_1^1,\ 0,\ \boxed{t_1^1 v_1 + x_1^{1,0} x_1^{1,0}}\ )$ $\widetilde{\mathbf{b}} := (\ \boxed{0},\ 0,\ \widetilde{s}_1^1,\ 0,\ r_1^1,\ \boxed{0},\ 0,\ \boxed{1}\ )$ $\mathbf{d} := (s_1^1, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^1, 0)$ $\mathbf{f} := (r_1^1, t_1^1, \boxed{t_1^1 v_1}, 0),\ h := 0$	$\underline{\mathsf{qCT}_2^1}$ $\mathbf{b} := (\ x_2^{1,0},\ \boxed{x_2^{1,1}},\ s_2^1 w_{1,2},\ s_2^1 w_{2,2},\ u_2,\ t_2^1,\ \boxed{t_2^1 v_1},\ \boxed{t_2^1 v_1 + x_1^{1,0} x_2^{1,0}}\ )$ $\widetilde{\mathbf{b}} := (\ x_2^{1,0},\ 0,\ 0,\ \widetilde{s}_2^1,\ r_2^1,\ v_2,\ 0,\ 0\ )$ $\mathbf{d} := (s_2^1, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^1, 0)$ $\mathbf{f} := (r_2^1, t_2^1, \boxed{t_2^1 v_1}, 0),\ h := 0$
$\underline{\mathsf{qCT}_1^2}$ $\mathbf{b} := (\ x_1^{2,0},\ \boxed{x_1^{2,1}},\ s_1^2 w_{1,1},\ s_1^2 w_{2,1},\ u_1,\ t_1^2,\ \boxed{t_1^2 v_1},\ 0\ )$ $\widetilde{\mathbf{b}} := (\ x_1^{2,0},\ 0,\ \widetilde{s}_1^2,\ 0,\ r_1^2,\ \boxed{0},\ \boxed{1},\ 0\ )$ $\mathbf{d} := (s_1^2, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^2, 0)$ $\mathbf{f} := (r_1^2, t_1^2, \boxed{t_1^2 v_1}, 0),\ h := 0$	$\underline{\mathsf{qCT}_2^2}$ $\mathbf{b} := (\ x_2^{2,0},\ \boxed{x_2^{2,1}},\ s_2^2 w_{1,2},\ s_2^2 w_{2,2},\ u_2,\ t_2^2,\ \boxed{t_2^2 v_1},\ \boxed{t_2^2 v_1 + x_1^{1,0} x_2^{2,0}}\ )$ $\widetilde{\mathbf{b}} := (\ x_2^{2,0},\ 0,\ 0,\ \widetilde{s}_2^2,\ r_2^2,\ v_2,\ 0,\ 0\ )$ $\mathbf{d} := (s_2^2, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^2, 0)$ $\mathbf{f} := (r_2^2, t_2^2, \boxed{t_2^2 v_1}, 0),\ h := 0$
$\underline{\mathsf{qSK}}$ $\widetilde{\mathbf{f}}_1 := (\sum_{\mu\in[2]} c_{1,\mu} u_\mu, \boxed{c_{2,1} v_2}, \boxed{c_{1,1}}, \boxed{c_{2,1}})$ $\widetilde{h}_1 := 0$	$\widetilde{\mathbf{f}}_2 := (\sum_{\mu\in[2]} c_{2,\mu} u_\mu, \boxed{c_{2,2} v_2}, \boxed{c_{1,2}}, \boxed{c_{2,2}})$ $\widetilde{h}_2 := 0$

Fig 7: Vectors in H_1.

This simplification showcases the basic strategy of the general proof, which is provided in Sec. 6. At a high-level view, our security proof is inspired by that of the IP-MIFE schemes by Abdalla *et al.* [4] in which the first queried ciphertexts of each slot are changed from bit 0 to bit 1 by the information-theoretic property of the one-time pad and the rest of ciphertexts are changed by the security of an IPFE scheme. In our case, the IPFE scheme will instead correspond to the IP-MIFE scheme in Sec. 5.1.

Intuitively, we want to prove $\mathsf{G}_0 \approx_c \mathsf{G}_1$ where G_β is the message-hiding security game (described in Fig 5). In G_β, the vectors in the ciphertexts and the secret key that the adversary obtains are defined as Fig 6. We introduce a series of hybrid games, $\mathsf{H}_1, \ldots, \mathsf{H}_{15}$, and prove $\mathsf{G}_0 \approx_c \mathsf{H}_1 \approx_c \cdots \approx_c \mathsf{H}_{15} \approx_c \mathsf{G}_1$. In each hybrid game, the vectors for generating the ciphertexts and the secret keys are changed from G_0, which is shown in Fig 7 to 21. We frame the parts that are changed from the previous game by a box and sometimes denote the parts that are not changed by —.

$\underline{\mathsf{G}_0 \approx_c \mathsf{H}_1.}$ We can justify this indistinguishability by the (partially) function-hiding property of pFE and gFE. For all $i, j, I, J \in [2]$, we can see that $\langle \mathbf{b}_i^j, \widetilde{\mathbf{b}}_I^J \rangle$

qCT_1^1	qCT_2^1
$\boxed{\ddot{v}_1^1 \leftarrow \mathbb{Z}_p}$ $\mathbf{b} := (\ x_1^{1,0},\ x_1^{1,1},\ s_1^1 w_{1,1},\ s_1^1 w_{2,1},\ u_1,\ t_1^1,\ 0,\ \boxed{\ddot{v}_1^1} + x_1^{1,0} x_1^{1,0}\ )$ $\widetilde{\mathbf{b}} := (\ 0,\ 0,\ \widetilde{s}_1^1,\ 0,\ r_1^1,\ 0,\ 0,\ 1\ )$ $\mathbf{d} := (s_1^1, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^1, 0)$ $\mathbf{f} := (r_1^1, t_1^1, \boxed{\ddot{v}_1^1}, 0),\ h := 0$	$\boxed{\ddot{v}_2^1 \leftarrow \mathbb{Z}_p}$ $\mathbf{b} := (\ x_2^{1,0},\ x_2^{1,1},\ s_2^1 w_{1,2},\ s_2^1 w_{2,2},\ u_2,\ t_2^1,\ \boxed{\ddot{v}_2^1},\ \boxed{\ddot{v}_2^1} + x_1^{1,0} x_2^{1,0}\ )$ $\widetilde{\mathbf{b}} := (\ x_2^{1,0},\ 0,\ 0,\ \widetilde{s}_2^1,\ r_2^1,\ v_2,\ 0,\ 0\ )$ $\mathbf{d} := (s_2^1, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^1, 0)$ $\mathbf{f} := (r_2^1, t_2^1, \boxed{\ddot{v}_2^1}, 0),\ h := 0$
qCT_1^2 $\boxed{\ddot{v}_1^2 \leftarrow \mathbb{Z}_p}$ $\mathbf{b} := (\ x_1^{2,0},\ x_1^{2,1},\ s_1^2 w_{1,1},\ s_1^2 w_{2,1},\ u_1,\ t_1^2,\ \boxed{\ddot{v}_1^2},\ 0\ )$ $\widetilde{\mathbf{b}} := (\ x_1^{2,0},\ 0,\ \widetilde{s}_1^2,\ 0,\ r_1^2,\ 0,\ 1,\ 0\ )$ $\mathbf{d} := (s_1^2, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^2, 0)$ $\mathbf{f} := (r_1^2, t_1^2, \boxed{\ddot{v}_1^2}, 0),\ h := 0$	qCT_2^2 $\boxed{\ddot{v}_2^2 \leftarrow \mathbb{Z}_p}$ $\mathbf{b} := (\ x_2^{2,0},\ x_2^{2,1},\ s_2^2 w_{1,2},\ s_2^2 w_{2,2},\ u_2,\ t_2^2,\ \boxed{\ddot{v}_2^2},\ \boxed{\ddot{v}_2^2} + x_1^{1,0} x_2^{2,0}\ )$ $\widetilde{\mathbf{b}} := (\ x_2^{2,0},\ 0,\ 0,\ \widetilde{s}_2^2,\ r_2^2,\ v_2,\ 0,\ 0\ )$ $\mathbf{d} := (s_2^2, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^2, 0)$ $\mathbf{f} := (r_2^2, t_2^2, \boxed{\ddot{v}_2^2}, 0),\ h := 0$
qSK $\widetilde{\mathbf{f}}_1 := (\sum_{\mu\in[2]} c_{1,\mu} u_\mu, c_{2,1} v_2, c_{1,1}, c_{2,1})$ $\widetilde{h}_1 := 0$	$\widetilde{\mathbf{f}}_2 := (\sum_{\mu\in[2]} c_{2,\mu} u_\mu, c_{2,2} v_2, c_{1,2}, c_{2,2})$ $\widetilde{h}_2 := 0$

Fig 8: Vectors in H_2.

qCT_1^1	qCT_2^1
$\ddot{v}_1^1 \leftarrow \mathbb{Z}_p$ $\mathbf{b} := (\ —, 0,\ \ddot{v}_1^1 + \boxed{x_1^{1,1} x_1^{1,1}}\ )$ $\widetilde{\mathbf{b}} := (\ —, 0,\ 1\ )$ $\mathbf{d} := (s_1^1, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^1, 0)$ $\mathbf{f} := (r_1^1, t_1^1, \ddot{v}_1^1 + \boxed{x_1^{1,1} x_1^{1,1} - x_1^{1,0} x_1^{1,0}}, 0),\ h := 0$	$\ddot{v}_2^1 \leftarrow \mathbb{Z}_p$ $\mathbf{b} := (\ —, \ddot{v}_2^1 + \boxed{x_1^{1,1} x_2^{1,1} - x_1^{1,0} x_2^{1,0}},\ \ddot{v}_2^1 + \boxed{x_1^{1,1} x_2^{1,1}}\ )$ $\widetilde{\mathbf{b}} := (\ —,\ 0,\ 0\ )$ $\mathbf{d} := (s_2^1, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^1, 0)$ $\mathbf{f} := (r_2^1, t_2^1, \ddot{v}_2^1 + \boxed{x_1^{1,1} x_2^{1,1} - x_1^{1,0} x_2^{1,0}}, 0),\ h := 0$
qCT_1^2 $\ddot{v}_1^2 \leftarrow \mathbb{Z}_p$ $\mathbf{b} := (\ —, \ddot{v}_1^2 + \boxed{x_1^{1,1} x_1^{1,1} - x_1^{1,0} x_1^{1,0}},\ 0\ )$ $\widetilde{\mathbf{b}} := (\ —,\ 1,\ 0\ )$ $\mathbf{d} := (s_1^2, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^2, 0)$ $\mathbf{f} := (r_1^2, t_1^2, \ddot{v}_1^2 + \boxed{x_1^{1,1} x_1^{1,1} - x_1^{1,0} x_1^{1,0}}, 0),\ h := 0$	qCT_2^2 $\ddot{v}_2^2 \leftarrow \mathbb{Z}_p$ $\mathbf{b} := (\ —, \ddot{v}_2^2 + \boxed{x_1^{1,1} x_2^{2,1} - x_1^{1,0} x_2^{2,0}},\ \ddot{v}_2^2 + \boxed{x_1^{1,1} x_2^{2,1}}\ )$ $\widetilde{\mathbf{b}} := (\ —,\ 0,\ 0\ )$ $\mathbf{d} := (s_2^2, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^2, 0)$ $\mathbf{f} := (r_2^2, t_2^2, \ddot{v}_2^2 + \boxed{x_1^{1,1} x_2^{2,1} - x_1^{1,0} x_2^{2,0}}, 0),\ h := 0$
qSK $\widetilde{\mathbf{f}}_1 := (\sum_{\mu\in[2]} c_{1,\mu} u_\mu, c_{2,1} v_2, c_{1,1}, c_{2,1})$ $\widetilde{h}_1 := 0$	$\widetilde{\mathbf{f}}_2 := (\sum_{\mu\in[2]} c_{2,\mu} u_\mu, c_{2,2} v_2, c_{1,2}, c_{2,2})$ $\widetilde{h}_2 := 0$

Fig 9: Vectors in H_3.

in G_0 and that in H_1 are equal unless $i = I$ and $j \neq J$. Recall that $\langle \mathbf{l}_i^j, \widetilde{\mathbf{l}}_I^J \rangle \neq 0$ with overwhelming probability if $i = I$ and $j \neq J$, since L is chosen from the exponentially large space, $\mathbb{Z}_p$. Hence, the indistinguishability of $\{\mathbf{b}, \widetilde{\mathbf{b}}\}$ between G_0 and H_1 is implied by the partially function-hiding property of pFE.

Similarly, for all $i, j \in [2]$, $\langle \mathbf{f}_i^j, \widetilde{\mathbf{f}}_i \rangle$ in G_0 and that in H_1 are equal, which implies, for all $j_1, j_2 \in [2]$, $\sum_{i\in[2]} (\langle \mathbf{f}_i^{j_i}, \widetilde{\mathbf{f}}_i \rangle + h_i^{j_i} \widetilde{h}_i)$ in G_0 and that in H_1 are equal. Thus, the indistinguishability of $\{\mathbf{f}, \widetilde{\mathbf{f}}\}$ between G_0 and H_1 is implied by the function-hiding property of gFE.

$\underline{\mathsf{H}_1 \approx_c \mathsf{H}_2.}$ We can justify this indistinguishability by the SXDH assumption, which implies $(\mathbb{G}, [\mathbf{t}]_1, [v_1 \mathbf{t}]_1) \approx_c (\mathbb{G}, [\mathbf{t}]_1, [\ddot{\mathbf{v}}]_1)$ where $\mathbb{G} \leftarrow \mathcal{G}_{\mathsf{BG}}(1^\lambda), \mathbf{t} = \{t_i^j\}_{i,j\in[2]}$, $\ddot{\mathbf{v}} = \{\ddot{v}_i^j\}_{i,j\in[2]} \leftarrow \mathbb{Z}_p^4, v_1 \leftarrow \mathbb{Z}_p$.

$\underline{\mathsf{qCT}_1^1}$	$\underline{\mathsf{qCT}_2^1}$
$\mathbf{b} := (\ -, 0, \boxed{t_1^1 v_1} + x_1^{1,1} x_1^{1,1}\ )$ $\widetilde{\mathbf{b}} := (\ -, 0, \quad 1 \quad\ )$ $\mathbf{d} := (s_1^1, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^1, 0)$ $\mathbf{f} := (r_1^1, t_1^1, \boxed{t_1^1 v_1} + x_1^{1,1} x_1^{1,1} - x_1^{1,0} x_1^{1,0}, 0),\ h := 0$	$\mathbf{b} := (\ -, \boxed{t_2^1 v_1} + x_1^{1,1} x_2^{1,1} - x_1^{1,0} x_2^{1,0}, \boxed{t_2^1 v_1} + x_1^{1,1} x_2^{1,1}\ )$ $\widetilde{\mathbf{b}} := (\ -, \quad 0, \quad 0\ )$ $\mathbf{d} := (s_2^1, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^1, 0)$ $\mathbf{f} := (r_2^1, t_2^1, \boxed{t_2^1 v_1} + x_1^{1,1} x_2^{1,1} - x_1^{1,0} x_2^{1,0}, 0),\ h := 0$
$\underline{\mathsf{qCT}_1^2}$	$\underline{\mathsf{qCT}_2^2}$
$\mathbf{b} := (\ -, \boxed{t_1^2 v_1} + x_1^{1,1} x_1^{1,1} - x_1^{1,0} x_1^{1,0}, 0\ )$ $\widetilde{\mathbf{b}} := (\ -, \quad 1, \quad 0\ )$ $\mathbf{d} := (s_1^2, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^2, 0)$ $\mathbf{f} := (r_1^2, t_1^2, \boxed{t_1^2 v_1} + x_1^{1,1} x_1^{1,1} - x_1^{1,0} x_1^{1,0}, 0),\ h := 0$	$\mathbf{b} := (\ -, \boxed{t_2^2 v_1} + x_1^{1,1} x_2^{2,1} - x_1^{1,0} x_2^{2,0}, \boxed{t_2^2 v_1} + x_1^{1,1} x_2^{2,1}\ )$ $\widetilde{\mathbf{b}} := (\ -, \quad 0, \quad 0\ )$ $\mathbf{d} := (s_2^2, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^2, 0)$ $\mathbf{f} := (r_2^2, t_2^2, \boxed{t_2^2 v_1} + x_1^{1,1} x_2^{2,1} - x_1^{1,0} x_2^{2,0}, 0),\ h := 0$
$\underline{\mathsf{qSK}}$	
$\widetilde{\mathbf{f}}_1 := (\sum_{\mu\in[2]} c_{1,\mu} u_\mu, c_{2,1} v_2, c_{1,1}, c_{2,1})$ $\widetilde{h}_1 := 0$	$\widetilde{\mathbf{f}}_2 := (\sum_{\mu\in[2]} c_{2,\mu} u_\mu, c_{2,2} v_2, c_{1,2}, c_{2,2})$ $\widetilde{h}_2 := 0$

Fig 10: Vectors in H_4.

$\underline{\mathsf{qCT}_1^1}$	$\underline{\mathsf{qCT}_2^1}$
$\mathbf{b} := (\ x_1^{1,0},\ x_1^{1,1},\ s_1^1 w_{1,1}, s_1^1 w_{2,1}, u_1,\ t_1^1,\ 0, \boxed{0}\ )$ $\widetilde{\mathbf{b}} := (\ 0,\ \boxed{x_1^{1,1}},\ \widetilde{s}_1^1,\ 0,\ r_1^1, \boxed{v_1}, 0, \boxed{0}\ )$ $\mathbf{d} := (s_1^1, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^1, 0)$ $\mathbf{f} := (r_1^1, t_1^1, \cancel{t_1^1 v_1} + x_1^{1,1} x_1^{1,1} - x_1^{1,0} x_1^{1,0}, 0),\ h := 0$	$\mathbf{b} := (\ -, \cancel{t_2^1 v_1} + x_1^{1,1} x_2^{1,1} - x_1^{1,0} x_2^{1,0}, \boxed{0}\ )$ $\widetilde{\mathbf{b}} := (\ -, \quad 0, \quad 0\ )$ $\mathbf{d} := (s_2^1, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^1, 0)$ $\mathbf{f} := (r_2^1, t_2^1, \cancel{t_2^1 v_1} + x_1^{1,1} x_2^{1,1} - x_1^{1,0} x_2^{1,0}, 0),\ h := 0$
$\underline{\mathsf{qCT}_1^2}$	$\underline{\mathsf{qCT}_2^2}$
$\mathbf{b} := (\ x_1^{2,0},\ x_1^{2,1},\ s_1^2 w_{1,1}, s_1^2 w_{2,1}, u_1,\ t_1^2,\ \cancel{t_1^2 v_1} + x_1^{1,1} x_1^{1,1} - x_1^{1,0} x_1^{1,0}, 0\ )$ $\widetilde{\mathbf{b}} := (\ x_1^{2,0},\ 0,\ \widetilde{s}_1^2,\ 0,\ r_1^2, \boxed{v_1},\ 1,\ 0\ )$ $\mathbf{d} := (s_1^2, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^2, 0)$ $\mathbf{f} := (r_1^2, t_1^2, \cancel{t_1^2 v_1} + x_1^{1,1} x_1^{1,1} - x_1^{1,0} x_1^{1,0}, 0),\ h := 0$	$\mathbf{b} := (\ -, \cancel{t_2^2 v_1} + x_1^{1,1} x_2^{2,1} - x_1^{1,0} x_2^{2,0}, \boxed{0}\ )$ $\widetilde{\mathbf{b}} := (\ -, \quad 0, \quad 0\ )$ $\mathbf{d} := (s_2^2, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^2, 0)$ $\mathbf{f} := (r_2^2, t_2^2, \cancel{t_2^2 v_1} + x_1^{1,1} x_2^{2,1} - x_1^{1,0} x_2^{2,0}, 0),\ h := 0$
$\underline{\mathsf{qSK}}$	
$\widetilde{\mathbf{f}}_1 := (\sum_{\mu\in[2]} c_{1,\mu} u_\mu, \boxed{\sum_{\mu\in[2]} c_{\mu,1} v_\mu}, c_{1,1}, c_{2,1})$ $\widetilde{h}_1 := 0$	$\widetilde{\mathbf{f}}_2 := (\sum_{\mu\in[2]} c_{2,\mu} u_\mu, \boxed{\sum_{\mu\in[2]} c_{\mu,2} v_\mu}, c_{1,2}, c_{2,2})$ $\widetilde{h}_2 := 0$

Fig 11: Vectors in H_5.

$\underline{\mathsf{H}_2 = \mathsf{H}_3.}$ These hybrid games are information-theoretically equivalent. This can be confirmed by setting $\ddot{v}_i^j := \begin{cases} \ddot{v}_i'^j + x_1^{1,1} x_i^{1,1} - x_1^{1,0} x_i^{1,0} & (i=1) \\ \ddot{v}_i'^j + x_1^{1,1} x_i^{j,1} - x_1^{1,0} x_i^{j,0} & (i=2) \end{cases}$ where $\ddot{v}_i'^j \leftarrow \mathbb{Z}_p$.

$\underline{\mathsf{H}_3 \approx_c \mathsf{H}_4.}$ We can justify this indistinguishability by the SXDH assumption, and the indistinguishability can be shown similarly to that between H_1 and H_2.

$\underline{\mathsf{H}_4 \approx_c \mathsf{H}_5.}$ We can justify this indistinguishability by the (partially) function-hiding property of pFE and gFE, similarly to the case of $\mathsf{G}_0 \approx_c \mathsf{H}_1$.

$\underline{\mathsf{H}_5 \approx_c \mathsf{H}_6.}$ We can justify this indistinguishability by the (partially) function-hiding property of pFE, iFE, and gFE, similarly to the case of $\mathsf{G}_0 \approx_c \mathsf{H}_1$. Note that here we also need to consider iFE since $\{\mathbf{d}, \widetilde{\mathbf{d}}\}$ is also changed, but it is easy to see that, for all $i, j, I, J \in [2]$, $\langle \mathbf{d}_i^j, \widetilde{\mathbf{d}}_I^J \rangle$ in H_5 and that in H_6 are equal.

$\underline{\mathsf{H}_6 \approx_c \mathsf{H}_7.}$ We can justify this indistinguishability by the SXDH assumption, which implies $(\mathbb{G}, [\mathbf{s}]_1, [\widetilde{s}_1^2 \mathbf{s}]_1) \approx_c (\mathbb{G}, [\mathbf{s}]_1, [\ddot{\mathbf{s}}]_1)$ and $(\mathbb{G}, [\mathbf{u}]_1, [r_1^2 \mathbf{u}]_1) \approx_c (\mathbb{G}, [\mathbf{u}]_1, [\ddot{\mathbf{u}}]_1)$ where $\mathbb{G} \leftarrow \mathcal{G}_{\mathsf{BG}}(1^\lambda), \mathbf{s} = \{s_i^j\}_{i,j\in[2]}, \ddot{\mathbf{s}} = \{\ddot{s}_i^j\}_{i,j\in[2]} \leftarrow \mathbb{Z}_p^4, \widetilde{s}_1^2 \leftarrow \mathbb{Z}_p, \mathbf{u} = \{u_i\}_{i\in[2]}$, $\ddot{\mathbf{u}} = \{\ddot{u}_i\}_{i\in[2]} \leftarrow \mathbb{Z}_p^2, r_1^2 \leftarrow \mathbb{Z}_p$.

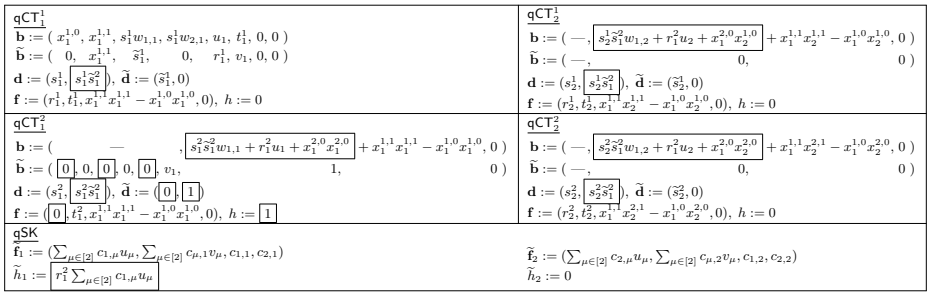

Fig 12: Vectors in H_6.

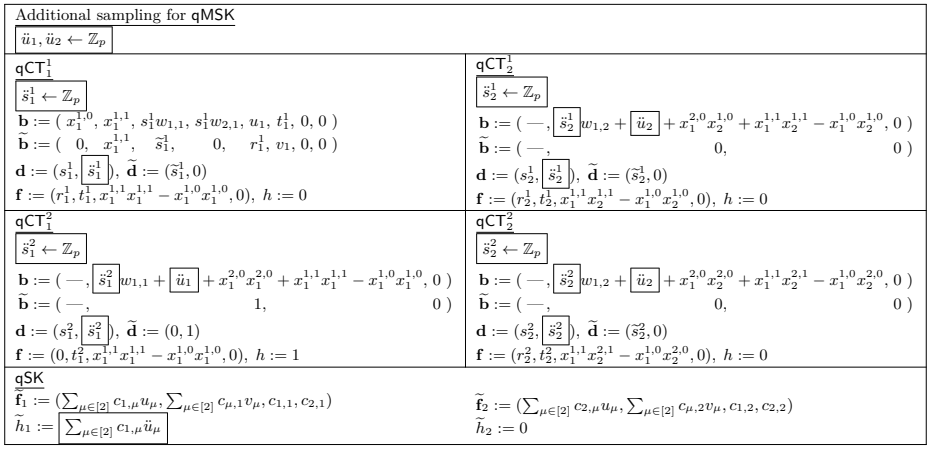

Fig 13: Vectors in H_7.

$\underline{\mathsf{H}_7 \approx_c \mathsf{H}_8}$. We can justify this indistinguishability by the message-hiding property of miFE. First, we prove that, for all $j \in [2]$, we have

$$\begin{aligned} &c_{1,1}(x_1^{2,0}x_1^{2,0} - x_1^{1,0}x_1^{1,0}) + c_{1,2}(x_1^{2,0}x_2^{j,0} - x_1^{1,0}x_2^{j,0}) \\ =&c_{1,1}(x_1^{2,1}x_1^{2,1} - x_1^{1,1}x_1^{1,1}) + c_{1,2}(x_1^{2,1}x_2^{j,1} - x_1^{1,1}x_2^{j,1}). \end{aligned} \tag{5.1}$$

Due to the game condition defined in Def. 2.1, the queries by the adversary satisfy

$$\sum_{i,\theta\in[2]} c_{i,\theta} x_i^{f(i),0} x_\theta^{f(\theta),0} = \sum_{i,\theta\in[2]} c_{i,\theta} x_i^{f(i),1} x_\theta^{f(\theta),1} \tag{5.2}$$

$$\sum_{i,\theta\in[2]} c_{i,\theta} x_i^{g(i),0} x_\theta^{g(\theta),0} = \sum_{i,\theta\in[2]} c_{i,\theta} x_i^{g(i),1} x_\theta^{g(\theta),1} \tag{5.3}$$

where $f(i) = \begin{cases} 2 & (i=1) \\ j & (i=2) \end{cases}$, $g(i) = \begin{cases} 1 & (i=1) \\ j & (i=2) \end{cases}$. Note that Eq. (5.2) represents the restriction $f(x_1^{2,0}, x_2^{j,0}) = f(x_1^{2,1}, x_2^{j,1})$, and Eq. (5.3) represents the restriction

Additional sampling for qMSK	
$\ddot{u}_1, \ddot{u}_2 \leftarrow \mathbb{Z}_p$	
$\underline{\mathsf{qCT}_1^1}$ $\ddot{s}_1^1 \leftarrow \mathbb{Z}_p$ $\mathbf{b} := (\ x_1^{1,0},\ x_1^{1,1},\ s_1^1 w_{1,1},\ s_1^1 w_{2,1},\ u_1,\ t_1^1,\ 0,\ 0\ )$ $\widetilde{\mathbf{b}} := (\ \ 0,\ \ x_1^{1,1},\ \ \widetilde{s}_1^1,\ \ \ 0,\ \ \ r_1^1,\ v_1,\ 0,\ 0\ )$ $\mathbf{d} := (s_1^1, \ddot{s}_1^1),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^1, 0)$ $\mathbf{f} := (r_1^1, t_1^1, x_1^{1,1}x_1^{1,1} - x_1^{1,0}x_1^{1,0}, 0),\ h := 0$	$\underline{\mathsf{qCT}_2^1}$ $\ddot{s}_2^1 \leftarrow \mathbb{Z}_p$ $\mathbf{b} := (\ —, \ddot{s}_2^1 w_{1,2} + \ddot{u}_2 + \boxed{x_1^{2,1}x_2^{1,1}},\ 0\ )$ $\widetilde{\mathbf{b}} := (\ —,\ \ \ 0,\ \ \ \ 0\ )$ $\mathbf{d} := (s_2^1, \ddot{s}_2^1),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^1, 0)$ $\mathbf{f} := (r_2^1, t_2^1, x_1^{1,1}x_2^{1,1} - x_1^{1,0}x_2^{1,0}, 0),\ h := 0$
$\underline{\mathsf{qCT}_1^2}$ $\ddot{s}_1^2 \leftarrow \mathbb{Z}_p$ $\mathbf{b} := (\ —, \ddot{s}_1^2 w_{1,1} + \ddot{u}_1 + \boxed{x_1^{2,1}x_1^{2,1}},\ 0\ )$ $\widetilde{\mathbf{b}} := (\ —,\ \ \ 1,\ \ \ \ 0\ )$ $\mathbf{d} := (s_1^2, \ddot{s}_1^2),\ \widetilde{\mathbf{d}} := (0, 1)$ $\mathbf{f} := (0, t_1^2, x_1^{1,1}x_1^{1,1} - x_1^{1,0}x_1^{1,0}, 0),\ h := 1$	$\underline{\mathsf{qCT}_2^2}$ $\ddot{s}_2^2 \leftarrow \mathbb{Z}_p$ $\mathbf{b} := (\ —, \ddot{s}_2^2 w_{1,2} + \ddot{u}_2 + \boxed{x_1^{2,1}x_2^{2,1}},\ 0\ )$ $\widetilde{\mathbf{b}} := (\ —,\ \ \ 0,\ \ \ \ 0\ )$ $\mathbf{d} := (s_2^2, \ddot{s}_2^2),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^2, 0)$ $\mathbf{f} := (r_2^2, t_2^2, x_1^{1,1}x_2^{2,1} - x_1^{1,0}x_2^{2,0}, 0),\ h := 0$
$\underline{\mathsf{qSK}}$ $\widetilde{\mathbf{f}}_1 := (\sum_{\mu\in[2]} c_{1,\mu}u_\mu, \sum_{\mu\in[2]} c_{\mu,1}v_\mu, c_{1,1}, c_{2,1})$ $\widetilde{h}_1 := \sum_{\mu\in[2]} c_{1,\mu}\ddot{u}_\mu$	$\widetilde{\mathbf{f}}_2 := (\sum_{\mu\in[2]} c_{2,\mu}u_\mu, \sum_{\mu\in[2]} c_{\mu,2}v_\mu, c_{1,2}, c_{2,2})$ $\widetilde{h}_2 := 0$

Fig 14: Vectors in H_8.

$f(x_1^{1,0}, x_2^{j,0}) = f(x_1^{1,1}, x_2^{j,1})$. Eq. (5.2) − Eq. (5.3) implies Eq. (5.1) by reflecting the fact that $c_{2,1} = 0$, which is defined in Def. 2.2.

Thanks to the message-hiding property of 2-slot miFE and Eq. (5.1), we have

$$\{\mathsf{miPP}, \mathsf{miCT}_1^{1,0}, \mathsf{miCT}_2^{1,0}, \mathsf{miCT}_2^{2,0}, \mathsf{miSK}\} \approx_c \{\mathsf{miPP}, \mathsf{miCT}_1^{1,1}, \mathsf{miCT}_2^{1,1}, \mathsf{miCT}_2^{2,1}, \mathsf{miSK}\}$$

where

$$\begin{aligned}
\mathsf{miPP} &= (\mathbb{G}, [w_{1,1}]_1, [w_{1,2}]_1) \\
\mathsf{miCT}_1^{1,\beta} &= ([\ddot{s}_1^2]_1, [\ddot{s}_1^2 w_{1,1} + \ddot{u}_1 + x_1^{2,\beta}x_1^{2,\beta} - x_1^{1,\beta}x_1^{1,\beta}]_1) \\
\mathsf{miCT}_2^{j,\beta} &= ([\ddot{s}_2^j]_1, [\ddot{s}_2^j w_{1,2} + \ddot{u}_2 + \underbrace{x_1^{2,\beta}x_2^{j,\beta} - x_1^{1,\beta}x_2^{j,\beta}}_{\text{message vectors}}]_1) \\
\mathsf{miSK} &= (\sum_{\mu\in[2]} c_{1,\mu}\ddot{u}_\mu, -c_{1,1}w_{1,1}, -c_{1,2}w_{1,2}, \underbrace{c_{1,1}, c_{1,2}}_{\text{key vector}}).
\end{aligned}$$

Roughly speaking, $[\mathbf{b}]_1$ in $\mathsf{qCT}_1^2, \mathsf{qCT}_2^1, \mathsf{qCT}_2^2$ is simulatable from $\mathsf{miCT}_1^{1,\beta}, \mathsf{miCT}_2^{1,\beta}$, $\mathsf{miCT}_2^{2,\beta}$, respectively, and $[\widetilde{h}_1]_1$ in qSK is simulatable from miSK, and the case of $\beta = 0$ corresponds to H_7 and $\beta = 1$ corresponds to H_8.

$\underline{\mathsf{H}_8 \approx_c \mathsf{H}_9.}$ We can justify this indistinguishability by the SXDH assumption similarly to the case of $\mathsf{H}_6 \approx_c \mathsf{H}_7$.

$\underline{\mathsf{H}_9 \approx_c \mathsf{H}_{10}.}$ We can justify this indistinguishability by the (partially) function-hiding property of pFE, iFE, and gFE, similarly to the case of $\mathsf{G}_5 \approx_c \mathsf{H}_6$. At this point, all ciphertexts for slot 1 are changed from encryption of 0-side to that of 1-side.

$\underline{\mathsf{qCT}_1^1}$ $\mathbf{b} := (\ x_1^{1,0},\ x_1^{1,1},\ s_1^1 w_{1,1},\ s_1^1 w_{2,1},\ u_1,\ t_1^1,\ 0,\ 0\ )$ $\widetilde{\mathbf{b}} := (\ 0,\ x_1^{1,1},\ \widetilde{s}_1^1,\ 0,\ r_1^1,\ v_1,\ 0,\ 0\ )$ $\mathbf{d} := (s_1^1, \boxed{s_1^1 \widetilde{s}_1^2}),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^1, 0)$ $\mathbf{f} := (r_1^1, t_1^1, x_1^{1,1} x_1^{1,1} - x_1^{1,0} x_1^{1,0}, 0),\ h := 0$	$\underline{\mathsf{qCT}_2^1}$ $\mathbf{b} := (\ —, \boxed{s_2^1 \widetilde{s}_1^2 w_{1,2}} + \boxed{r_1^2 u_2} + x_1^{2,1} x_2^{1,1},\ 0\ )$ $\widetilde{\mathbf{b}} := (\ —,\ 0,\ 0\ )$ $\mathbf{d} := (s_2^1, \boxed{s_2^1 \widetilde{s}_1^2}),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^1, 0)$ $\mathbf{f} := (r_2^1, t_2^1, x_1^{1,1} x_2^{1,1} - x_1^{1,0} x_2^{1,0}, 0),\ h := 0$
$\underline{\mathsf{qCT}_1^2}$ $\mathbf{b} := (\ x_1^{2,0},\ x_1^{2,1},\ s_1^2 w_{1,1},\ s_1^2 w_{2,1},\ u_1,\ t_1^2,\ \boxed{s_1^2 \widetilde{s}_1^2 w_{1,1}} + \boxed{r_1^2 u_1} + x_1^{2,1} x_1^{2,1},\ 0\ )$ $\widetilde{\mathbf{b}} := (\ 0,\ 0,\ \widetilde{s}_1^2,\ 0,\ 0,\ v_1,\ 1,\ 0\ )$ $\mathbf{d} := (s_1^2, \boxed{s_1^2 \widetilde{s}_1^2}),\ \widetilde{\mathbf{d}} := (0, 1)$ $\mathbf{f} := (0, t_1^2, x_1^{1,1} x_1^{1,1} - x_1^{1,0} x_1^{1,0}, 0),\ h := 1$	$\underline{\mathsf{qCT}_2^2}$ $\mathbf{b} := (\ —, \boxed{s_2^2 \widetilde{s}_1^2 w_{1,2}} + \boxed{r_1^2 u_2} + x_1^{2,1} x_2^{2,1},\ 0\ )$ $\widetilde{\mathbf{b}} := (\ —,\ 0,\ 0\ )$ $\mathbf{d} := (s_2^2, \boxed{s_2^2 \widetilde{s}_1^2}),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^2, 0)$ $\mathbf{f} := (r_2^2, t_2^2, x_1^{1,1} x_2^{2,1} - x_1^{1,0} x_2^{2,0}, 0),\ h := 0$
$\underline{\mathsf{qSK}}$ $\widetilde{\mathbf{f}}_1 := (\sum_{\mu\in[2]} c_{1,\mu} u_\mu, \sum_{\mu\in[2]} c_{\mu,1} v_\mu, c_{1,1}, c_{2,1})$ $\widetilde{h}_1 := \boxed{r_1^2 \sum_{\mu\in[2]} c_{1,\mu} u_\mu}$	$\widetilde{\mathbf{f}}_2 := (\sum_{\mu\in[2]} c_{2,\mu} u_\mu, \sum_{\mu\in[2]} c_{\mu,2} v_\mu, c_{1,2}, c_{2,2})$ $\widetilde{h}_2 := 0$

Fig 15: Vectors in H_9.

$\underline{\mathsf{qCT}_1^1}$ $\mathbf{b} := (\ x_1^{1,0},\ x_1^{1,1},\ s_1^1 w_{1,1},\ s_1^1 w_{2,1},\ u_1,\ t_1^1,\ 0,\ 0)$ $\widetilde{\mathbf{b}} := (\ 0,\ x_1^{1,1},\ \widetilde{s}_1^1,\ 0,\ r_1^1,\ v_1,\ 0,\ 0)$ $\mathbf{d} := (s_1^1, \boxed{0}),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^1, 0)$ $\mathbf{f} := (r_1^1, t_1^1, x_1^{1,1} x_1^{1,1} - x_1^{1,0} x_1^{1,0}, 0),\ h := 0$	$\underline{\mathsf{qCT}_2^1}$ $\mathbf{b} := (\ x_2^{1,0},\ x_2^{1,1},\ s_2^1 w_{1,2},\ s_2^1 w_{2,2},\ u_2,\ t_2^1,\ \boxed{0},\ 0)$ $\widetilde{\mathbf{b}} := (\ x_2^{1,0},\ 0,\ 0,\ \widetilde{s}_2^1,\ r_2^1,\ v_2,\ 0,\ 0)$ $\mathbf{d} := (s_2^1, \boxed{0}),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^1, 0)$ $\mathbf{f} := (r_2^1, t_2^1, x_1^{1,1} x_2^{1,1} - x_1^{1,0} x_2^{1,0}, 0),\ h := 0$
$\underline{\mathsf{qCT}_1^2}$ $\mathbf{b} := (\ x_1^{2,0},\ x_1^{2,1},\ s_1^2 w_{1,1},\ s_1^2 w_{2,1},\ u_1,\ t_1^2,\ \boxed{0},\ 0)$ $\widetilde{\mathbf{b}} := (\ 0,\ \boxed{x_1^{2,1}},\ \boxed{\widetilde{s}_1^2},\ 0,\ \boxed{r_1^2},\ v_1,\ \boxed{0},\ 0)$ $\mathbf{d} := (s_1^2, \boxed{0}),\ \widetilde{\mathbf{d}} := (\boxed{\widetilde{s}_1^2}, \boxed{0})$ $\mathbf{f} := (r_1^2, t_1^2, x_1^{1,1} x_1^{1,1} - x_1^{1,0} x_1^{1,0}, 0),\ h := \boxed{0}$	$\underline{\mathsf{qCT}_2^2}$ $\mathbf{b} := (\ x_2^{2,0},\ x_2^{2,1},\ s_2^2 w_{1,2},\ s_2^2 w_{2,2},\ u_2,\ t_2^2,\ \boxed{0},\ 0)$ $\widetilde{\mathbf{b}} := (\ x_2^{2,0},\ 0,\ 0,\ \widetilde{s}_2^2,\ r_2^2,\ v_2,\ 0,\ 0)$ $\mathbf{d} := (s_2^2, \boxed{0}),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^2, 0)$ $\mathbf{f} := (r_2^2, t_2^2, x_1^{1,1} x_2^{2,1} - x_1^{1,0} x_2^{2,0}, 0),\ h := 0$
$\underline{\mathsf{qSK}}$ $\widetilde{\mathbf{f}}_1 := (\sum_{\mu\in[2]} c_{1,\mu} u_\mu, \sum_{\mu\in[2]} c_{\mu,1} v_\mu, c_{1,1}, c_{2,1})$ $\widetilde{h}_1 := \boxed{0}$	$\widetilde{\mathbf{f}}_2 := (\sum_{\mu\in[2]} c_{2,\mu} u_\mu, \sum_{\mu\in[2]} c_{\mu,2} v_\mu, c_{1,2}, c_{2,2})$ $\widetilde{h}_2 := 0$

Fig 16: Vectors in H_{10}.

$\underline{\mathsf{H}_{10} \approx_c \mathsf{H}_{11}.}$ As stated above, G_0 to H_{10} are hybrid games for processing the ciphertexts for slot 1. Next, we apply a similar procedure to slot 2. H_{11} in the process for slot 2 corresponds to H_7 in the process for slot 1. That is, $\mathsf{H}_{10} \approx_c \mathsf{H}_{11}$ can be proven similarly to $\mathsf{G}_0 \approx_c \mathsf{H}_7$.

$\underline{\mathsf{H}_{11} \approx_c \mathsf{H}_{12}.}$ This indistinguishability can be prove similarly to the case of $\mathsf{H}_7 \approx_c \mathsf{H}_8$, but we need an additional tweak in this case. First, we prove that, for all $j \in [2]$, we have

$$\begin{aligned} &c_{2,1}(x_2^{2,0} x_1^{j,0} - x_2^{1,0} x_1^{j,0}) + c_{2,2}(x_2^{2,0} x_2^{2,0} - x_2^{1,0} x_2^{1,0}) + c_{1,2}(x_1^{1,0} x_2^{2,0} - x_1^{1,0} x_2^{1,0}) \\ =&c_{2,1}(x_2^{2,1} x_1^{j,1} - x_2^{1,1} x_1^{j,1}) + c_{2,2}(x_2^{2,1} x_2^{2,1} - x_2^{1,1} x_2^{1,1}) + c_{1,2}(x_1^{1,1} x_2^{2,1} - x_1^{1,1} x_2^{1,1}). \end{aligned} \tag{5.4}$$

Due to the game condition defined in Def. 2.1, the queries by the adversary satisfy

$$\sum_{i,\theta\in[2]} c_{i,\theta} x_i^{f(i),0} x_\theta^{f(\theta),0} = \sum_{i,\theta\in[2]} c_{i,\theta} x_i^{f(i),1} x_\theta^{f(\theta),1} \tag{5.5}$$

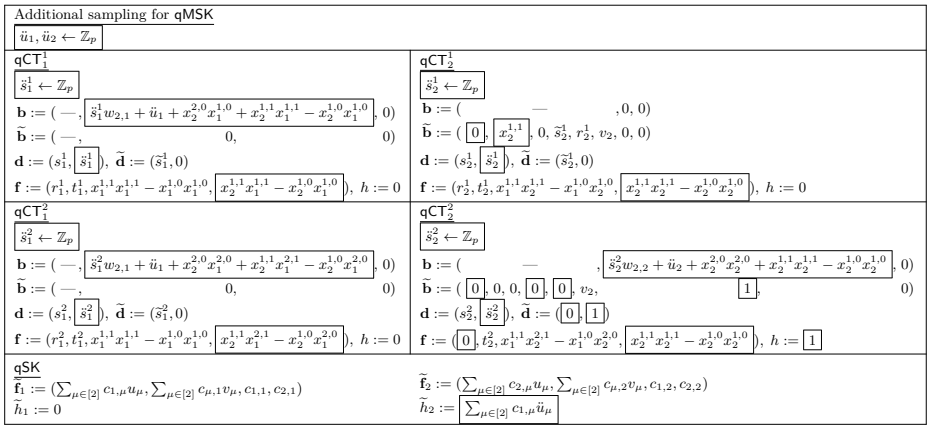

Fig 17: Vectors in H_{11}.

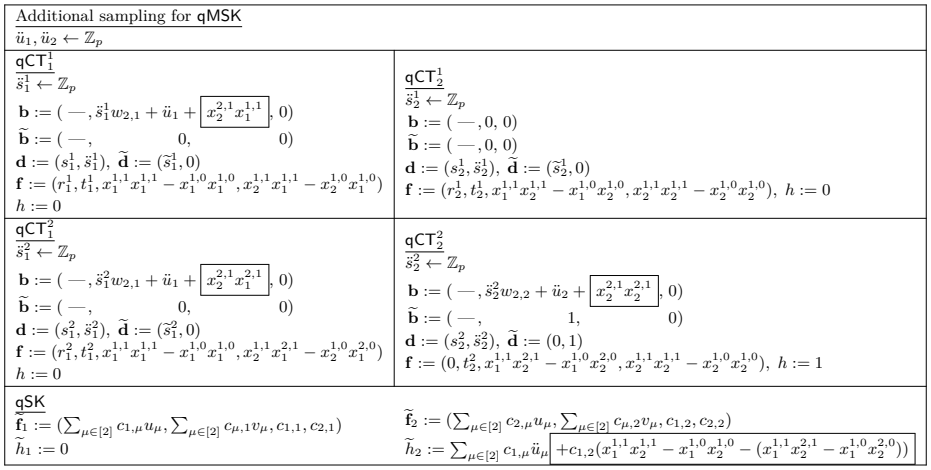

Fig 18: Vectors in H_{12}.

$$\sum_{i,\theta\in[2]} c_{i,\theta} x_i^{g(i),0} x_\theta^{g(\theta),0} = \sum_{i,\theta\in[2]} c_{i,\theta} x_i^{g(i),1} x_\theta^{g(\theta),1} \tag{5.6}$$

where $f(i) = \begin{cases} 1 & (i=1) \\ 2 & (i=2) \end{cases}$, $g(i) = \begin{cases} 1 & (i=1) \\ 1 & (i=2) \end{cases}$. Note that Eq. (5.5) represents the restriction $f(x_1^{1,0}, x_2^{2,0}) = f(x_1^{1,1}, x_2^{2,1})$, and Eq. (5.6) represents the restriction $f(x_1^{1,0}, x_2^{1,0}) = f(x_1^{1,1}, x_2^{1,1})$. Eq. (5.5) − Eq. (5.6) implies Eq. (5.4) by reflecting the fact that $c_{2,1} = 0$, which is defined in Def. 2.2.

Thanks to the message-hiding property of 3-slot miFE and Eq. (5.4), we have

$$\begin{aligned} &\{\mathsf{miPP}, \mathsf{miCT}_1^{1,0}, \mathsf{miCT}_1^{2,0}, \mathsf{miCT}_2^{1,0}, \mathsf{miCT}_3^{1,0}, \mathsf{miSK}\} \\ \approx_c &\{\mathsf{miPP}, \mathsf{miCT}_1^{1,1}, \mathsf{miCT}_1^{2,1}, \mathsf{miCT}_2^{1,1}, \mathsf{miCT}_3^{1,1}, \mathsf{miSK}\} \end{aligned}$$

Additional sampling for qMSK	
$\ddot{u}_1, \ddot{u}_2 \leftarrow \mathbb{Z}_p$	
$\underline{\mathsf{qCT}_1^1}$ $\ddot{s}_1^1 \leftarrow \mathbb{Z}_p$ $\mathbf{b} := (-, \ddot{s}_1^1 w_{2,1} + \ddot{u}_1 + x_2^{2,1} x_1^{1,1}, 0)$ $\widetilde{\mathbf{b}} := (-, 0, 0)$ $\mathbf{d} := (s_1^1, \ddot{s}_1^1),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^1, 0)$ $\mathbf{f} := (r_1^1, t_1^1, x_1^{1,1} x_1^{1,1} - x_1^{1,0} x_1^{1,0}, x_2^{1,1} x_1^{1,1} - x_2^{1,0} x_1^{1,0})$ $h := 0$	$\underline{\mathsf{qCT}_2^1}$ $\ddot{s}_2^1 \leftarrow \mathbb{Z}_p$ $\mathbf{b} := (-, 0, 0)$ $\widetilde{\mathbf{b}} := (-, 0, 0)$ $\mathbf{d} := (s_2^1, \ddot{s}_2^1),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^1, 0)$ $\mathbf{f} := (r_2^1, t_2^1, x_1^{1,1} x_2^{1,1} - x_1^{1,0} x_2^{1,0}, x_2^{1,1} x_2^{1,1} - x_2^{1,0} x_2^{1,0}),\ h := 0$
$\underline{\mathsf{qCT}_1^2}$ $\ddot{s}_1^2 \leftarrow \mathbb{Z}_p$ $\mathbf{b} := (-, \ddot{s}_1^2 w_{2,1} + \ddot{u}_1 + x_2^{2,1} x_1^{2,1}, 0)$ $\widetilde{\mathbf{b}} := (-, 0, 0)$ $\mathbf{d} := (s_1^2, \ddot{s}_1^2),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^2, 0)$ $\mathbf{f} := (r_1^2, t_1^2, x_1^{1,1} x_1^{1,1} - x_1^{1,0} x_1^{1,0}, \boxed{x_2^{1,1} x_1^{1,1} - x_2^{1,0} x_1^{1,0}})$ $h := 0$	$\underline{\mathsf{qCT}_2^2}$ $\ddot{s}_2^2 \leftarrow \mathbb{Z}_p$ $\mathbf{b} := (-, \ddot{s}_2^2 w_{2,2} + \ddot{u}_2 + x_2^{2,1} x_2^{2,1}, 0)$ $\widetilde{\mathbf{b}} := (-, 1, 0)$ $\mathbf{d} := (s_2^2, \ddot{s}_2^2),\ \widetilde{\mathbf{d}} := (0, 1)$ $\mathbf{f} := (0, t_2^2, \boxed{x_1^{1,1} x_2^{1,1} - x_1^{1,0} x_2^{1,0}}, x_2^{1,1} x_2^{1,1} - x_2^{1,0} x_2^{1,0}),\ h := 1$
$\underline{\mathsf{qSK}}$ $\widetilde{\mathbf{f}}_1 := (\sum_{\mu\in[2]} c_{1,\mu} u_\mu, \sum_{\mu\in[2]} c_{\mu,1} v_\mu, c_{1,1}, c_{2,1})$ $\widetilde{h}_1 := 0$	$\widetilde{\mathbf{f}}_2 := (\sum_{\mu\in[2]} c_{2,\mu} u_\mu, \sum_{\mu\in[2]} c_{\mu,2} v_\mu, c_{1,2}, c_{2,2})$ $\widetilde{h}_2 := \sum_{\mu\in[2]} c_{1,\mu} \ddot{u}_\mu + c_{1,2}(\cancel{x_1^{1,1} x_2^{1,1} - x_1^{1,0} x_2^{1,0}} - (\cancel{x_1^{1,1} x_2^{2,1} - x_1^{1,0} x_2^{2,0}}))$

Fig 19: Vectors in H_{13}.

$\underline{\mathsf{qCT}_1^1}$ $\mathbf{b} := (\ x_1^{1,0},\ x_1^{1,1},\ s_1^1 w_{1,1},\ s_1^1 w_{2,1},\ u_1,\ t_1^1,\ \boxed{0},\ 0)$ $\widetilde{\mathbf{b}} := (\ 0,\ x_1^{1,1},\ \widetilde{s}_1^1,\ 0,\ r_1^1,\ v_1,\ 0,\ 0)$ $\mathbf{d} := (s_1^1, \boxed{0}),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^1, 0)$ $\mathbf{f} := (r_1^1, t_1^1, x_1^{1,1} x_1^{1,1} - x_1^{1,0} x_1^{1,0}, x_2^{1,1} x_1^{1,1} - x_2^{1,0} x_1^{1,0}),\ h := 0$	$\underline{\mathsf{qCT}_2^1}$ $\mathbf{b} := (\ x_2^{1,0},\ x_2^{1,1},\ s_2^1 w_{1,2},\ s_2^1 w_{2,2},\ u_2,\ t_2^1,\ 0,\ 0)$ $\widetilde{\mathbf{b}} := (\ 0,\ x_2^{1,1},\ 0,\ \widetilde{s}_2^1,\ r_2^1,\ v_2,\ 0,\ 0)$ $\mathbf{d} := (s_2^1, \boxed{0}),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^1, 0)$ $\mathbf{f} := (r_2^1, t_2^1, x_1^{1,1} x_2^{1,1} - x_1^{1,0} x_2^{1,0}, x_2^{1,1} x_2^{1,1} - x_2^{1,0} x_2^{1,0}),\ h := 0$
$\underline{\mathsf{qCT}_1^2}$ $\mathbf{b} := (\ x_1^{2,0},\ x_1^{2,1},\ s_1^2 w_{1,1},\ s_1^2 w_{2,1},\ u_1,\ t_1^2,\ \boxed{0},\ 0)$ $\widetilde{\mathbf{b}} := (\ 0,\ x_1^{2,1},\ 0,\ \widetilde{s}_1^2,\ r_1^2,\ v_1,\ 0,\ 0)$ $\mathbf{d} := (s_1^2, \boxed{0}),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^2, 0)$ $\mathbf{f} := (r_1^2, t_1^2, x_1^{1,1} x_1^{1,1} - x_1^{1,0} x_1^{1,0}, x_2^{1,1} x_1^{1,1} - x_2^{1,0} x_1^{1,0}),\ h := 0$	$\underline{\mathsf{qCT}_2^2}$ $\mathbf{b} := (\ x_2^{2,0},\ x_2^{2,1},\ s_2^2 w_{1,2},\ s_2^2 w_{2,2},\ u_2,\ t_2^2,\ \boxed{0},\ 0)$ $\widetilde{\mathbf{b}} := (\ 0,\ \boxed{x_2^{2,1}},\ \boxed{\widetilde{s}_2^2},\ 0,\ \boxed{r_2^2},\ v_2,\ \boxed{0},\ 0)$ $\mathbf{d} := (s_2^2, \boxed{0}),\ \widetilde{\mathbf{d}} := (\boxed{\widetilde{s}_2^2}, \boxed{0})$ $\mathbf{f} := (\boxed{r_2^2}, t_2^2, x_1^{1,1} x_2^{1,1} - x_1^{1,0} x_2^{1,0}, x_2^{1,1} x_2^{1,1} - x_2^{1,0} x_2^{1,0}),\ h := \boxed{0}$
$\underline{\mathsf{qSK}}$ $\widetilde{\mathbf{f}}_1 := (\sum_{\mu\in[2]} c_{1,\mu} u_\mu, \sum_{\mu\in[2]} c_{\mu,1} v_\mu, c_{1,1}, c_{2,1})$ $\widetilde{h}_1 := 0$	$\widetilde{\mathbf{f}}_2 := (\sum_{\mu\in[2]} c_{2,\mu} u_\mu, \sum_{\mu\in[2]} c_{\mu,2} v_\mu, c_{1,2}, c_{2,2})$ $\widetilde{h}_2 := \boxed{0}$

Fig 20: Vectors in H_{14}.

$\underline{\mathsf{qCT}_1^1}$ $\mathbf{b} := (\ x_1^{1,0},\ x_1^{1,1},\ s_1^1 w_{1,1},\ s_1^1 w_{2,1},\ u_1,\ t_1^1,\ 0,\ 0)$ $\widetilde{\mathbf{b}} := (\ 0,\ x_1^{1,1},\ \widetilde{s}_1^1,\ 0,\ r_1^1,\ v_1,\ 0,\ 0)$ $\mathbf{d} := (s_1^1, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^1, 0)$ $\mathbf{f} := (r_1^1, t_1^1, \boxed{0}, \boxed{0}),\ h := 0$	$\underline{\mathsf{qCT}_2^1}$ $\mathbf{b} := (\ x_2^{1,0},\ x_2^{1,1},\ s_2^1 w_{1,2},\ s_2^1 w_{2,2},\ u_2,\ t_2^1,\ 0,\ 0)$ $\widetilde{\mathbf{b}} := (\ 0,\ x_2^{1,1},\ 0,\ \widetilde{s}_2^1,\ r_2^1,\ v_2,\ 0,\ 0)$ $\mathbf{d} := (s_2^1, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^1, 0)$ $\mathbf{f} := (r_2^1, t_2^1, \boxed{0}, \boxed{0}),\ h := 0$
$\underline{\mathsf{qCT}_1^2}$ $\mathbf{b} := (\ x_1^{2,0},\ x_1^{2,1},\ s_1^2 w_{1,1},\ s_1^2 w_{2,1},\ u_1,\ t_1^2,\ 0,\ 0)$ $\widetilde{\mathbf{b}} := (\ 0,\ x_1^{2,1},\ 0,\ \widetilde{s}_1^2,\ r_1^2,\ v_1,\ 0,\ 0)$ $\mathbf{d} := (s_1^2, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_1^2, 0)$ $\mathbf{f} := (r_1^2, t_1^2, \boxed{0}, \boxed{0}),\ h := 0$	$\underline{\mathsf{qCT}_2^2}$ $\mathbf{b} := (\ x_2^{2,0},\ x_2^{2,1},\ s_2^2 w_{1,2},\ s_2^2 w_{2,2},\ u_2,\ t_2^2,\ 0,\ 0)$ $\widetilde{\mathbf{b}} := (\ 0,\ x_2^{2,1},\ \widetilde{s}_2^2,\ 0,\ r_2^2,\ v_2,\ 0,\ 0)$ $\mathbf{d} := (s_2^2, 0),\ \widetilde{\mathbf{d}} := (\widetilde{s}_2^2, 0)$ $\mathbf{f} := (r_2^2, t_2^2, \boxed{0}, \boxed{0}),\ h := 0$
$\underline{\mathsf{qSK}}$ $\widetilde{\mathbf{f}}_1 := (\sum_{\mu\in[2]} c_{1,\mu} u_\mu, \sum_{\mu\in[2]} c_{\mu,1} v_\mu, \boxed{0}, \boxed{0})$ $\widetilde{h}_1 := 0$	$\widetilde{\mathbf{f}}_2 := (\sum_{\mu\in[2]} c_{2,\mu} u_\mu, \sum_{\mu\in[2]} c_{\mu,2} v_\mu, \boxed{0}, \boxed{0})$ $\widetilde{h}_2 := 0$

Fig 21: Vectors in H_{15}.

where

$$\mathsf{miPP} = (\mathbb{G}, [w_{2,1}]_1, [w_{2,2}]_1, [w_{2,3}]_1)$$
$$\mathsf{miCT}_1^{j,\beta} = ([\ddot{s}_1^j]_1, [\ddot{s}_1^j w_{2,1} + \ddot{u}_1 + x_2^{2,\beta} x_1^{j,\beta} - x_2^{1,\beta} x_1^{j,\beta}]_1)$$
$$\mathsf{miCT}_2^{1,\beta} = ([\ddot{s}_2^2]_1, [\ddot{s}_2^2 w_{2,2} + \ddot{u}_2 + x_2^{2,\beta} x_2^{2,\beta} - x_2^{1,\beta} x_2^{1,\beta}]_1)$$
$$\mathsf{miCT}_3^{1,\beta} = ([\ddot{s}_3^1]_1, [\ddot{s}_3^1 w_{2,3} + \ddot{u}_3 + \underbrace{x_1^{1,\beta} x_2^{2,\beta} - x_1^{1,\beta} x_2^{1,\beta}}_{\text{message vectors}}]_1)$$
$$\mathsf{miSK} = (\sum_{\mu\in[2]} c_{2,\mu}\ddot{u}_\mu + c_{1,2}\ddot{u}_3, -c_{2,1}w_{2,1}, -c_{2,2}w_{2,2}, -c_{1,2}w_{2,3}, \underbrace{c_{2,1}, c_{2,2}, c_{1,2}}_{\text{key vector}}).$$

Roughly speaking, $[\mathbf{b}]_1$ in $\mathsf{qCT}_1^1, \mathsf{qCT}_1^2, \mathsf{qCT}_2^2$ is simulatable from $\mathsf{miCT}_1^{1,\beta}, \mathsf{miCT}_1^{2,\beta}$, $\mathsf{miCT}_2^{1,\beta}$, respectively, and $[\widetilde{h}_2]_1$ in qSK is simulatable from miSK and $\mathsf{miCT}_3^{1,\beta}$. More precisely,

$$\widetilde{h}_2 = \mathsf{K}_1 - \mathsf{C}_1\mathsf{K}_4 - c_{1,2}(\mathsf{C}_2 + x_1^{1,0}x_2^{2,0} - x_1^{1,0}x_2^{1,0})$$

where $\mathsf{miCT}_3^{1,\beta} = ([\mathsf{C}_1]_1, [\mathsf{C}_2]_1)$ and $\mathsf{miSK} = (\mathsf{K}_1, \ldots, \mathsf{K}_7)$. The case of $\beta = 0$ corresponds to H_{11} and $\beta = 1$ corresponds to H_{12}.

$\underline{\mathsf{H}_{12} \approx_c \mathsf{H}_{13}.}$ We can justify this indistinguishability by the function-hiding property of gFE. For all $i, j \in [2]$, $\langle \mathbf{f}_i^j, \widetilde{\mathbf{f}}_i \rangle + h_i^j \widetilde{h}_i$ in H_{12} and that in H_{13} are equal (recall that $c_{2,1} = 0$), which implies, for all $j_1, j_2 \in [2]$, $\sum_{i\in[2]}(\langle \mathbf{f}_i^{j_i}, \widetilde{\mathbf{f}}_i \rangle + h_i^{j_i}\widetilde{h}_i)$ in H_{12} and that in H_{13} are equal. Thus, the indistinguishability of $\{\mathbf{f}, \widetilde{\mathbf{f}}, h, \widetilde{h}\}$ between H_{12} and H_{13} is implied by the function-hiding property of gFE.

$\underline{\mathsf{H}_{13} \approx_c \mathsf{H}_{14}.}$ This indistinguishability can be proven similarly to $\mathsf{H}_8 \approx_c \mathsf{H}_{10}$.

$\underline{\mathsf{H}_{14} \approx_c \mathsf{H}_{15}.}$ Due to the game condition defined in Def. 2.1, the queries by the adversary satisfy $\sum_{i,\theta\in[2]} c_{i,\theta}(x_i^{1,1}x_\theta^{1,1} - x_i^{1,0}x_\theta^{1,0}) = 0$, which implies, for all $j_1, j_2 \in [2]$, $\sum_{i\in[2]}(\langle \mathbf{f}_i^{j_i}, \widetilde{\mathbf{f}}_i \rangle + h_i^{j_i}\widetilde{h}_i)$ in H_{14} and that in H_{15} are equal. Thus, the indistinguishability of $\{\mathbf{f}, \widetilde{\mathbf{f}}\}$ between H_{14} and H_{15} is implied by the function-hiding property of gFE.

$\underline{\mathsf{H}_{15} \approx_c \mathsf{G}_1.}$ It is easy to see that this indistinguishability is implied by the partially function-hiding property of pFE, since, for all $i, j, I, J \in [2]$, $\langle \mathbf{b}_i^j, \widetilde{\mathbf{b}}_I^J \rangle$ in H_{15} and that in G_1 are equal.

6 Quadratic Multi-Input Functional Encryption

We present our quadratic MIFE scheme for $\mathcal{F}_{m,n,X,C}^{\mathsf{MQF}}$. We define the following functions that relate indices in $[n] \times [m]$ with those in $[mn]$:

- location function, $\mathsf{lo} : [n] \times [m] \to [mn]$, defined as $\mathsf{lo}(x, y) = (x-1)m + y$;
- location set function, $\mathsf{ls} : [n] \to 2^{[mn]}$, defined as $\mathsf{ls}(x) = \{\mathsf{lo}(x, 1), \ldots, \mathsf{lo}(x, m)\}$;
- slot function, $\mathsf{sl} : [mn] \to [n]$, defined as $\mathsf{sl}(x) = \lceil x/m \rceil$;

$\underline{\mathsf{qSetup}(1^\lambda) \to \mathsf{qPP}, \mathsf{qMSK}}$
$\mathbb{G} \leftarrow \mathcal{G}_{\mathsf{BG}}(1^\lambda), \mathbf{A}_1, \ldots, \mathbf{A}_n \leftarrow \mathcal{D}_k,\ \{\mathbf{w}_{i,j}\}_{i,j\in[mn]} \leftarrow \mathbb{Z}_p^{k+1},\ \widetilde{\mathbf{U}}_1, \ldots, \widetilde{\mathbf{U}}_{mn} \leftarrow \mathbb{Z}_p^{k\times k}$
$\mathbf{u}_1, \ldots, \mathbf{u}_{mn} \leftarrow \mathbb{Z}_p^k,\ \mathbf{V}_1, \ldots, \mathbf{V}_{mn} \leftarrow \mathbb{Z}_p^{k\times k},\ \widetilde{\mathbf{v}}_1, \ldots, \widetilde{\mathbf{v}}_{mn} \leftarrow \mathbb{Z}_p^k$
$\mathsf{pPP}, \mathsf{pMSK} \leftarrow \mathsf{pSetup}(1^\lambda),\ \mathsf{iPP}, \mathsf{iMSK} \leftarrow \mathsf{iSetup}(1^\lambda),\ \mathsf{gPP}, \mathsf{gMSK} \leftarrow \mathsf{gSetup}(1^\lambda)$
$\mathsf{qPP} := (\mathbb{G}, \mathsf{pPP}, \mathsf{iPP}, \mathsf{gPP})$
$\mathsf{qMSK} := (\{\mathbf{A}_i\}_{i\in[n]}, \{\mathbf{w}_{i,j}\}_{i,j\in[mn]}, \{\widetilde{\mathbf{U}}_i, \mathbf{u}_i, \mathbf{V}_i, \widetilde{\mathbf{v}}_i\}_{i\in[mn]}, \mathsf{pMSK}, \mathsf{iMSK}, \mathsf{gMSK}).$

$\underline{\mathsf{qEnc}(\mathsf{qMSK}, i, x_i) \to \mathsf{qCT}_i}$
$\mathbf{S} \leftarrow \mathbb{Z}_p^{k\times k},\ \widetilde{\mathbf{s}}, \mathbf{r}, \mathbf{t} \leftarrow \mathbb{Z}_p^k,\ L \leftarrow \mathbb{Z}_p,\ \mathbf{l} := \mathbf{e}_{i/n} \otimes (1, L) \in \mathbb{Z}_p^{2n},\ \widetilde{\mathbf{l}} := \mathbf{e}_{i/n} \otimes (L, -1) \in \mathbb{Z}_p^{2n}$
$\mathbf{b}_{\kappa,1} := (x_{i,\kappa}, 0) \in \mathbb{Z}_p^2,\ \mathbf{b}_{\kappa,2} := (\mathbf{w}_{\mathsf{lo}(i,\kappa)}^\top(\mathbf{I}_{mn} \otimes \mathbf{A}_i\mathbf{S}), \mathbf{u}_{\mathsf{lo}(i,\kappa)}) \in \mathbb{Z}_p^{(mn+1)k}$
$\mathbf{b}_{\kappa,3} := \mathbf{t}^\top\mathbf{V}_{\mathsf{lo}(i,\kappa)} \in \mathbb{Z}_p^k,\ \mathbf{b}_{\kappa,4} = \mathbf{b}_{\kappa,5} := \mathbf{0} \in \mathbb{Z}_p^m,\ \mathbf{b}_{\kappa,6} := \mathbf{0} \in \mathbb{Z}_p^{km},\ \mathbf{b}_\kappa := (\mathbf{b}_{\kappa,1}, \ldots, \mathbf{b}_{\kappa,6})$
$\widetilde{\mathbf{b}}_{\kappa,1} := (x_{i,\kappa}, 0) \in \mathbb{Z}_p^2,\ \widetilde{\mathbf{b}}_{\kappa,2} := (\mathbf{e}_{\mathsf{lo}(i,\kappa)/mn} \otimes \widetilde{\mathbf{s}}, \mathbf{r}^\top\widetilde{\mathbf{U}}_{\mathsf{lo}(i,\kappa)}) \in \mathbb{Z}_p^{(mn+1)k}$
$\widetilde{\mathbf{b}}_{\kappa,3} := \widetilde{\mathbf{v}}_{\mathsf{lo}(i,\kappa)}^\top \in \mathbb{Z}_p^k,\ \widetilde{\mathbf{b}}_{\kappa,4} = \widetilde{\mathbf{b}}_{\kappa,5} := \mathbf{0} \in \mathbb{Z}_p^m,\ \widetilde{\mathbf{b}}_{\kappa,6} := \mathbf{0} \in \mathbb{Z}_p^{km},\ \widetilde{\mathbf{b}}_\kappa := (\widetilde{\mathbf{b}}_{\kappa,1}, \ldots, \widetilde{\mathbf{b}}_{\kappa,6})$
$\mathbf{d}_\tau := (\mathbf{a}_{i,\tau}^\top\mathbf{S}, 0) \in \mathbb{Z}_p^{k+1},\ \widetilde{\mathbf{d}} := (\widetilde{\mathbf{s}}, 0) \in \mathbb{Z}_p^{k+1}$
$\mathbf{f}_1 := (\mathbf{r}, \mathbf{t}) \in \mathbb{Z}_p^{2k},\ \mathbf{f}_{2,1} = \cdots = \mathbf{f}_{2,n} := \mathbf{0} \in \mathbb{Z}_p^{m^2},\ \mathbf{f} := (\mathbf{f}_1, \mathbf{f}_{2,1}, \ldots, \mathbf{f}_{2,n}),\ h := 0$
$\mathsf{pCT}_{\mathsf{lo}(i,\kappa)} \leftarrow \mathsf{pEnc}(\mathsf{pMSK}, (\mathbf{l}, [\mathbf{b}_\kappa]_1)),\ \mathsf{pSK}_{\mathsf{lo}(i,\kappa)} \leftarrow \mathsf{pKeyGen}(\mathsf{pMSK}, (\widetilde{\mathbf{l}}, [\widetilde{\mathbf{b}}_\kappa]_2))$
$\mathsf{iCT}_{i,\tau} \leftarrow \mathsf{iEnc}(\mathsf{iMSK}, [\mathbf{d}_\tau]_1),\ \mathsf{iSK}_i \leftarrow \mathsf{iKeyGen}(\mathsf{iMSK}, [\widetilde{\mathbf{d}}]_2),\ \mathsf{gCT}_i \leftarrow \mathsf{gEnc}(\mathsf{gMSK}, i, ([\mathbf{f}]_1, [h]_2))$
$\mathsf{qCT}_i := (\{\mathsf{pCT}_{\mathsf{lo}(i,\kappa)}, \mathsf{pSK}_{\mathsf{lo}(i,\kappa)}\}_{\kappa\in[m]}, \{\mathsf{iCT}_{i,\tau}\}_{\tau\in[k+1]}, \mathsf{iSK}_i, \mathsf{gCT}_i).$

$\underline{\mathsf{qKeyGen}(\mathsf{qMSK}, \mathbf{c} = \{c_{\mu,\nu}\}_{\mu,\nu\in[2]}) \to \mathsf{qSK}}$
$\widetilde{\mathbf{f}}_{i,1} := \left(\sum_{\substack{\mu\in\mathsf{ls}(i)\\ \nu\in[mn]}} c_{\mu,\nu}\widetilde{\mathbf{U}}_\mu\mathbf{u}_\nu, \sum_{\substack{\mu\in[mn]\\ \nu\in\mathsf{ls}(i)}} c_{\mu,\nu}\mathbf{V}_\nu\widetilde{\mathbf{v}}_\mu\right) \in \mathbb{Z}_p^{2k},\ \widetilde{\mathbf{f}}_{i,2,1} = \cdots = \widetilde{\mathbf{f}}_{i,2,n} := \mathbf{0} \in \mathbb{Z}_p^{m^2}$
$\widetilde{\mathbf{f}}_i := (\widetilde{\mathbf{f}}_{i,1}, \widetilde{\mathbf{f}}_{i,2,1}, \ldots, \widetilde{\mathbf{f}}_{i,2,n}),\ \widetilde{h}_i := 0,\ \boldsymbol{\sigma}_{i,\theta} := \sum_{\substack{\mu\in\mathsf{ls}(i),\\ \nu\in\mathsf{ls}(\theta)}} c_{\mu,\nu}\mathbf{w}_{\mu,\nu} \in \mathbb{Z}_p^{k+1}$
$\mathsf{gSK} \leftarrow \mathsf{gKeyGen}(\mathsf{gMSK}, \{[\widetilde{\mathbf{f}}_i]_2, [\widetilde{h}_i]_1\}_{i\in[n]}),\ \mathsf{qSK} := (\mathbf{c}, \mathsf{gSK}, \{\boldsymbol{\sigma}_{i,\theta}\}_{i,\theta\in[n]}).$

$\underline{\mathsf{qDec}(\mathsf{qCT}_1, , \ldots, \mathsf{qCT}_n \mathsf{qSK}) \to z}$
$[z_1]_T := \prod_{\mu,\nu\in[mn]} \mathsf{pDec}(\mathsf{pCT}_\nu, \mathsf{pSK}_\mu)^{c_{\mu,\nu}},\ [\mathbf{z}_{2,i,\theta}]_T := (\mathsf{iDec}(\mathsf{iCT}_{\theta,1}, \mathsf{iSK}_i), \ldots, \mathsf{iDec}(\mathsf{iCT}_{\theta,k+1}, \mathsf{iSK}_i))$
$[z_3]_T := \mathsf{gDec}(\mathsf{gCT}_1, \ldots, \mathsf{gCT}_n, \mathsf{gSK}),\ [z]_T := [z_1 - \sum_{i,\theta\in[n]}\langle\mathbf{z}_{2,i,\theta}, \boldsymbol{\sigma}_{i,\theta}\rangle - z_3]_T.$
Searches for z within the range of $z \le |m^2n^2CX^2|$

Fig 22: Our n-input quadratic MIFE scheme.

– entry function, $\mathsf{en} : [mn] \to [m]$, defined as $\mathsf{en}(x) = x - m(\mathsf{sl}(x) - 1)$.

Note that we have $\mathsf{lo}(\mathsf{sl}(x), \mathsf{en}(x)) = x$ for all $x \in [mn]$. Let $\mathcal{D}_k$ be a matrix distribution. Let $\mathsf{pFE} = (\mathsf{pSetup}, \mathsf{pEnc}, \mathsf{pKeyGen}, \mathsf{pDec})$ be an FE scheme for $\mathcal{F}^{\mathsf{PIP}}_{2n,2+(mn+2)k+(2+k)m,\mathbb{G}}$ (Def. 3.2), $\mathsf{iFE} = (\mathsf{iSetup}, \mathsf{iEnc}, \mathsf{iKeyGen}, \mathsf{iDec})$ be an FE scheme for $\mathcal{F}^{\mathsf{IP}}_{k+1,\mathbb{G}}$ (Def. 3.1), and $\mathsf{gFE} = (\mathsf{gSetup}, \mathsf{gEnc}, \mathsf{gKeyGen}, \mathsf{gDec})$ be an FE scheme for $\mathcal{F}^{\mathsf{MGIP}}_{2k+m^2n,1,n,\mathbb{G}}$ (Def. 4.2). We construct our quadratic MIFE scheme $\mathsf{qFE} = (\mathsf{qSetup}, \mathsf{qEnc}, \mathsf{qKeyGen}, \mathsf{qDec})$ from pFE, iFE, and gFE as shown in Fig 22.

Due to space constraints, we present the proof of correctness and security analysis of our scheme in the full version.

References

1. Abdalla, M., Benhamouda, F., Gay, R.: From single-input to multi-client inner-product functional encryption. In: Galbraith, S.D., Moriai, S. (eds.) ASIACRYPT 2019, Part III. LNCS, vol. 11923, pp. 552–582. Springer, Heidelberg (Dec 2019)

2. Abdalla, M., Benhamouda, F., Kohlweiss, M., Waldner, H.: Decentralizing inner-product functional encryption. In: Lin, D., Sako, K. (eds.) PKC 2019, Part II. LNCS, vol. 11443, pp. 128–157. Springer, Heidelberg (Apr 2019)
3. Abdalla, M., Bourse, F., De Caro, A., Pointcheval, D.: Simple functional encryption schemes for inner products. In: Katz, J. (ed.) PKC 2015. LNCS, vol. 9020, pp. 733–751. Springer, Heidelberg (Mar / Apr 2015)
4. Abdalla, M., Catalano, D., Fiore, D., Gay, R., Ursu, B.: Multi-input functional encryption for inner products: Function-hiding realizations and constructions without pairings. In: Shacham, H., Boldyreva, A. (eds.) CRYPTO 2018, Part I. LNCS, vol. 10991, pp. 597–627. Springer, Heidelberg (Aug 2018)
5. Abdalla, M., Catalano, D., Gay, R., Ursu, B.: Inner-product functional encryption with fine-grained access control. Cryptology ePrint Archive, Report 2020/577 (2020), https://eprint.iacr.org/2020/577
6. Abdalla, M., Gay, R., Raykova, M., Wee, H.: Multi-input inner-product functional encryption from pairings. In: Coron, J.S., Nielsen, J.B. (eds.) EUROCRYPT 2017, Part I. LNCS, vol. 10210, pp. 601–626. Springer, Heidelberg (Apr / May 2017)
7. Ananth, P., Jain, A., Lin, H., Matt, C., Sahai, A.: Indistinguishability obfuscation without multilinear maps: New paradigms via low degree weak pseudorandomness and security amplification. In: Boldyreva, A., Micciancio, D. (eds.) CRYPTO 2019, Part III. LNCS, vol. 11694, pp. 284–332. Springer, Heidelberg (Aug 2019)
8. Ananth, P., Jain, A.: Indistinguishability obfuscation from compact functional encryption. In: Gennaro, R., Robshaw, M.J.B. (eds.) CRYPTO 2015, Part I. LNCS, vol. 9215, pp. 308–326. Springer, Heidelberg (Aug 2015)
9. Baltico, C.E.Z., Catalano, D., Fiore, D., Gay, R.: Practical functional encryption for quadratic functions with applications to predicate encryption. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017, Part I. LNCS, vol. 10401, pp. 67–98. Springer, Heidelberg (Aug 2017)
10. Bishop, A., Jain, A., Kowalczyk, L.: Function-hiding inner product encryption. In: Iwata, T., Cheon, J.H. (eds.) ASIACRYPT 2015, Part I. LNCS, vol. 9452, pp. 470–491. Springer, Heidelberg (Nov / Dec 2015)
11. Bitansky, N., Vaikuntanathan, V.: Indistinguishability obfuscation from functional encryption. In: Guruswami, V. (ed.) 56th FOCS. pp. 171–190. IEEE Computer Society Press (Oct 2015)
12. Boneh, D., Sahai, A., Waters, B.: Functional encryption: Definitions and challenges. In: Ishai, Y. (ed.) TCC 2011. LNCS, vol. 6597, pp. 253–273. Springer, Heidelberg (Mar 2011)
13. Brakerski, Z., Segev, G.: Function-private functional encryption in the private-key setting. In: Dodis, Y., Nielsen, J.B. (eds.) TCC 2015, Part II. LNCS, vol. 9015, pp. 306–324. Springer, Heidelberg (Mar 2015)
14. Chen, J., Wee, H.: Semi-adaptive attribute-based encryption and improved delegation for Boolean formula. In: Abdalla, M., Prisco, R.D. (eds.) SCN 14. LNCS, vol. 8642, pp. 277–297. Springer, Heidelberg (Sep 2014)
15. Chotard, J., Dufour Sans, E., Gay, R., Phan, D.H., Pointcheval, D.: Decentralized multi-client functional encryption for inner product. In: Peyrin, T., Galbraith, S. (eds.) ASIACRYPT 2018, Part II. LNCS, vol. 11273, pp. 703–732. Springer, Heidelberg (Dec 2018)
16. Datta, P., Dutta, R., Mukhopadhyay, S.: Functional encryption for inner product with full function privacy. In: Cheng, C.M., Chung, K.M., Persiano, G., Yang, B.Y. (eds.) PKC 2016, Part I. LNCS, vol. 9614, pp. 164–195. Springer, Heidelberg (Mar 2016)

17. Datta, P., Okamoto, T., Tomida, J.: Full-hiding (unbounded) multi-input inner product functional encryption from the k-Linear assumption. In: Abdalla, M., Dahab, R. (eds.) PKC 2018, Part II. LNCS, vol. 10770, pp. 245–277. Springer, Heidelberg (Mar 2018)
18. Escala, A., Herold, G., Kiltz, E., Ràfols, C., Villar, J.L.: An algebraic framework for Diffie-Hellman assumptions. Journal of Cryptology 30(1), 242–288 (Jan 2017)
19. Garg, S., Gentry, C., Halevi, S., Raykova, M., Sahai, A., Waters, B.: Candidate indistinguishability obfuscation and functional encryption for all circuits. In: 54th FOCS. pp. 40–49. IEEE Computer Society Press (Oct 2013)
20. Garg, S., Gentry, C., Halevi, S., Zhandry, M.: Functional encryption without obfuscation. In: Kushilevitz, E., Malkin, T. (eds.) TCC 2016-A, Part II. LNCS, vol. 9563, pp. 480–511. Springer, Heidelberg (Jan 2016)
21. Gay, R.: A new paradigm for public-key functional encryption for degree-2 polynomials. In: Kiayias, A., Kohlweiss, M., Wallden, P., Zikas, V. (eds.) PKC 2020, Part I. LNCS, vol. 12110, pp. 95–120. Springer, Heidelberg (May 2020)
22. Goldwasser, S., Gordon, S.D., Goyal, V., Jain, A., Katz, J., Liu, F.H., Sahai, A., Shi, E., Zhou, H.S.: Multi-input functional encryption. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 578–602. Springer, Heidelberg (May 2014)
23. Goyal, R., Koppula, V., Waters, B.: Semi-adaptive security and bundling functionalities made generic and easy. In: Hirt, M., Smith, A.D. (eds.) TCC 2016-B, Part II. LNCS, vol. 9986, pp. 361–388. Springer, Heidelberg (Oct / Nov 2016)
24. Jain, A., Lin, H., Sahai, A.: Indistinguishability obfuscation from well-founded assumptions. Cryptology ePrint Archive, Report 2020/1003 (2020), https://eprint.iacr.org/2020/1003
25. Katz, J., Sahai, A., Waters, B.: Predicate encryption supporting disjunctions, polynomial equations, and inner products. In: Smart, N.P. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 146–162. Springer, Heidelberg (Apr 2008)
26. Kim, S., Lewi, K., Mandal, A., Montgomery, H., Roy, A., Wu, D.J.: Function-hiding inner product encryption is practical. In: Catalano, D., De Prisco, R. (eds.) SCN 18. LNCS, vol. 11035, pp. 544–562. Springer, Heidelberg (Sep 2018)
27. Libert, B., Titiu, R.: Multi-client functional encryption for linear functions in the standard model from LWE. In: Galbraith, S.D., Moriai, S. (eds.) ASIACRYPT 2019, Part III. LNCS, vol. 11923, pp. 520–551. Springer, Heidelberg (Dec 2019)
28. Lin, H.: Indistinguishability obfuscation from SXDH on 5-linear maps and locality-5 PRGs. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017, Part I. LNCS, vol. 10401, pp. 599–629. Springer, Heidelberg (Aug 2017)
29. O'Neill, A.: Definitional issues in functional encryption. Cryptology ePrint Archive, Report 2010/556 (2010), http://eprint.iacr.org/2010/556
30. Tomida, J.: Tightly secure inner product functional encryption: Multi-input and function-hiding constructions. In: Galbraith, S.D., Moriai, S. (eds.) ASIACRYPT 2019, Part III. LNCS, vol. 11923, pp. 459–488. Springer, Heidelberg (Dec 2019)
31. Tomida, J., Abe, M., Okamoto, T.: Efficient functional encryption for inner-product values with full-hiding security. In: Bishop, M., Nascimento, A.C.A. (eds.) ISC 2016. LNCS, vol. 9866, pp. 408–425. Springer, Heidelberg (Sep 2016)